Pega Customer Decision Hub User Guide

Pega Customer Decision Hub uses decision management and AI technology for the purpose of transforming how an enterprise interacts with its customers to be one-to-one, in-context, omnichannel, and relevant.

Pega Customer Decision Hub

This 1:1 Customer Engagement enables you to take full advantage of every opportunity to interact with a customer by speaking about what is relevant to that customer in the context they are in at that moment, regardless of which channel the interaction takes place on. In most businesses, being relevant also means being empathetic and profitable in the long run. These goals are not at odds but they do require listening carefully and learning from each of those customer interactions.

As some of the largest and most forward-thinking enterprises in the world have used Pega's decision management and AI technology to achieve tremendous returns, we are pleased to offer a prepackaged set of best practices for applying this technology in a way that is performance-friendly and practical, both in business and technical terms. Those practices are embodied within the product through the use of Next-Best-Action Designer. To help you map Pega Customer Decision Hub capabilities to your enterprise without starting from scratch, Pega offers accelerator templates for several industries, available on our Pega Marketplace.

In addition to centralized intelligence, Pega Customer Decision Hub also provides the means to activate that intelligence over real-world marketing channels, such as websites, e-mail, SMS, Facebook, and Google Ads. Although the product offers first-rate capabilities for executing segment-driven outbound campaigns, the fastest way to create intelligence that can naturally scale out over multiple channels is by using Next-Best-Action Designer.

Pega Customer Decision Hub encompasses not only the foundational decision management technology of the Pega Platform, but also the application of that technology for 1:1 Customer Engagement and the channel connections to activate the intelligence. The product formerly known as Pega Marketing has been incorporated into Pega Customer Decision Hub from 8.5 onwards.

This documentation is meant to not only explain how the product works technically, but to also offer guidance as to which features and functions best represent the future of 1:1 Customer Engagement, as well as advising the reader on which functions lead towards outdated approaches. These documents are a companion to a suite of Pega Academy courses which are designed to impart the philosophy of one-to-one decision making in the context of simple end-to-end scenarios that culminate in a working product. While the courses are not a substitute for this documentation, if you are new to our technology, we invite you to start with Cross-Sell on the Web, which teaches the concepts using an inbound channel or Always-On Outbound Customer Engagement, which teaches the same concepts, but with an outbound channel. Pega welcomes you and wishes you well on your path to transforming how your enterprise interacts with your current and future customers to being empathetic and relevant!

Understanding the relationship between Pega Customer Decision Hub, Pega Marketing, and decision management

Starting in version 8.5, Pega Customer Decision Hub is Pega's main solution for one-to-one customer engagement, combining the features available in previous versions of Pega Marketing and Pega Customer Decision Hub with many new features.

Pega Customer Decision Hub

Pega Customer Decision Hub and Pega Marketing

In versions prior to 8.5, Pega Customer Decision Hub was a one-to-one customer engagement application bundled with Pega Marketing. Some features of Pega Marketing, such as campaigns or microsites, were not available in Pega Customer Decision Hub.

In version 8.5, the two products were streamlined into the new Pega Customer Decision Hub, which now has all the features previously available only in Pega Marketing. Pega Marketing is no longer available as a separate product. If you have a Pega Marketing application, you can easily update it to Pega Customer Decision Hub 8.6. For more information, see Pega Customer Decision Hub Update Guides.

Pega Customer Decision Hub and decision management

Decision management (also known as decisioning or previously, Decision Strategy Manager or DSM) is a Pega Platform capability that provides technology for decision making, such as proposition management, Interaction History, adaptive and predictive models, decision strategies, and other features that Pega Customer Decision Hub builds upon. Decision management is intended for organizations that want to make intelligent, automated decisions in customer service and process automation. For example, in customer service, you can use decisioning to suggest the most relevant set of tasks to a call center agent who is speaking with a customer. In process automation, you can use analytics to route the cases to the right service agent based on the agent's skills and past performance in handling similar cases.

If you built your own marketing or customer engagement application based on the decision management capability of Pega Platform, you can migrate to Pega Customer Decision Hub 8.6 to use Pega's best practice approach to omnichannel one-to-one customer engagement:

1. Install Pega Customer Decision Hub.

   For more information, see Pega Customer Decision Hub Installation Guide.

2. Create a Pega Customer Decision Hub application.

   For more information, see Pega Customer Decision Hub Implementation Guide.

3. In the Pega Customer Decision Hub application, configure the artifacts and settings required to implement next best actions in your organization.

   For more information, see:

   - Improving customer experience with next-best-action on owned channels
   - Pega Decisioning Consultant

4. Enable your business users to access the Pega Customer Decision Hub portal in order to configure various marketing and decisioning artifacts, such as actions and treatments.

Improving customer experience with next-best-action on owned channels

Next-best-action provides personalized decisions based on predictive and adaptive analytics, and traditional business rules such as engagement policies, contact policies and arbitration. You can define next-best-action decisions and requirements which apply to the whole organization, or to specific business issues or groups.

Pega Customer Decision Hub

Next-Best-Action Designer is organized according to the high-level workflow which is needed to fully configure next best action for your organization. Follow the workflow to define the business structure for your organization, implement channel limits and constraints, define the rules that control which actions you offer to which customers, configure how actions are prioritized, and finally specify when and where you want to trigger Next-Best-Action. As you define various aspects of Next-Best-Action, Pega Customer Decision Hub builds an underlying Next-Best-Action Strategy framework based on your criteria. If later you want to modify this strategy, you can do that from Next-Best-Action Designer's intuitive, transparent interface, instead of by editing the strategy canvas directly.

You can quickly import them by using a predefined template, or configure them manually.

Importing next-best-action settings from a template

To speed up and simplify the process of implementing next-best-action, Pega provides predefined templates that contain next-best-action settings commonly used by organizations in the Communications and Financial Services industries. For more information, see Importing Next-Best-Action settings from a template.

Configuring next-best-action settings manually

You can tweak each relevant part of next-best-action settings to match the requirements of your organization. For more information, see the following topics:

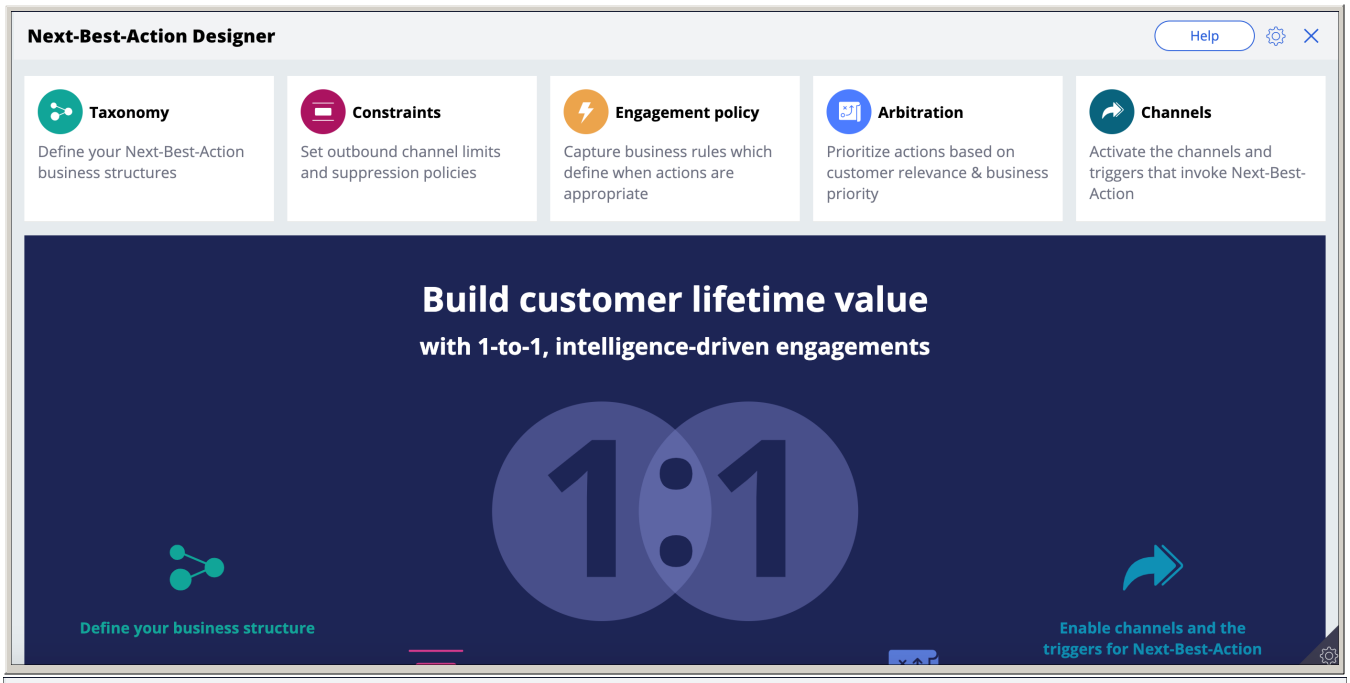
- Understanding Next-Best-Action Designer basics
- Importing Next-Best-Action settings from a template
- Manually defining the structure of business issues and groups
- Managing actions
- Setting constraints, contact policy limits, and controls
- Configuring engagement policies for business groups
- Configuring engagement policies for actions
- Prioritizing actions based on customer relevance and business priority
- Enabling channels and defining triggers
- Defining treatments for actions
- Defining customer audiences with customer segments
- Seed Lists
- Predicting customer behavior
- Managing content in real-time channels with real-time containers
- Responding to real-time customer engagement with real-time event triggers
- Increasing customer engagement with microsites
- Extending the capabilities of next best action with customized strategies
- Supplementing next-best-action with traditional outbound campaigns
- Using customer journeys to influence next-best-actions

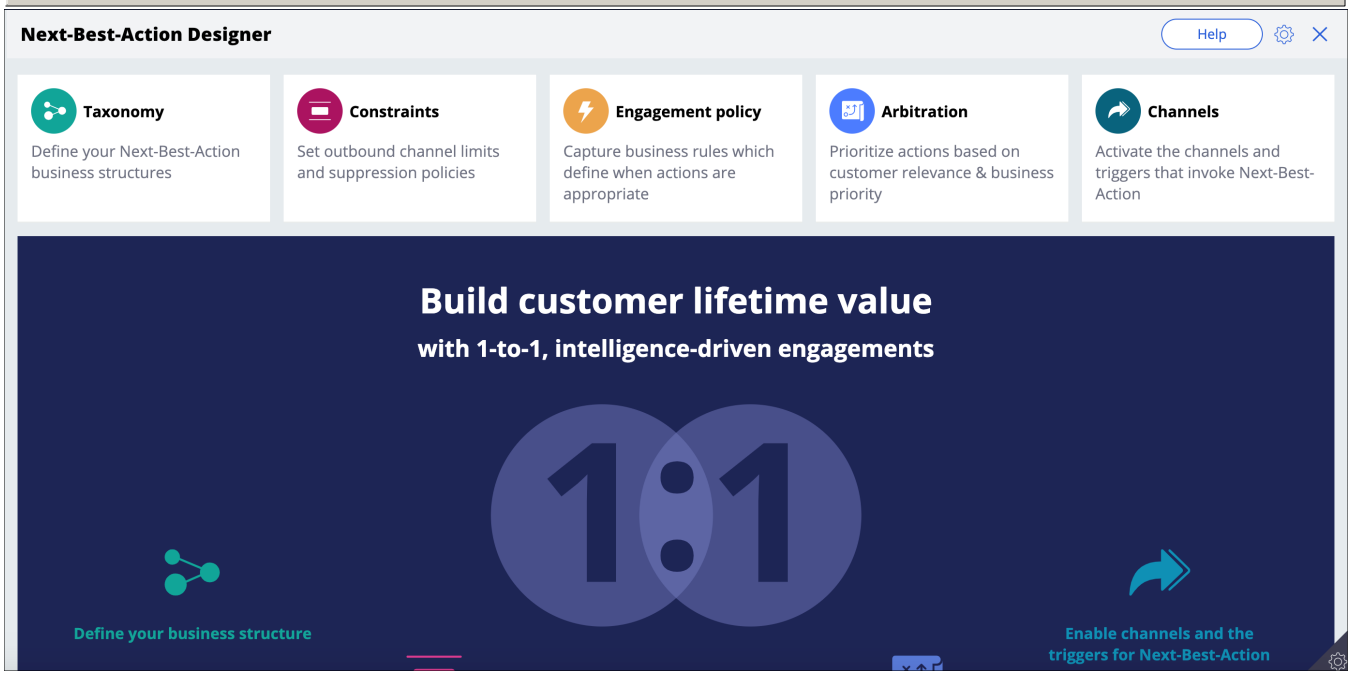
Understanding Next-Best-Action Designer basics

The Next-Best-Action Designer user interface provides an easy to use, guided way to define, manage and monitor your next-best-actions across your inbound and outbound channels.

Pega Customer Decision Hub

Next-Best-Action Designer lays out the set of steps that need to be completed to fully define your next best actions. Each tab contains a set of metadata that is used by the strategy framework to determine the personalized set of actions for customers.





- Understanding multilevel decisioning and the Context Dictionary

- Next-Best-Action Designer taxonomy
- Next-Best-Action Designer constraints
- Understanding Next-Best-Action Designer engagement policies
- Understanding Next-Best-Action Designer arbitration
- Understanding Next-Best-Action Designer channels
- Understanding the Next-Best-Action strategy framework
- Configuring Next-Best-Action Designer for Pega Customer Service
- Upgrading the Next-Best-Action Designer from legacy versions

Understanding multilevel decisioning and the Context Dictionary

Use multilevel decisioning in Pega Customer Decision Hub to control how the next best actions are presented to various levels of customer contacts, such as a head of household, account owner, or device owner.

These levels are referred to as customer contexts and are treated as different audiences that must be communicated with in different ways. For example, you can offer the account owner a family data plan that applies to the whole account, and also send out new phone offers to individual household members that qualify for them.
Pega Customer Decision Hub

Understanding the relationship between context entities, associated data, and others

Refer to the following section for an explanation of several terms used to describe various aspects of multilevel context creation, and the relationships between them:

Context entity
: Any entity on which next-best-action decisions can be made. Depending on the industry, this entity can represent, for example, an account, a household, or a policy membership.

Primary context
: This context represents the individual to whom you communicate next-best-action decisions. For example, the primary context for an account is the account owner, and the primary context for a policy is a policy member. After a decision is made for the context entity, the primary context receives the resulting next best action.

Associated data
: A data entity directly related to either a context entity, or another associated data entity. For example, data about a customer's product holdings can be associated with the customer context entity, and data about product details can be associated with data about a customer's product holdings. Associated data provides additional information about the context that the Next-Best-Action strategy can then use to make decisions.

Context Dictionary
: The Context Dictionary enables you to define context entities and primary contexts, as well as the relationship between them. You can also use it to manage the associations between various contexts and associated data. You can access the Context Dictionary through the **Settings Context Dictionary** menu in App Studio.

Supported context definitions

The relationship between context entities and primary contexts must have a parent-child structure and follow one of the following patterns:

Definition	Example
• Context entity ◦ Primary context	• Account ◦ Subscriber
• Primary context ◦ Context entity	• Customer ◦ Account
• Context entity ◦ Context entity ▪ Primary context	• Account ◦ Subscription ▪ Owner
• Context entity ◦ Primary context ▪ Context entity	• Account ◦ Subscriber ▪ Device

Other configurations, such as having multiple sibling entities under one parent entity, are not supported. If you have multiple brands, organize the customer contexts in such a way as to cover all the brands in one parent-child structure.

Configuring associated data

Pega Customer Decision Hub supports associating the customer class with other entities. For example, each customer record can have associated purchase data information. After creating the customer contexts in the Context Dictionary, you can create and configure any associated entities.

You can use the Customer Profile Designer to configure data associations for each customer context. The intuitive Profile Designer interface guides you through the process of selecting the data set or data flow that you want to use as a data source. If the data source contains event data, you can create an aggregated summary of the data directly on the Summaries tab of Customer Profile Designer.

For more information, see Adding associated static data in the Customer Profile Designer.

Next-Best-Action Designer taxonomy

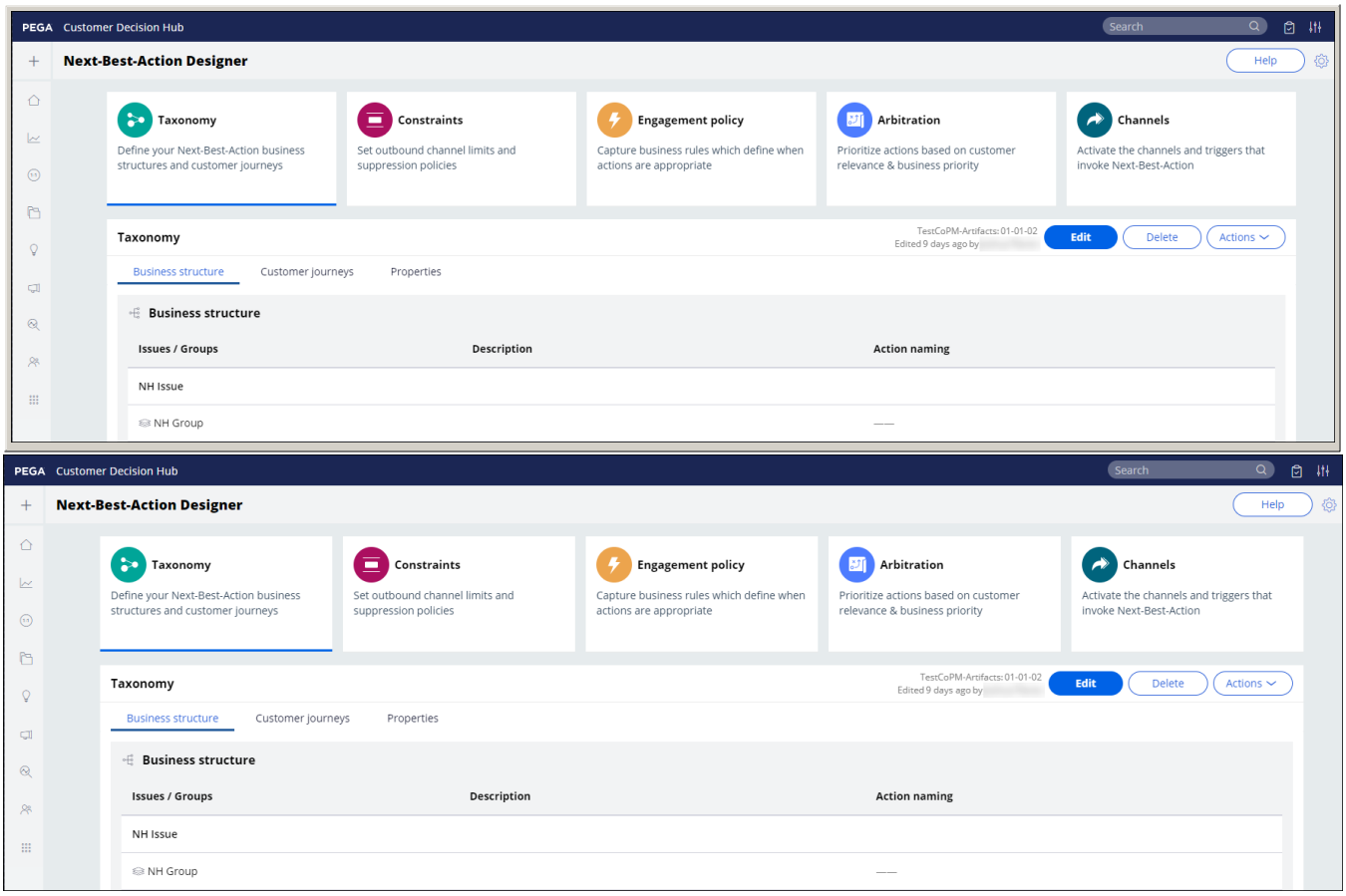
Use the **Taxonomy** tab to define the traditional issue and group hierarchy for decisioning, create and manage the customer journeys, as well as create and manage strategy result properties used in used in the strategy framework and your next-best-actions.

Pega Customer Decision Hub

Business structure

The **Business structure** tab displays the issues and groups defined in your application. All issues and groups should be defined here instead of on the **Proposition Management** landing page.

A business issue is the purpose behind the actions you offer to your customers. For example, actions which started with the purpose of retaining existing customers should be grouped under the business issue of Retention. Actions started with the purpose of acquiring new customers belong to the business issue of Acquisition.



Together, issues and groups form the organizing structure for your customer interactions. Each action you prepare for your customers is associated with an issue and a group. Because of that, you must always define the hierarchy before creating actions.

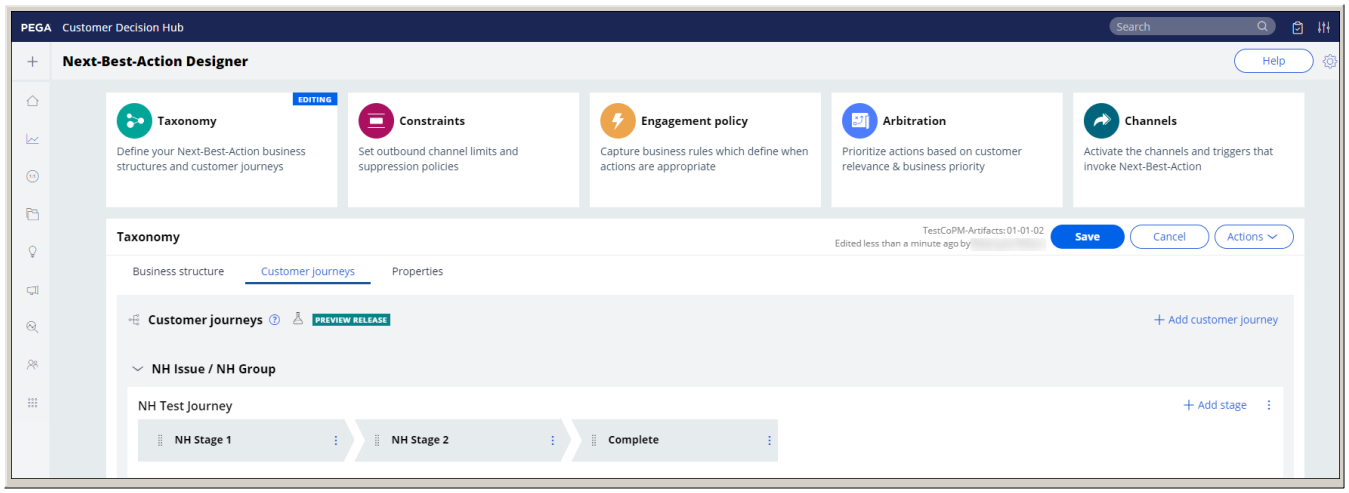
Business groups organize customer actions into categories. For example, as part of the business issue of Acquisition, you can create groups for products like Credit Cards, Mortgages, or Personal Loans, with the intention of offering these to potential new customers.

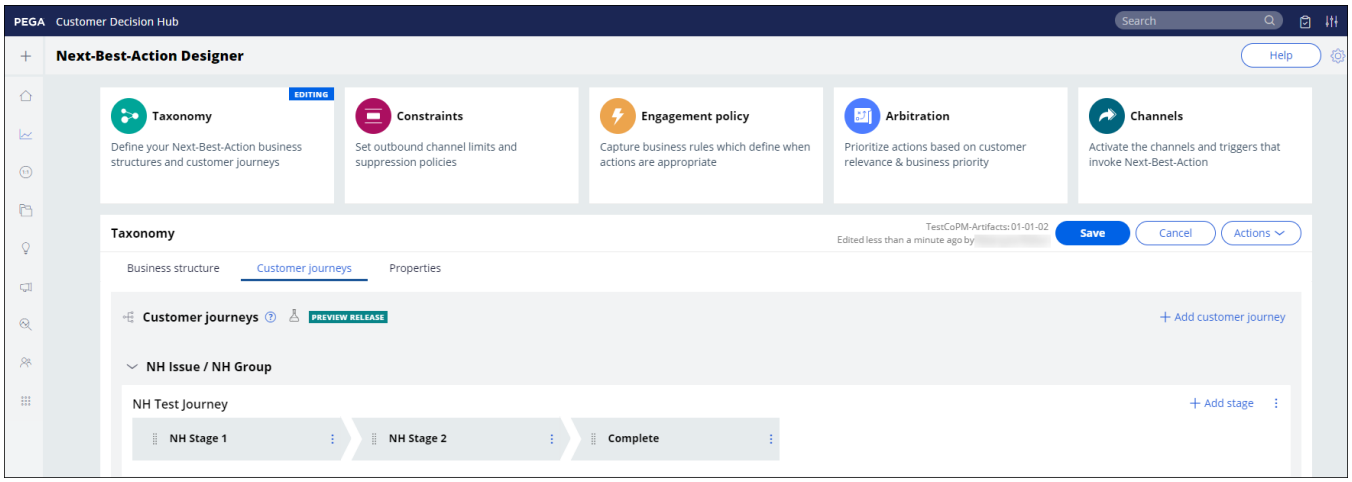
In addition to defining the issues and groups, you can add a short and long description for each issue and group that helps explain the intent and business purpose. For more information, see Manually defining the structure of business issues and groups.

Tip: By default, Pega Customer Decision Hub refers to the recommendations that come out of the strategy framework as actions. If you would like to refer to these actions differently for a specific group of actions, you can do so under each group. For example, you can refer to actions in the **Acquisition > Cards** group as Nudges, and to actions in the **Acquisition > Mortgages** group as Promotions.

Customer journeys

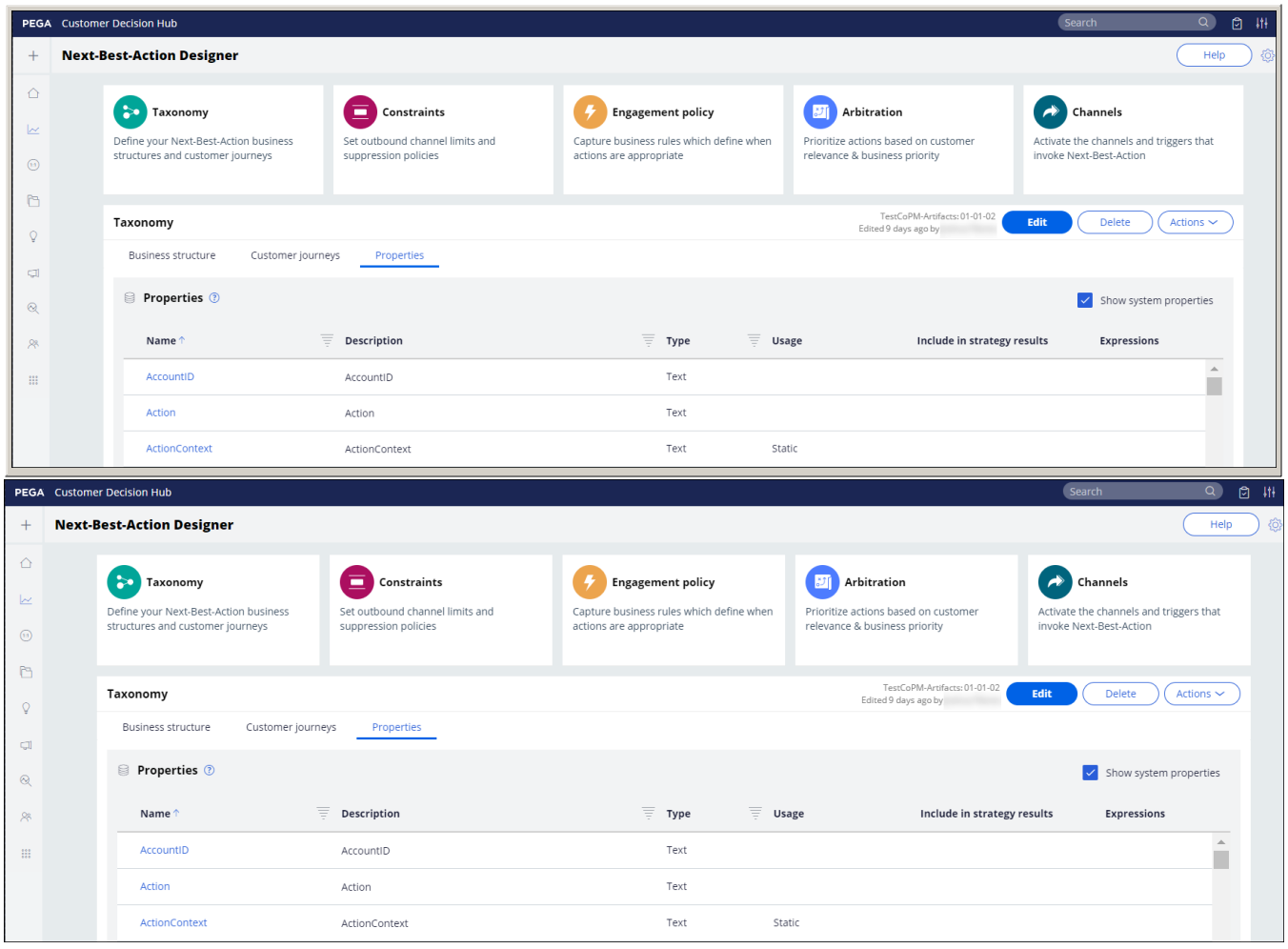
The **Customer journeys** tab displays the customer journeys defined in your application. You can use it to add new journeys, edit existing ones, or define entry conditions for each journey stage.





Properties

The **Properties** tab displays the strategy result and action properties available in your application, including both default system properties, and custom properties that you may define for your outbound treatments or real-time containers. You can define new properties directly on this page, without needing to switch to Dev Studio.

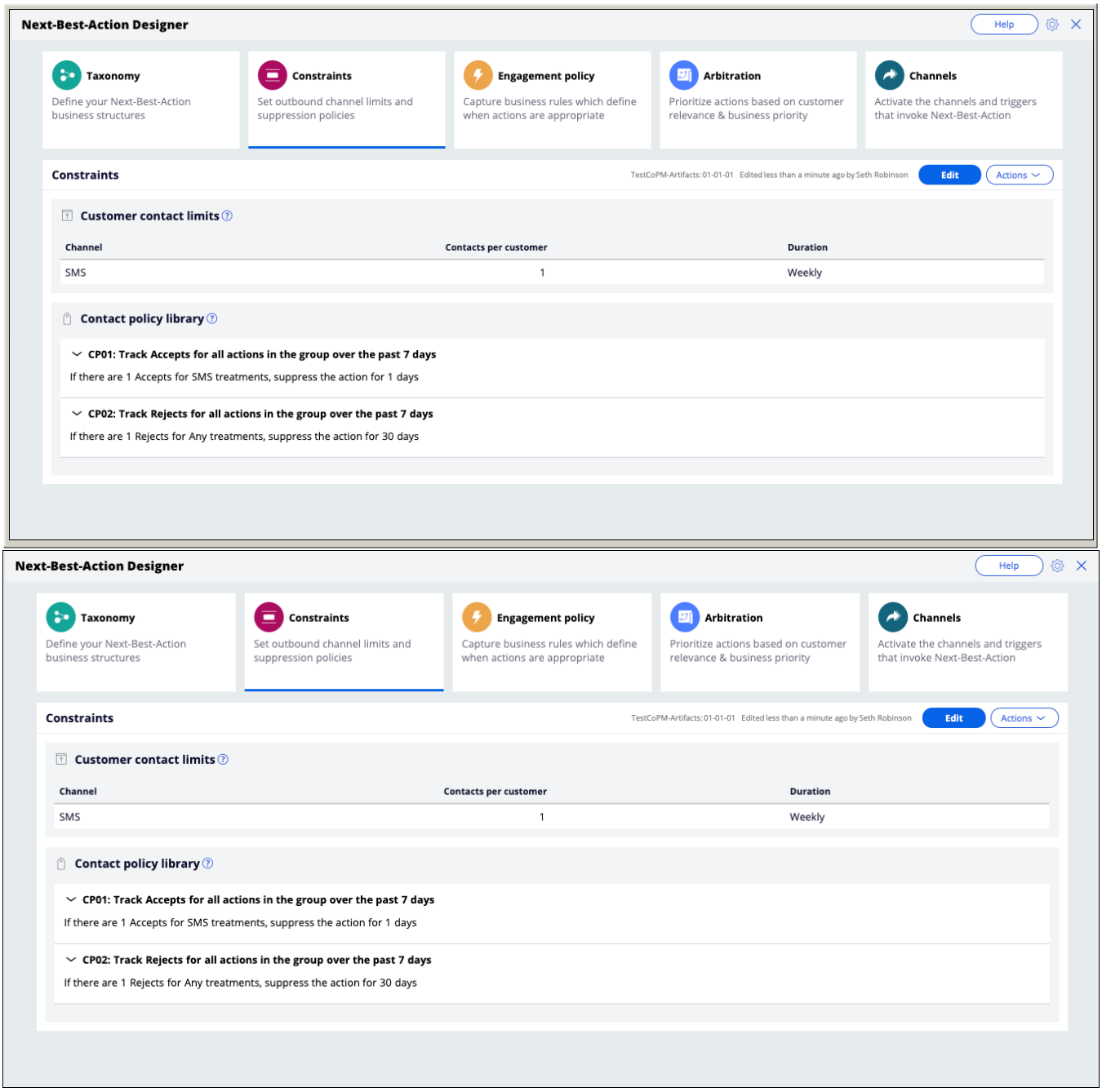


To add a new property, click **Edit**, and then click **Add property**. For more information about configuring properties, see Properties.

Next-Best-Action Designer constraints

Use constraints to ensure you do not over-communicate to customers (for example, one email per week, two SMS per month), or over-expose customers to a specific action or group of actions.

Pega Customer Decision Hub



Next-Best-Action Designer
Help
Taxonomy
Define your Next-Best-Action business structures
Constraints
Set outbound channel limits and suppression policies
Engagement policy
Capture business rules which define when actions are appropriate
Arbitration
Prioritize actions based on customer relevance & business priority
Channels
Activate the channels and triggers that invoke Next-Best-Action
Constraints
TestCoPM-Artifacts: 01-01-01 Edited less than a minute ago by Seth Robinson
Edit
Actions
Customer contact limits
Channel
Contacts per customer
Duration
SMS
1
Weekly
Contact policy library
CP01: Track Accepts for all actions in the group over the past 7 days
If there are 1 Accepts for SMS treatments, suppress the action for 1 days
CP02: Track Rejects for all actions in the group over the past 7 days
If there are 1 Rejects for Any treatments, suppress the action for 30 days

Customer contact limits
: Limit the number of interactions which a customer can receive over a given period of time on a specific channel. For example, you can decide that you do not want your customers to receive more than two emails per week. When in Edit mode, you can open the contact limit rule directly and make any required changes. This rule is automatically saved into your implementation ruleset as part of the Context Dictionary generation process.

Contact policy library
: Define suppressions, that is, automatically put an action on hold after a specific number of outcomes are recorded for some or all channels. For example, an action can be suppressed for a customer for seven days after the customer has seen an ad for that action for five times. Suppressing or pausing an action prevents over-saturation by limiting the number of times that a customer is exposed to the same action. This limit is enforced by the strategy framework and stored in a decision data record (DDR). Contact policies are associated with an action or group and enforced by the strategy framework. The check as to whether a customer has exceeded a specific limit is handled as part of the *ProcessResponse* mechanism.

Understanding Next-Best-Action Designer engagement policies

Engagement policies allow users to define when specific actions or groups of actions are appropriate for customers. There are four types of engagement policies:

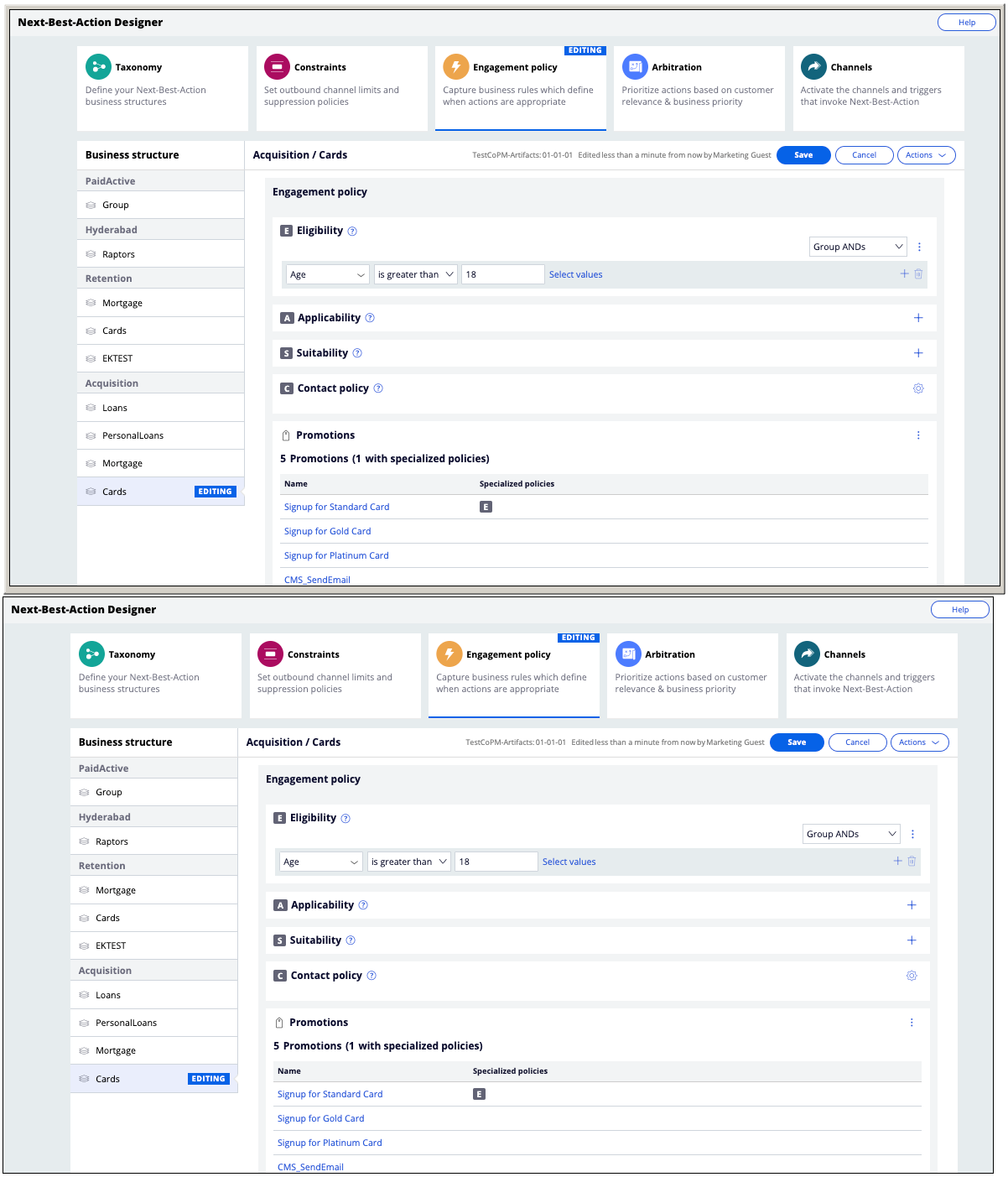
Pega Customer Decision Hub

Eligibility
: The criteria which the customer must fulfill in order to qualify for the action or group of actions. For example, an action may only be available for customers over a specific age or living in a specific geographic location.

Applicability
: The criteria which determine if this action or group of actions is appropriate at this point in time. For example, a discount on a specific phone model may not be relevant for a customer who already owns a phone of this type.

Suitability
: The criteria which must be true in order for the action or group of actions to be deemed in the best interest of the customer. For example, a new credit card offer may not be appropriate for a customer who is likely to default on payments.

Contact Policy
: The policies which determine when an action or group of actions should be suppressed and for how long. For example, you can suppress an action after a specific number of promotional messages have been sent to customers.



The **Engagement policy** tab generates a number of artifacts, including a proposition filter for each policy type and a group level strategy. The strategy imports the actions, ensures that the action is active, and then applies a series of proposition filters. The output of the strategy is a set of eligible and appropriate actions to be evaluated by the strategy framework.

Understanding Next-Best-Action Designer arbitration

Arbitration determines how the strategy framework prioritizes the list of eligible and appropriate actions that come out of each group. The configuration and metadata for your arbitration options are stored in several decision data rules (DDR) which are used by the strategy framework.

Pega Customer Decision Hub

Propensity

The Next-Best-Action strategy framework includes adaptive models that learn at the action as well as the treatment level. These models apply the principles of outcome optimization at the treatment level and can be extended if you wish to apply advanced techniques or challenge the default models.

The propensity used in arbitration can be applied from the treatment level or at the action level. It is recommended to use treatment-level propensity, since it provides the best treatment on the best channel for a particular customer.

Note: The models always learn, even if you decide not to apply analytics to your prioritization formula.

Context weighting

Real-time contextual data is an important part of making highly relevant recommendations. Context weighting allows you to assign weighting for a specific context value to all actions within an issue or a group.

The contextual weighting definitions are stored in a decision data rule (DDR), which is managed by the Next-Best-Action Designer and used by the strategy framework.

Business value

You can specify the value of an action in the definition of that action. If more sophisticated calculations are needed, the extension strategy CalculateValueExtension can be

customized to your specific requirements.

Business levers

To highlight or push a particular action or group of actions, specify a weight on the individual action or add a business purpose weighting and selecting an issue only or group and then the weighting value.

These levers can be toggled on or off, which determines whether or not these weightings are applied to the final prioritization formula.

Model maturity

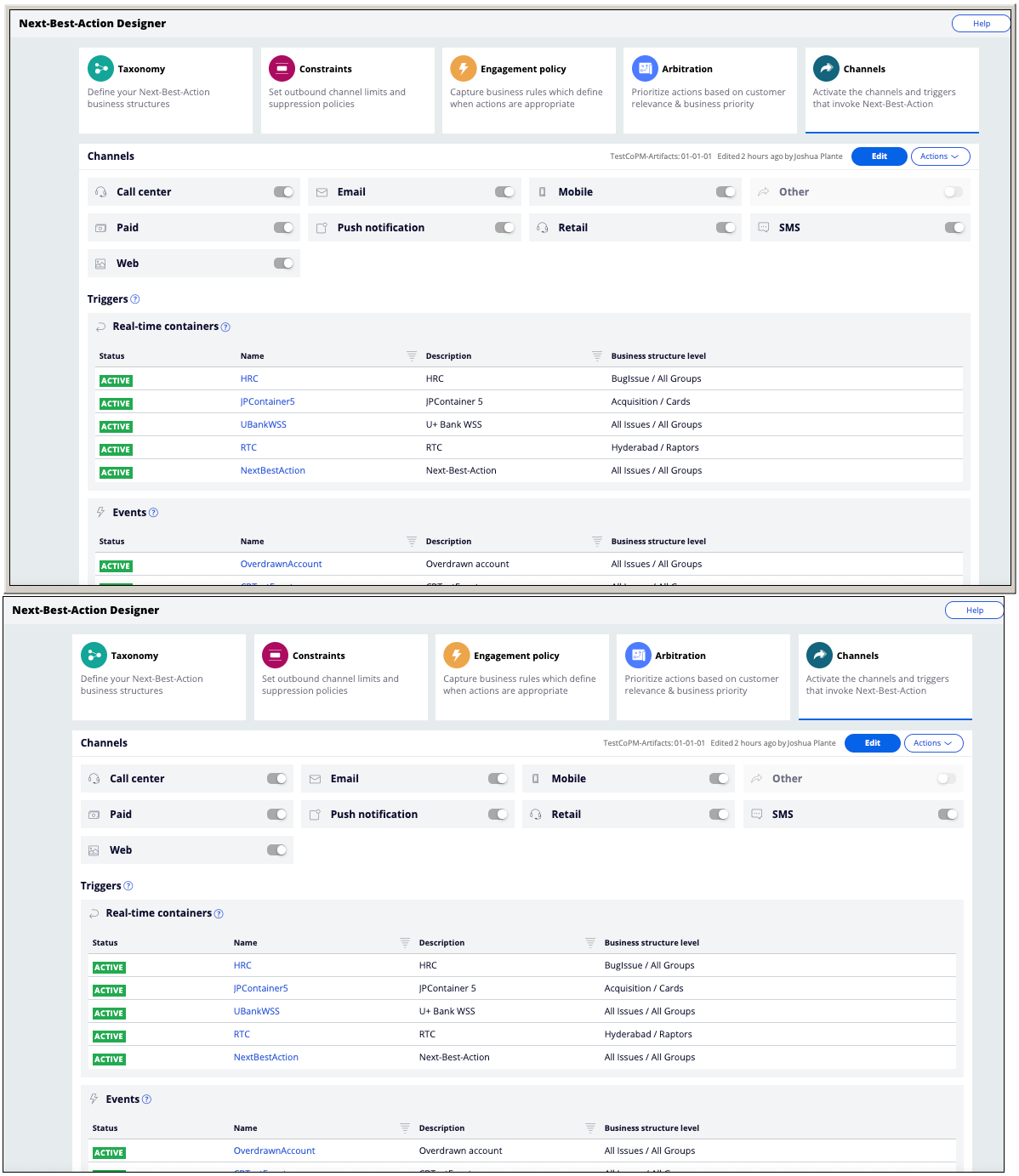
For newly introduced outbound actions, AI models may not yet know which customers to target. To minimize the possibility of customers receiving actions that are not relevant, only 2% of the customer population are initially targeted with new actions. As the model collects the resulting learnings and matures, the targeted percentage increases until it reaches 100% for fully mature models. A model is considered mature after it has received 200 or more positive responses.

To prevent multiple new actions from having the same starting propensity, Pega Customer Decision Hub uses Thompson Sampling instead of the standard propensity smoothing formula. The propensity is initially set to a random value between 0 and 1. As the model matures, this quickly converges to the model propensity.

Understanding Next-Best-Action Designer channels

Next-Best-Action Designer comes with a full set of channels and triggers which are part of the strategy framework.

Pega Customer Decision Hub



Channels

Channels can be toggled on or off. If a channel is toggled off, its status is stored in a decision data rule (DDR) and the related part of the strategy framework is turned off. Setting the toggle to on reactivates the related part of the strategy framework.

To add additional channels, edit the *ChannelSettings* DDR, and then implement these channels in the channel extension strategies within the framework. These custom channels appear under **Other** on the Treatment landing page. When creating a new Other treatment, you can specify the custom channel, and then pick the direction

(inbound or outbound). On an Action rule, to use a custom channel, add the Other channel, and then pick the appropriate treatment.

Triggers

The Next-Best-Action strategy framework can be invoked by real-time channels through our Container REST API and Kafka event stream, as well as on a scheduled basis.

As you define one of these triggers, Next-Best-Action Designer generates a dataflow with a strategy based on the specified business structure. These generated dataflows can be seen on the Dataflows landing page if the **View system generated flows** setting is enabled.

Understanding the Next-Best-Action strategy framework

The Next-Best-Action strategy framework is shipped with Pega Customer Decision Hub and provides the best practice for implementing enterprise-level next-best-action.

Pega Customer Decision Hub
The strategy framework is applied to the top (all issues and groups), issue or group level of the business hierarchy after you define a trigger in the **Channels** tab of Next-Best-Action Designer. Each trigger generates a strategy which references the correct level of the business structure, imports the actions, and applies the Eligibility, Applicability, and Suitability rules. The strategy then passes these results to the strategy framework for processing. A trigger may be a real-time container, an event, or a recurring (outbound) schedule.**Note:**

Starting in version 8.6, Pega Customer Decision Hub introduces the globally optimized strategy (GOS), which greatly increases the performance of strategy execution. GOS decreases the run time and memory usage of strategy execution in batch and real-time scenarios by wrapping your top-level (trigger or entry) strategy within a globally optimized strategy.

This capability is built into Next-Best-Action Designer and does not require any additional configuration. Triggers are automatically optimized as soon as you edit and save the **Channels** tab. You can also use GOS with your custom strategies. For more information, see Understanding globally optimized strategies.

- Next-Best-Action Designer metadata
- Trigger strategy
- NBA Strategy framework
- Testing the Next-Best-Action strategy framework

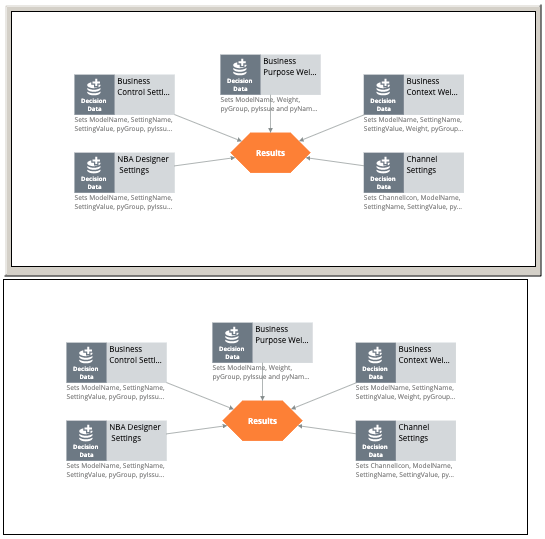
Next-Best-Action Designer metadata

The configuration information managed by the Next-Best-Action Designer interface is stored in a set of decision data rules (DDRs). The strategy framework references the *RealtimeControl* sub-strategy and pulls in the appropriate DDR based on the strategy.

Pega Customer Decision Hub
These DDRs should not be updated directly because they are managed by Next-Best-Action Designer. If you use the Revision Management functionality, these DDR rules are automatically packaged as part of the change request.

Realtime Controls strategy

When referencing this sub-strategy, an individual DDR may be specified as a single component within the strategy to avoid retrieval of unneeded controls.



Many of the settings are in the form of key / value pairs using the properties below:

SettingName
: The key, or name of the setting.

SettingValue
: The value of the setting.

You can obtain individual setting values by using a **Filter** shape based on the *SettingName* property.

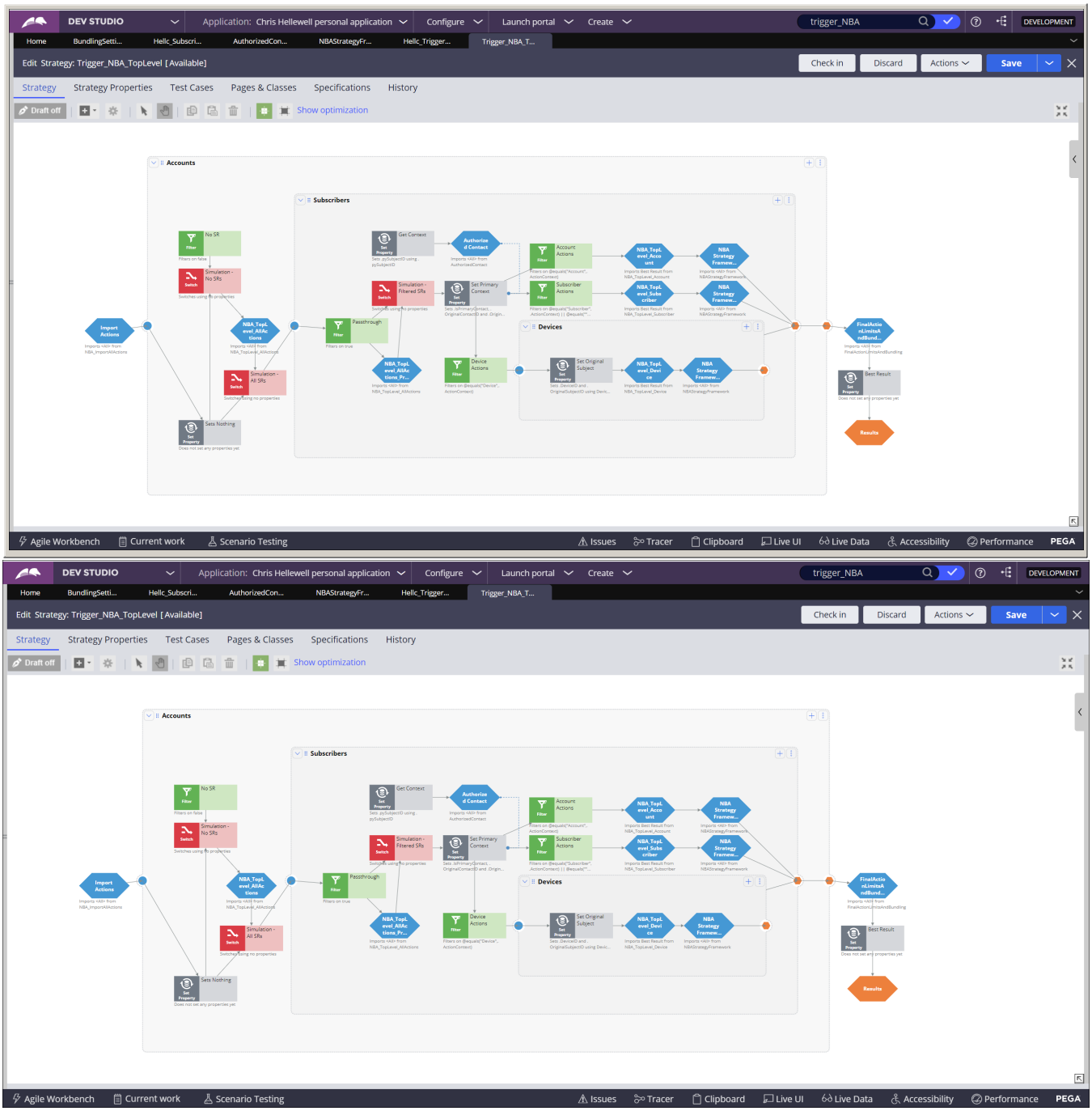
Trigger strategy

A trigger strategy is generated after you define a trigger within Next-Best-Action Designer and specify the business structure to apply. You cannot modify it directly. This strategy is referenced by a data flow, which is also generated and managed by the system.

Pega Customer Decision Hub
A separate trigger strategy is generated for each business structure specified as part of a trigger configuration and the strategies are named based on the business structure as shown in the following table:

Issue	Group	Trigger strategy name	Example
All issues	All groups	Trigger_NBA_TopLevel	
<Issue>	All groups	Trigger_H_NBA_*<Issue>*	Trigger_H_NBA_Acquisition
<Issue>	*<Group>*	Trigger_NBA_*<Issue>*_*<Group>*	Trigger_H_NBA_Acquisition_Bundle

The layout of the strategy depends on the configuration of the context dictionary. The following example shows an implementation of multiple Subscribers within an Account, where each Subscriber may have multiple Devices, for example, for a Communications application. Note that this example is for all issues and all groups.



Note: Three context levels is the most complex configuration available for the Context Dictionary. Most NBA Trigger strategies are simpler than the above example.

The high-level logic flow is as follows:

- Import actions
- Evaluate All Actions engagement policies for all action contexts
- Identify Authorized Contacts
- Split actions into a separate stream for each action context
- Evaluate the context level engagement policies
- Execute the NBA Strategy Framework separately for each context
- Merge all context streams
- Apply final action limits and bundling options

The arrangement of Switch rules around the All Actions engagement policy sub-strategies immediately after the action import is only required if the primary context is not the top-level context. Its purpose is to execute the engagement policy within the primary context if the Trigger strategy is being executed as part of a simulation, otherwise it is executed in the top-level context.

The **Set Primary Contact** shape sets the following properties:

- *IsPrimaryContact* - set to true if the contact is identified as an Authorized Contact
- *OriginalSubjectID* - set to pySubjectID
- *OriginalContactID* - set to pySubjectID

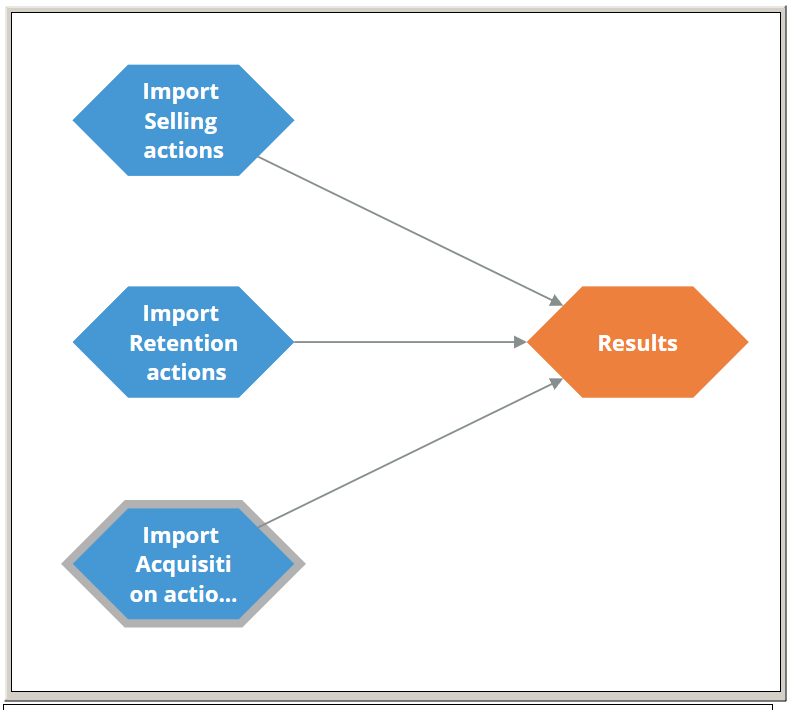
- Import Actions strategy
- Authorized Contact strategy
- Engagement Policy strategies

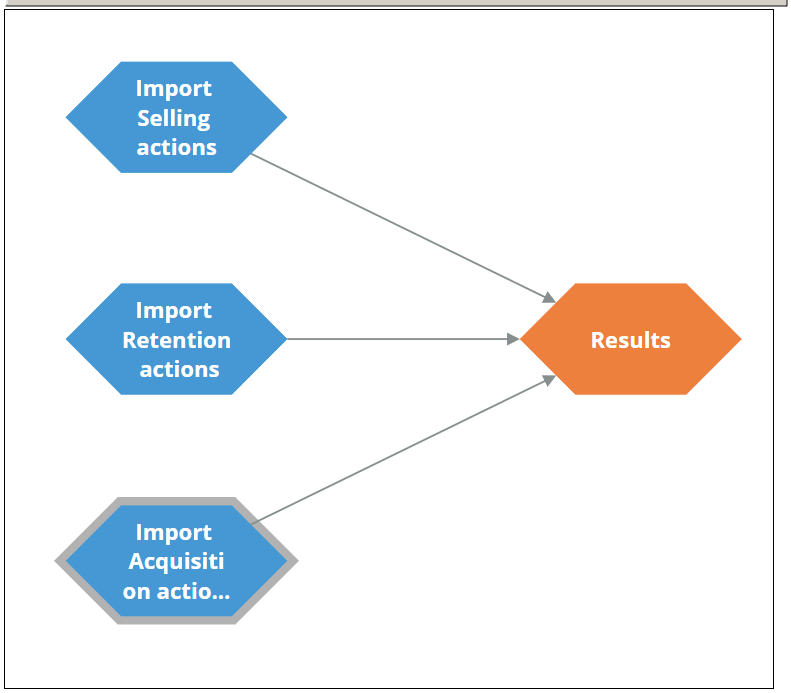
Import Actions strategy

The actual strategy used for importing actions depends on the trigger strategy as described in the table below.

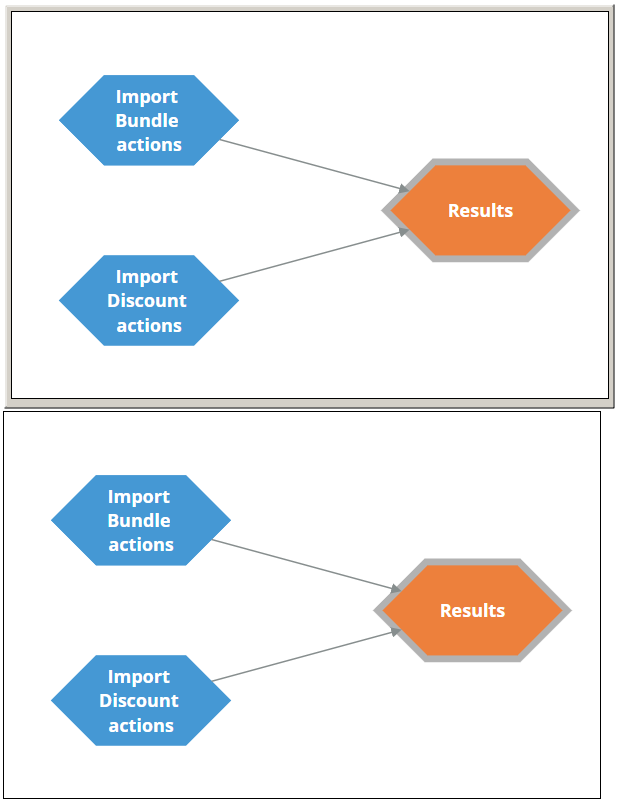
Trigger Strategy Name	Action Import Strategy Name	Purpose
Trigger_NBA_TopLevel	*NBA_Import_AllActions*	Calls the *NBA_Import_<Issue>* strategies for each issue in the taxonomy
Trigger_H_NBA_<Issue>	*NBA_Import_<Issue>*	Calls the *NBA_Import_<Issue>_<Group>* strategies for each group within the issue
Trigger_NBA_<Issue>_<Group>	*NBA_Import_<Issue>_<Group>*	Includes the **Proposition Data** shape for the individual issue and group required

The Import Actions strategy combines references to the action import sub-strategies for each business issue in the next-best-action taxonomy required by the trigger; this will be either a single issue, or all issues as shown in the above table. It is generated by Next-Best-Action Designer and is not modifiable. The example below is for a configuration with three business issues: Selling, Retention and Acquisition.

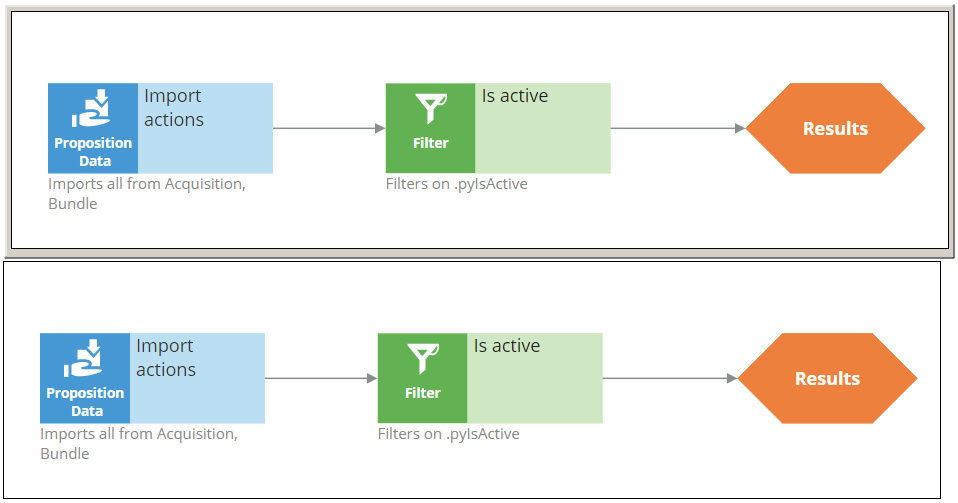




Each of these business issue sub-strategies combines references to the action import sub-strategies for each group in the next-best-action taxonomy required by the trigger; this will be either a single group, or all groups within the issue. They are generated by Next-Best-Action Designer and are not modifiable. The example below contains two groups: Bundle and Discount.

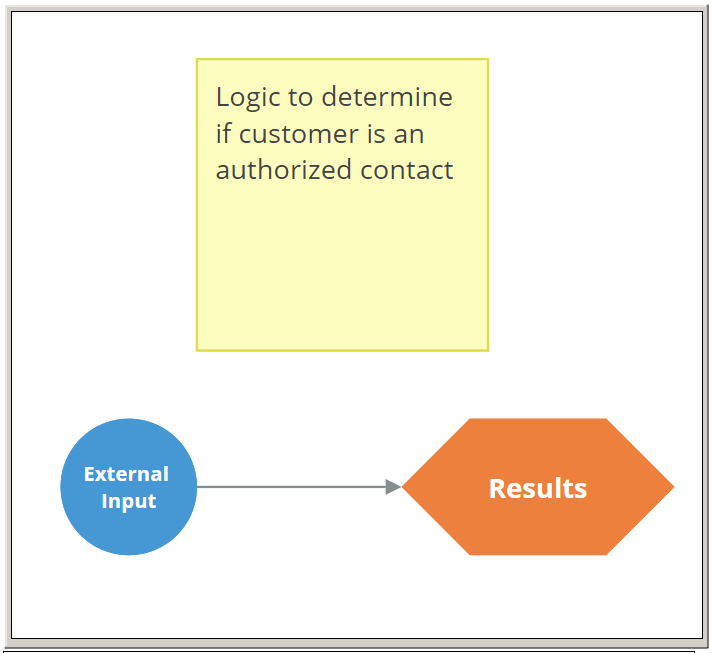


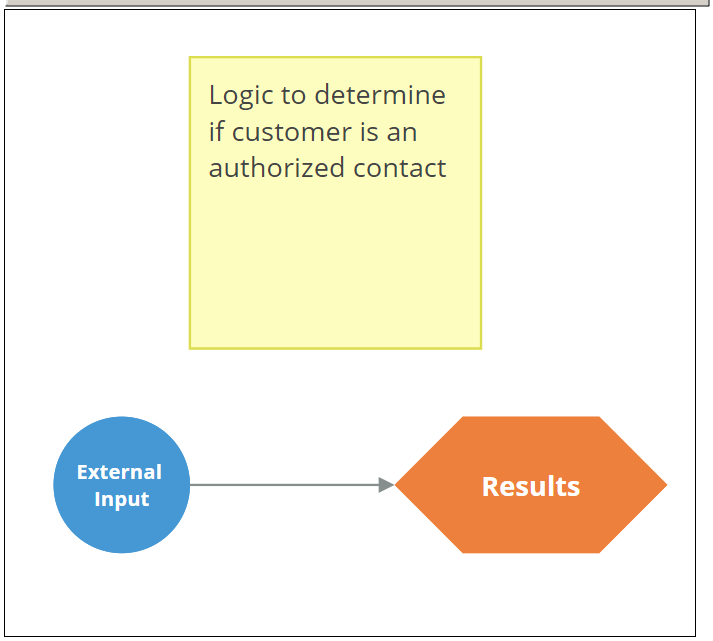
Each of the group sub-strategies contains a single Proposition Data shape that imports all actions for the Issue / Group combination, and then uses a Filter shape to select only active propositions (actions). They are generated by Next-Best-Action Designer and are not modifiable; the example shown below is for the Acquisition / Bundle combination.



Authorized Contact strategy

This strategy is provided as an empty canvas as shown below. In an implementation where there is a concept of an authorized or primary contact where some actions may only be delivered to such a contact (for example, head of household or primary account holder), it should be modified to include logic to identify such contacts.





Engagement Policy strategies

Engagement policies are implemented through a series of Proposition Filters, one for each of the Eligibility, Applicability and Suitability conditions - in that order - contained in a sub-strategy.

Pega Customer Decision Hub

Separate sets of Proposition Filters (and hence sub-strategies) are required for each issue and group combination in the taxonomy, and for each context level within that combination. In addition, a further set of All Actions Proposition Filters may be optionally defined that apply to actions from all contexts.

An additional set of Engagement Policies may be defined at the top of the taxonomy hierarchy for all issues and all groups, and these are applied to all actions in any issue and group. Just like the individual issue and group-level policies, a separate policy may be defined for each context, and each of these has a separate proposition filter for each of the Eligibility, Applicability and Suitability conditions.

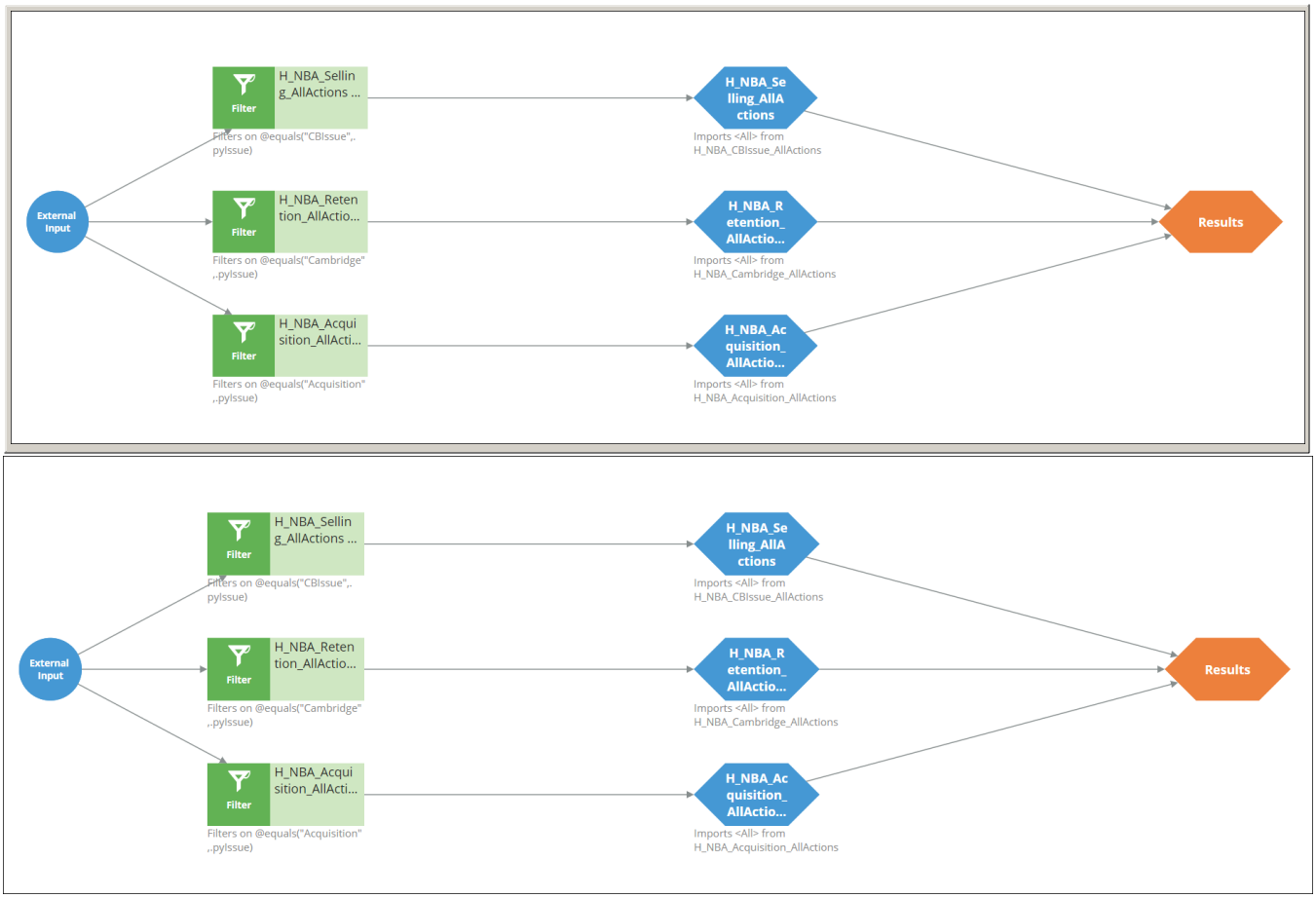
All Actions Engagement Policies

The All Actions strategy is executed first, and the strategy used depends on the trigger strategy as described in the table below.

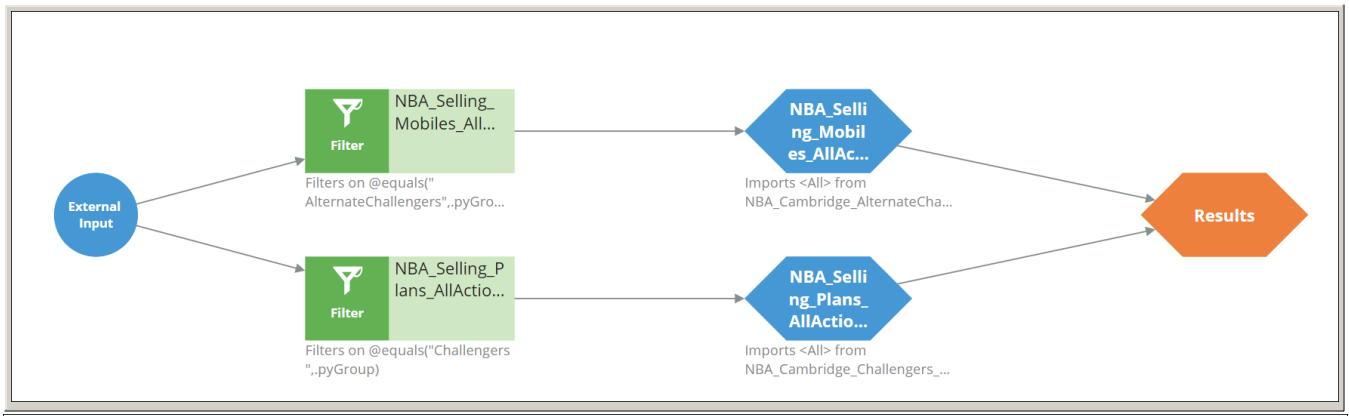
Trigger Strategy Name	All Actions Strategy	Purpose
Trigger_NBA_TopLevel	*NBA_TopLevel_AllActions*	Calls the *H_NBA_<Issue>_AllActions* strategies for each issue in the taxonomy
Trigger_H_NBA_<Issue>	*H_NBA_<Issue>_AllActions*	Calls the *NBA_<Issue>_<Group>_AllActions* strategies for each group within the issue
Trigger_NBA_<Issue>_<Group>	*NBA_<Issue>_<Group>_AllActions*	Calls the *NBA_<Issue>_<Group>_AllActions* strategies for each group within the issue

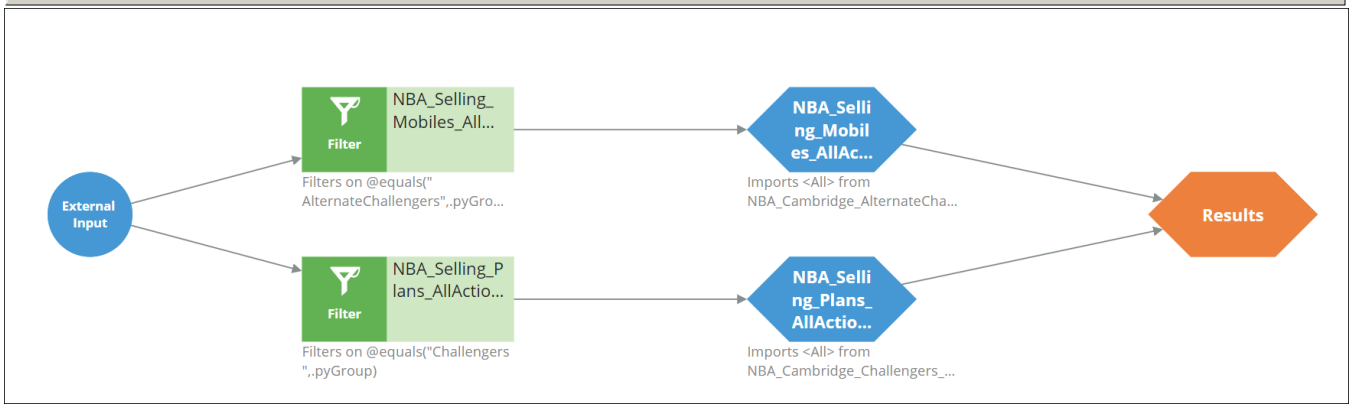
These strategies are generated by Next-Best-Action Designer and are not modifiable.

The following figure shows an example of the top-level All Actions strategy for issues Selling, Retention and Acquisition:

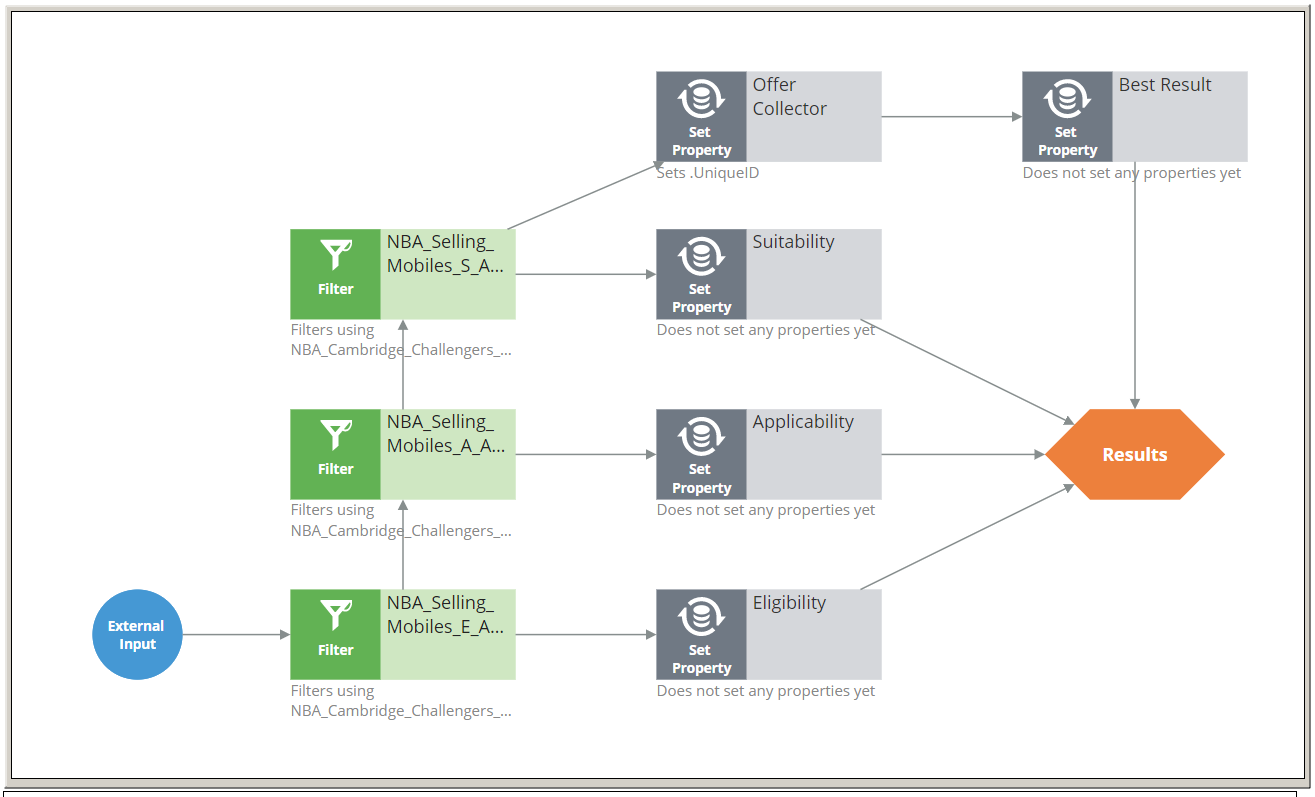


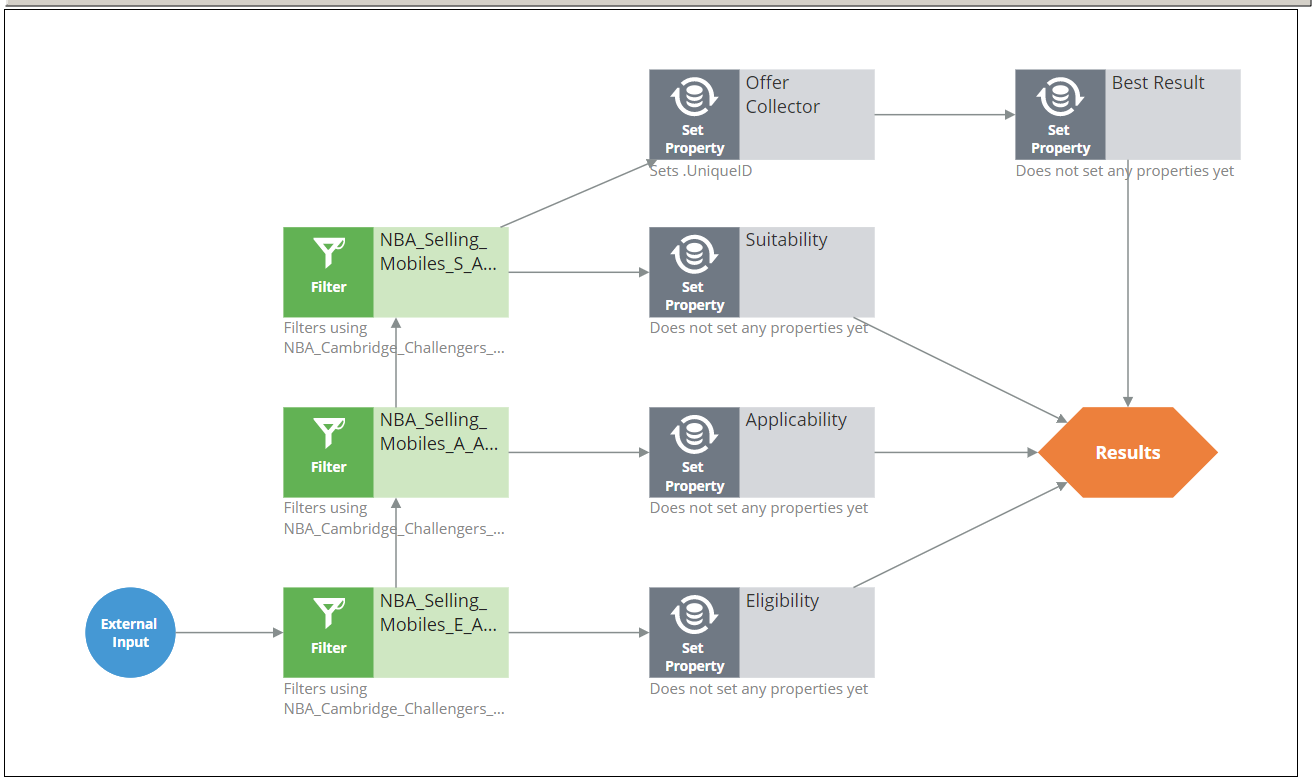
Next is an example of an issue-level All Actions strategy for the Selling issue with groups Mobiles and Plans:





Finally, the following example shows the group-level strategy containing the Eligibility, Applicability and Suitability proposition filters for the Selling issue, and Mobile group:

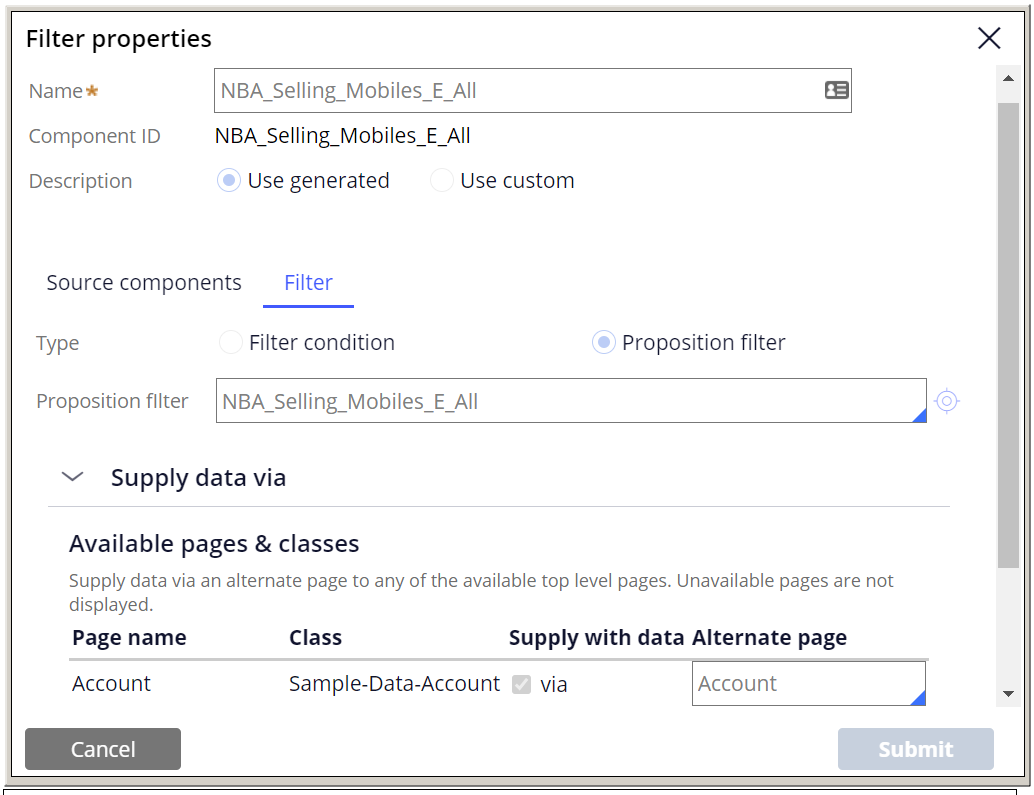


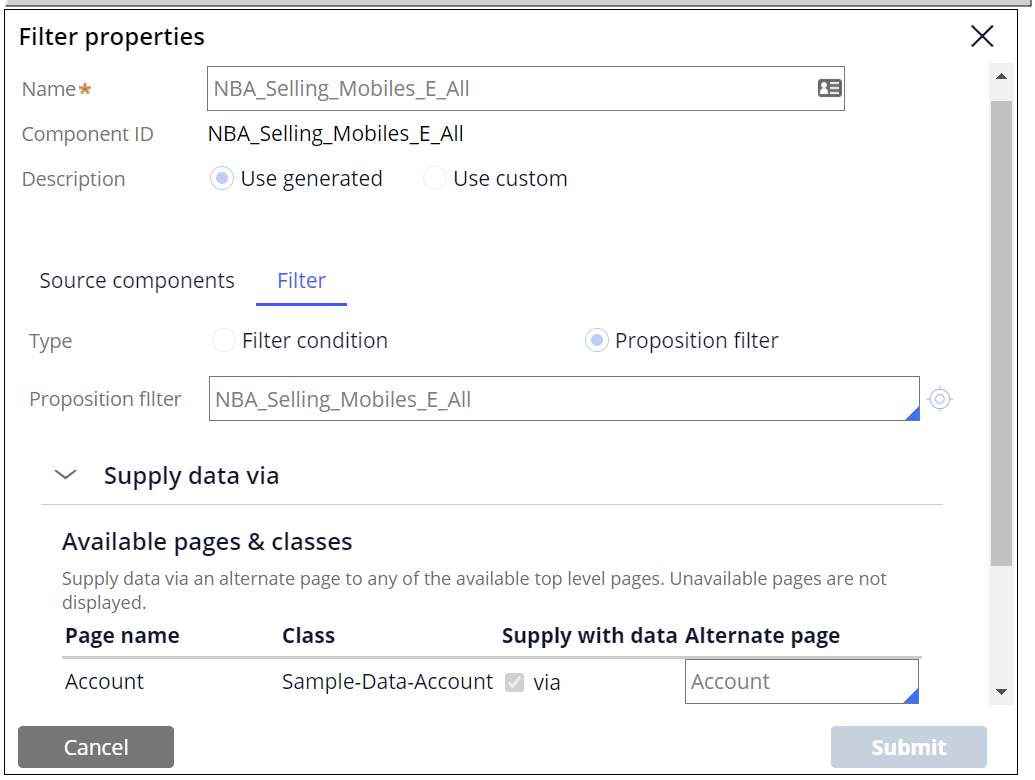


The Offer Collector shape sets the *UniqueID* property for internal use, and there are four exit points:

- Best Result is used by the Next-Best-Action Strategy Framework for actions that pass all conditions.
- Eligibility, Applicability and Suitability are used by the Simulation feature to show how actions successively drop out as they progress through the filter stages.

Since these All Actions proposition filters are all executed at the top context level, they need access to the data page from that context which is defined in the **Pages & Classes** tab of the proposition filter definition. The page name is the same as the context name and is passed to the proposition filter via the **Available pages & classes** section of the Filter properties shape, as shown in the example below for the Account context:





Context-level Engagement Policies

The individual context level strategies are executed next and again will depend on the trigger strategy as described in the table below. Note that a separate strategy is executed for each context in the context dictionary.

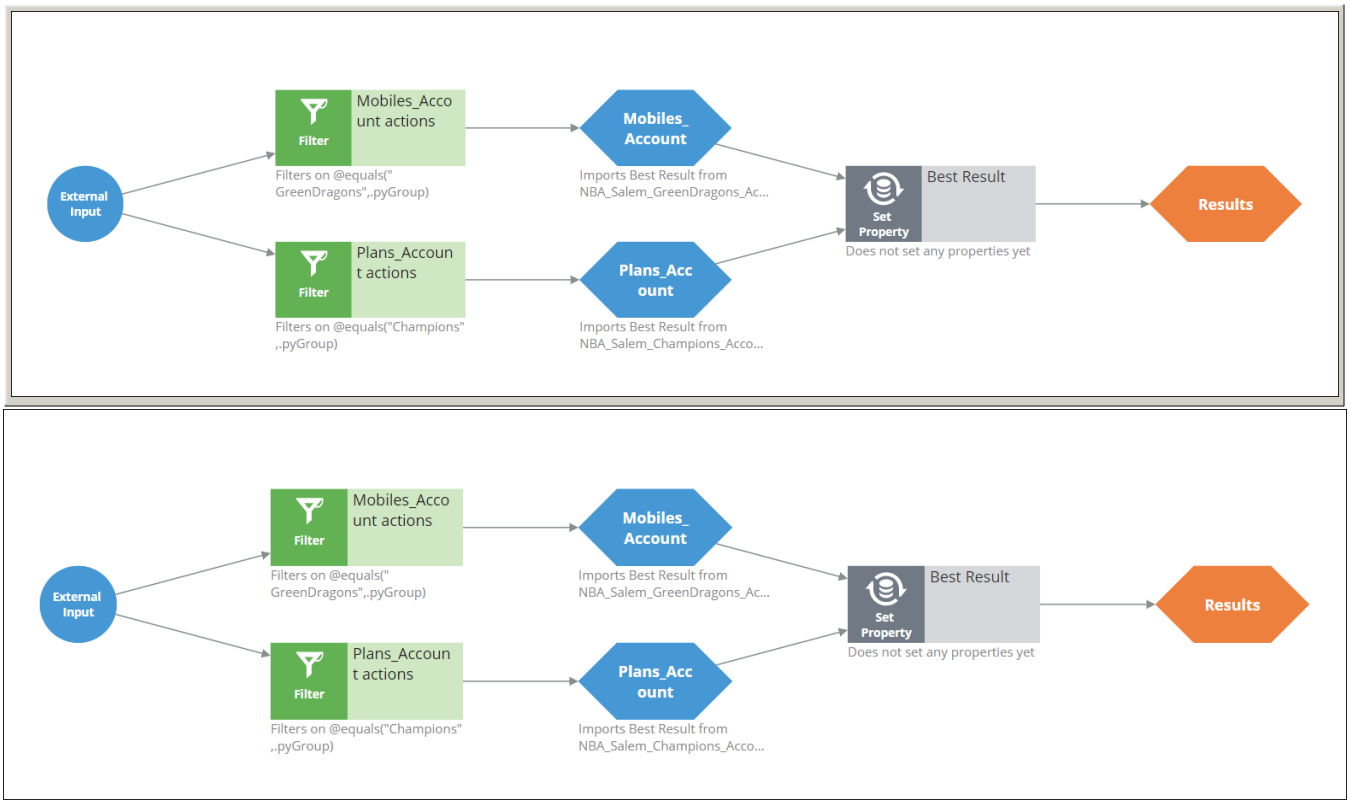
Trigger Strategy Name	All Actions Strategy	Purpose
Trigger_NBA_TopLevel	*NBA_TopLevel_<context>*	Calls the *H_NBA_<Issue>_<context>* strategies for each issue in the taxonomy
Trigger_H_NBA_<Issue>	*H_NBA_<Issue>_<context>*	Calls the *NBA_<Issue>_<Group>_<context>* strategies for each group within the issue
Trigger_NBA_<Issue>_<Group>	*NBA_<Issue>_<Group>_<context>*	Includes the context level Proposition Filters for Eligibility, Applicability and Suitability for the individual issue and group required. Also calls *NBA_AllIssue_AllGroups_<Context>* for each of Eligibility, Applicability and Suitability prior to their individual Proposition Filters

These strategies are generated by Next-Best-Action Designer and are not modifiable.

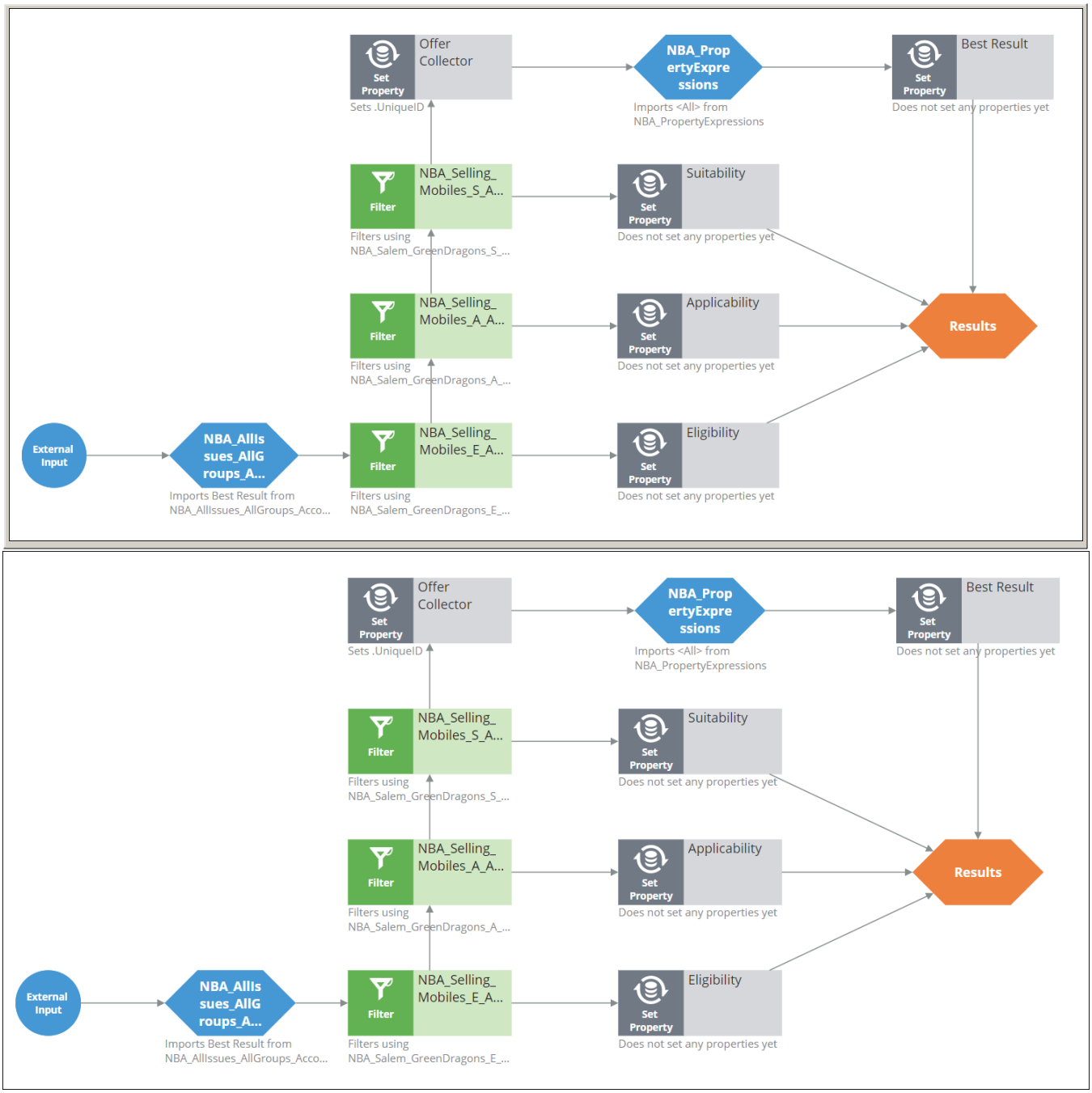
The following figure shows an example of the top-level Account context strategy for issues Selling, Retention and Acquisition:



Next is an example of an issue-level Account context strategy for the Selling issue with groups Mobiles and Plans:



Finally, the following example shows the group-level Account context strategy containing the All Issues / All Groups strategy and the Eligibility, Applicability and Suitability proposition filters for the Selling issue, and Mobile group:



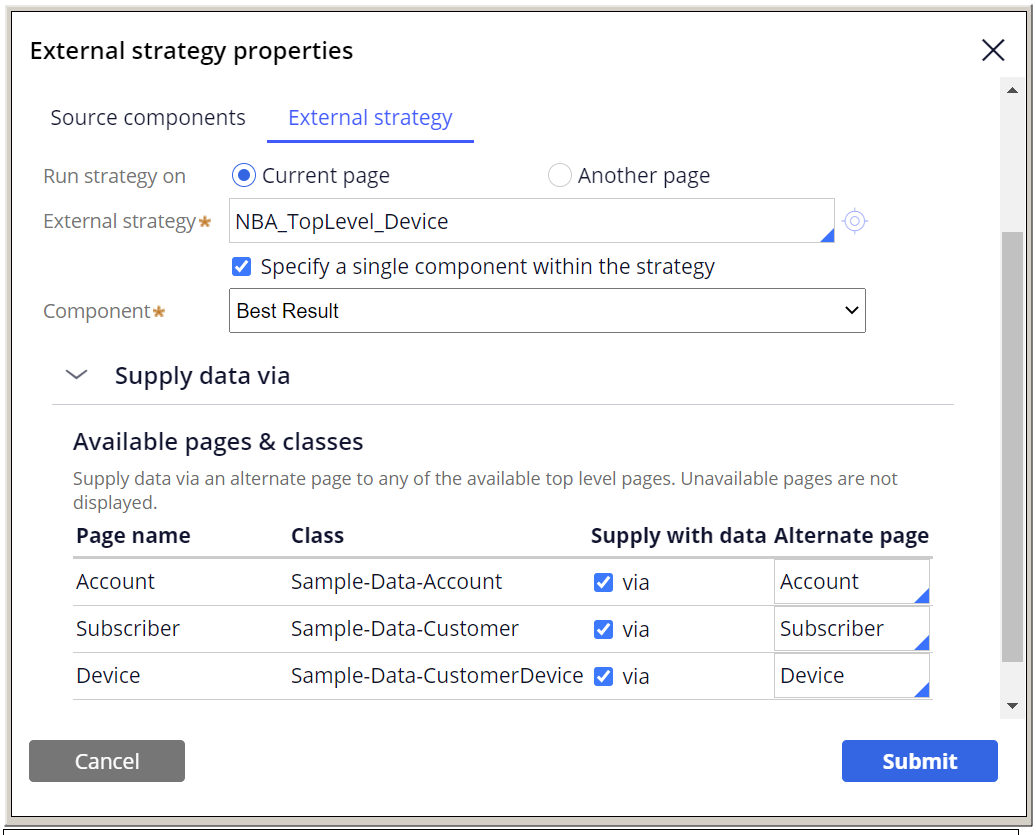
The Offer Collector shape sets the *UniqueID* property for internal use, and these strategies also contains a *NBA Property Expressions* sub-strategy for evaluating custom property expressions – a new feature introduced in Pega Customer Decision Hub 8.6.

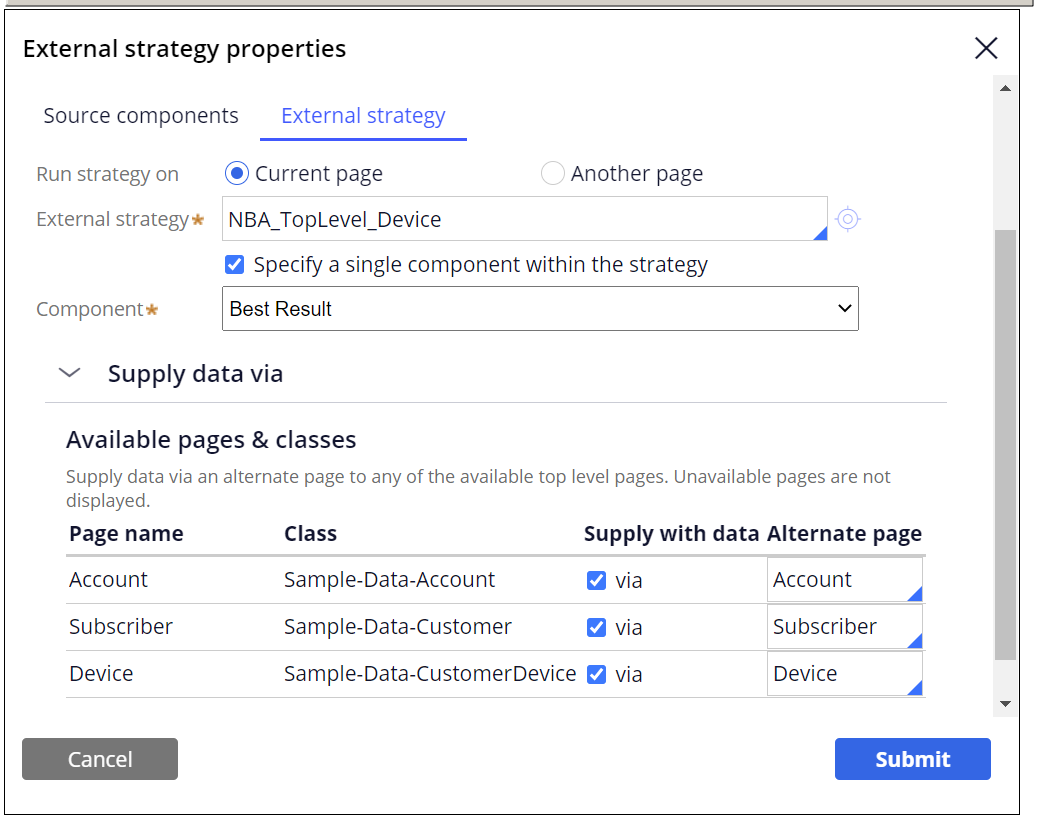
There are four exit points:

- Best Result is used by the Next-Best-Action Strategy Framework for actions that pass all conditions.
- Eligibility, Applicability and Suitability are used by the Simulation feature to show how actions successively drop out as they progress through the filter stages.

Since these context-level proposition filters need access to the data page from the context they are evaluating, and also for any of the pages for the higher-level contexts, all such pages are defined in the **Pages & Classes** tab of the proposition filter definition. The page names are the same as the context names and are passed to the proposition filter via the **Available pages & classes** section of the Filter properties shape, as shown in the example in the example for the All Actions Engagement Policies.

Furthermore, all of the sub-strategies that are called on the way to access these proposition filters also need these page definitions in their **Pages & Classes** tab, and need to pass the page data using the **Available pages & classes** section of the sub-strategy shape as shown in the example for the top-level Device context below:





NBA Strategy framework

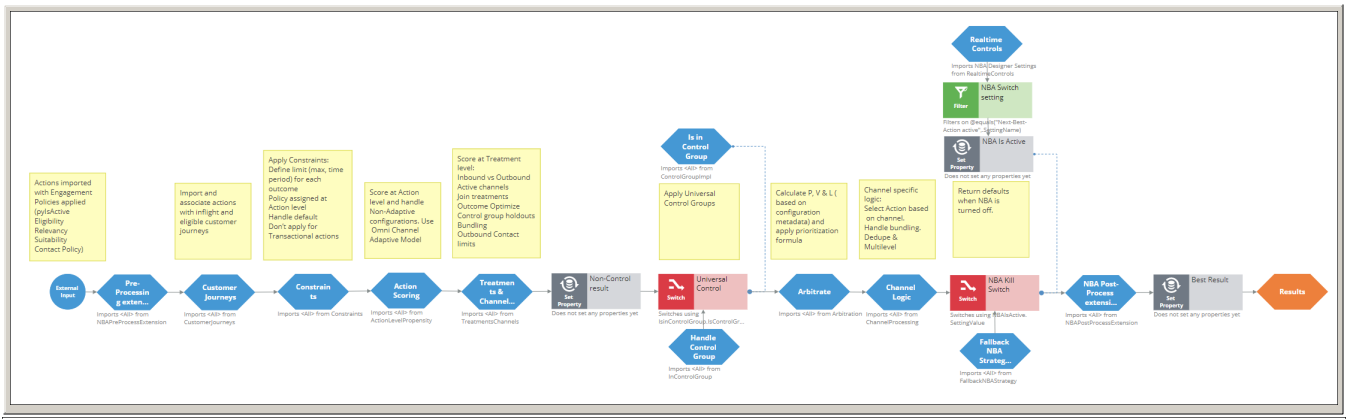
This is executed separately for each context after the engagement polices have been evaluated for that context, and performs the majority of the work within Next-Best-Action Designer.

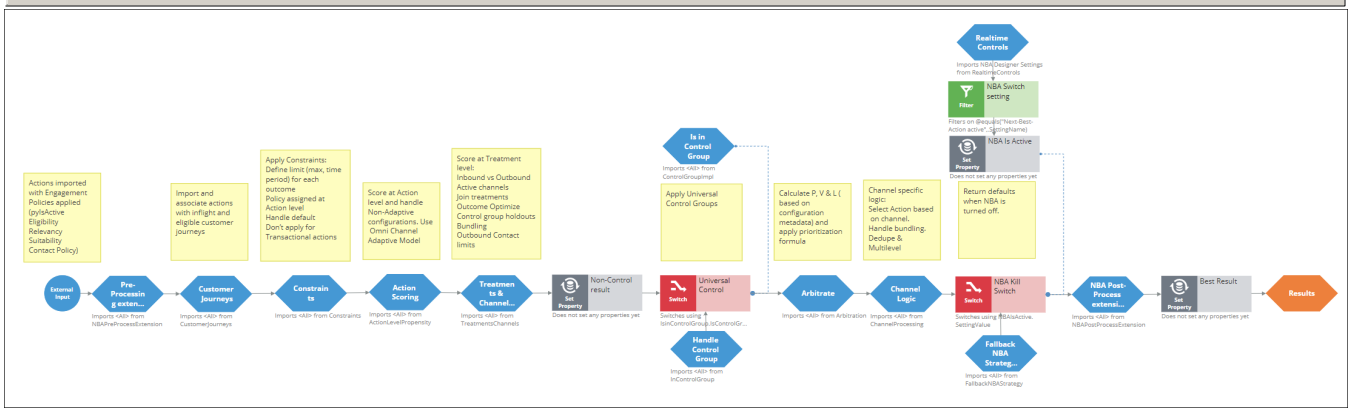
Pega Customer Decision Hub

The strategy framework is shipped with Pega Customer Decision Hub and is managed by Next-Best-Action Designer. It comprises a top-level strategy called NBA Strategy Framework and a set of related sub-strategies, most of which are provided as is, but some are generated by Next-Best-Action Designer based on the Context Dictionary.**Note:** For more information about the Context Dictionary, see Understanding multilevel decisioning and the Context Dictionary.

There are several extension points within the framework, and with the exception of these designated extension points, the framework strategies are not modifiable. To ensure upgradeability, avoid overtaking any part of the framework that is not a designated extension point. If you cannot implement the functionality you require using the extension points and feel that you need to overtake any part of the framework, please reach out to the Pega product team before doing so.

The figure below shows the whole of the top-level strategy framework. The following sections describe individual components of the framework in more detail.



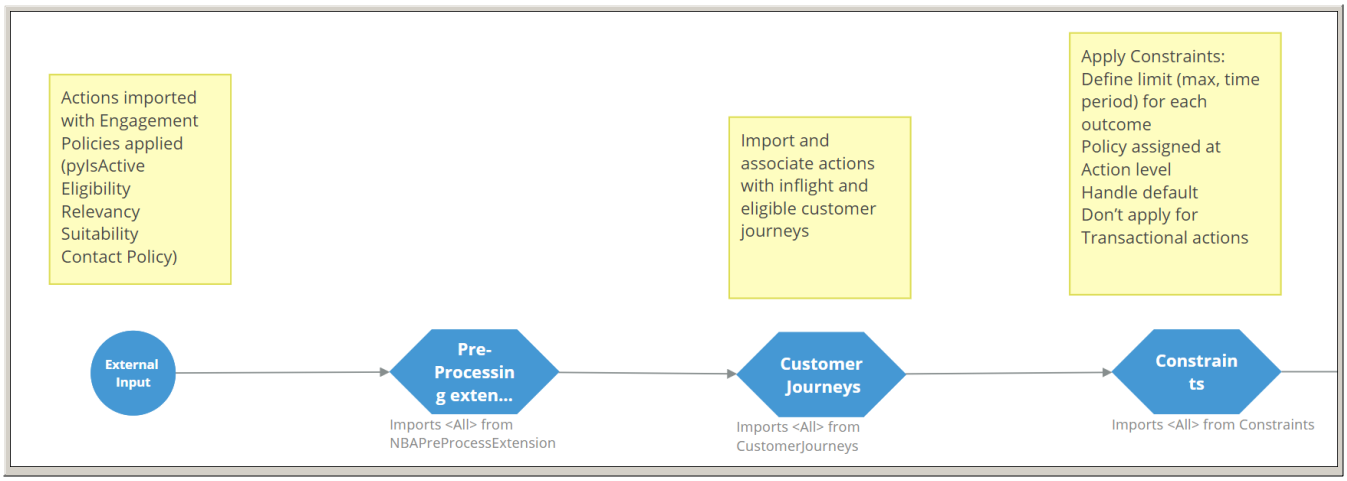


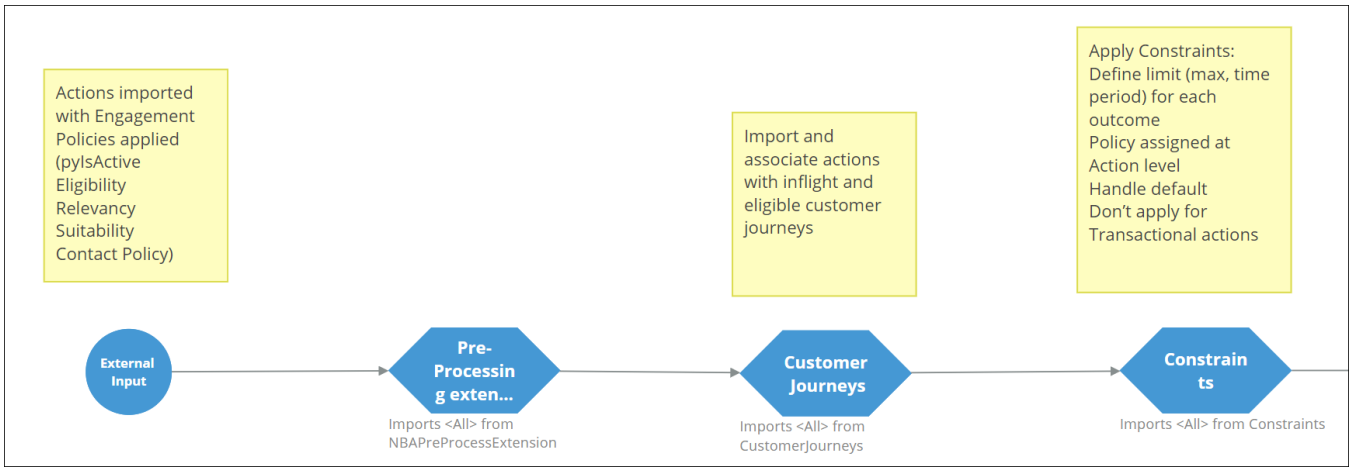
- NBA Strategy framework components: External Input through to Constraints
- NBA Strategy framework components: Action scoring through to control group handling
- NBA Strategy framework components: Arbitrate through NBA Post-Process extension
- NBA Strategy framework components: Final Action Limits and Bundling strategy
- Additional NBA Strategy framework components
- NBA Strategy framework extension points

NBA Strategy framework components: External Input through to Constraints

External input includes imported active actions that have already had individual group-level and All Group level engagement policies applied.

Pega Customer Decision Hub





Pre-processing extension point

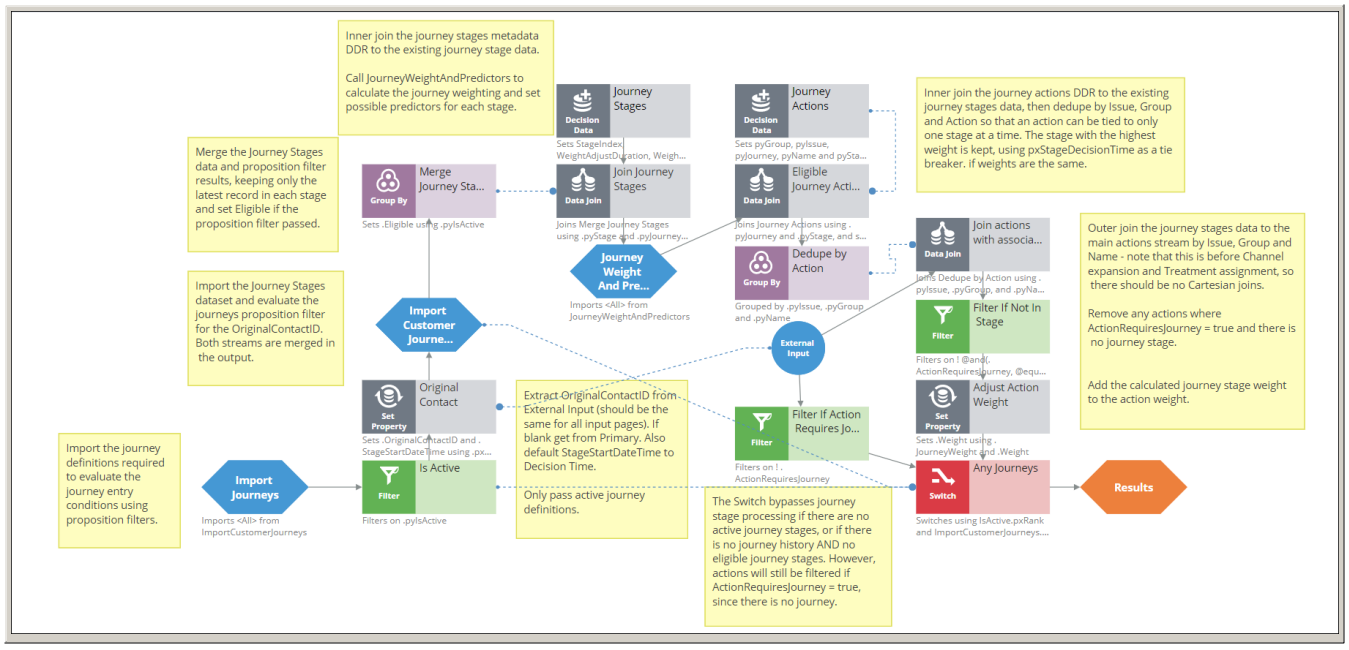
The first component within the strategy framework is an extension point named NBAPreProcessExtensionPoint that is used for any action pre-processing which you might need to perform after actions have passed all of the engagement policies and any custom property expressions have been evaluated, but before any further processing occurs. This is shipped as an empty strategy that be overtaken by the user.

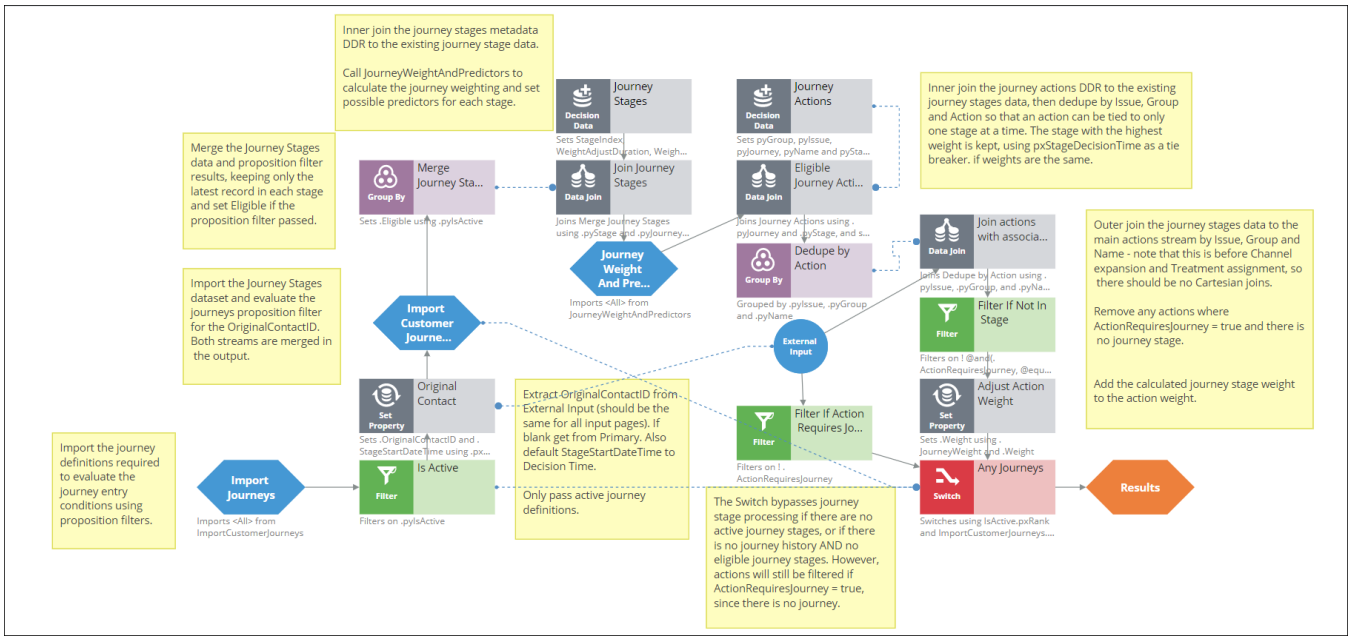
Customer Journeys

The CustomerJourneys strategy associates actions with inflight and eligible customer journeys.

It determines Journey Stage eligibility for the customer and assigns journey properties to their related actions as required for the AI models to learn, and for recording journey progress. It also adjusts action weights based on the Journey Stage weighting settings.

Each Journey Stage definition has a corresponding pseudo-action defined in the NBA_Journey or the NBA_Journey issue and group. These are not real actions but are created purely to enable a proposition filter to be used to evaluate the Journey Stage entry criteria. They are not imported by the Import Actions strategies and do not participate in any NBA arbitration.

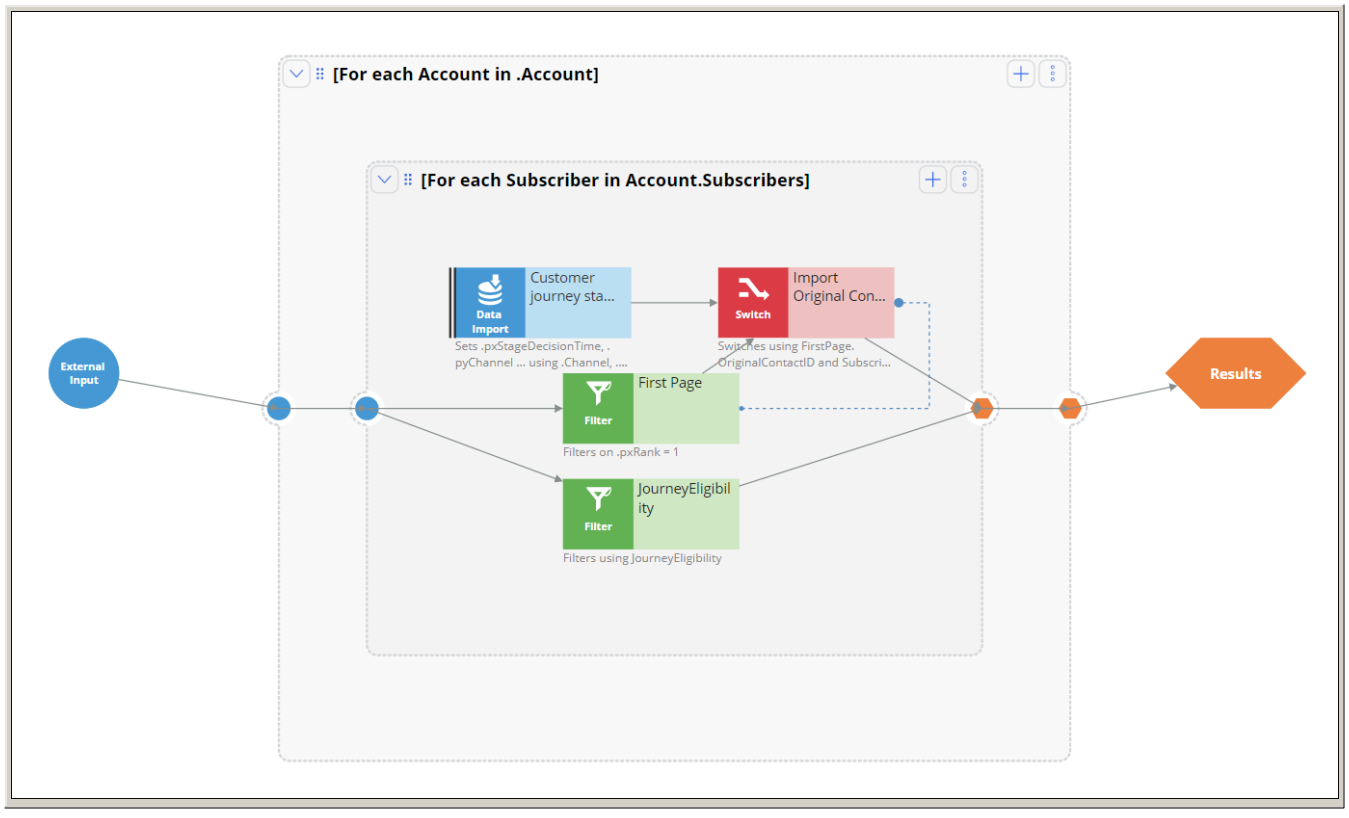


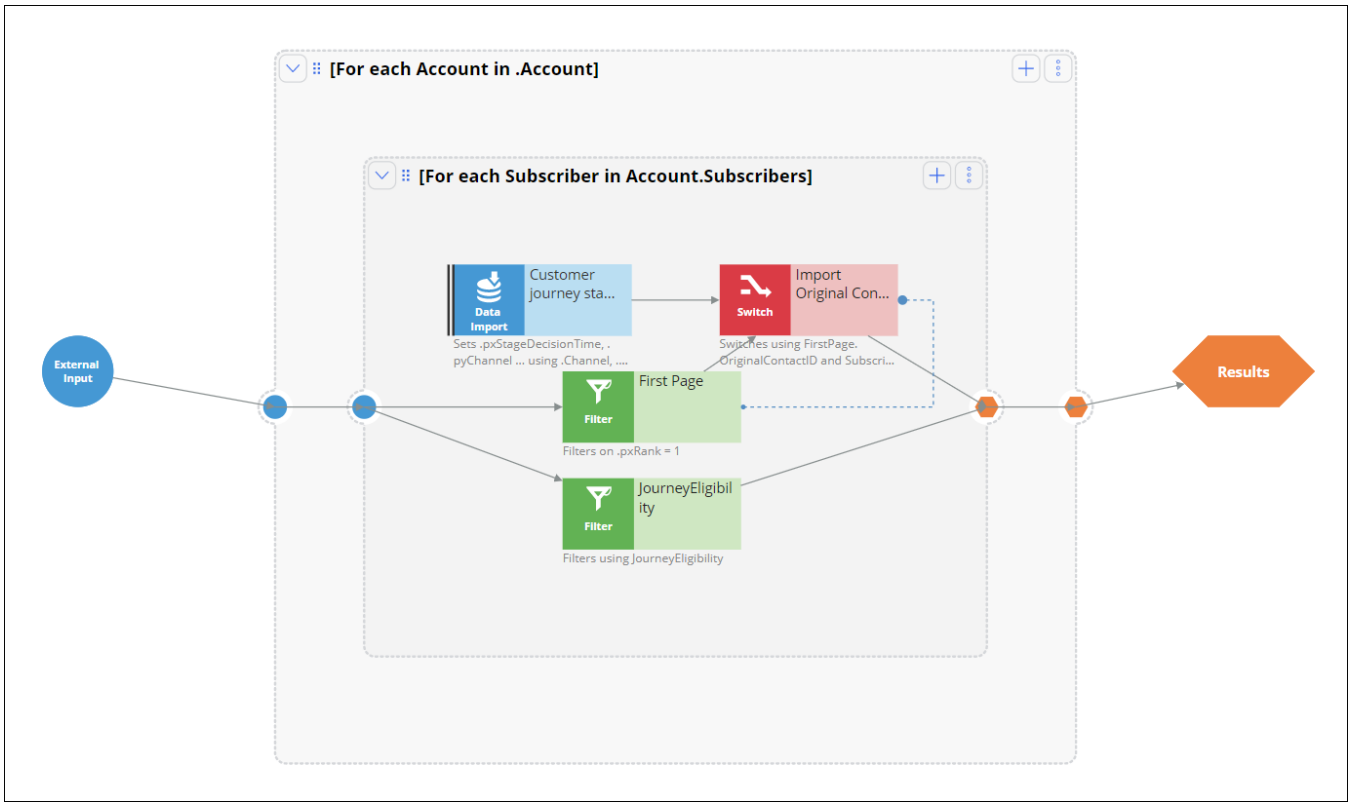


For a detailed description of the logic, refer to the annotations on the strategy.

Customer Journey Impl

This strategy is generated by NBA Designer based on the context dictionary – it is not modifiable.





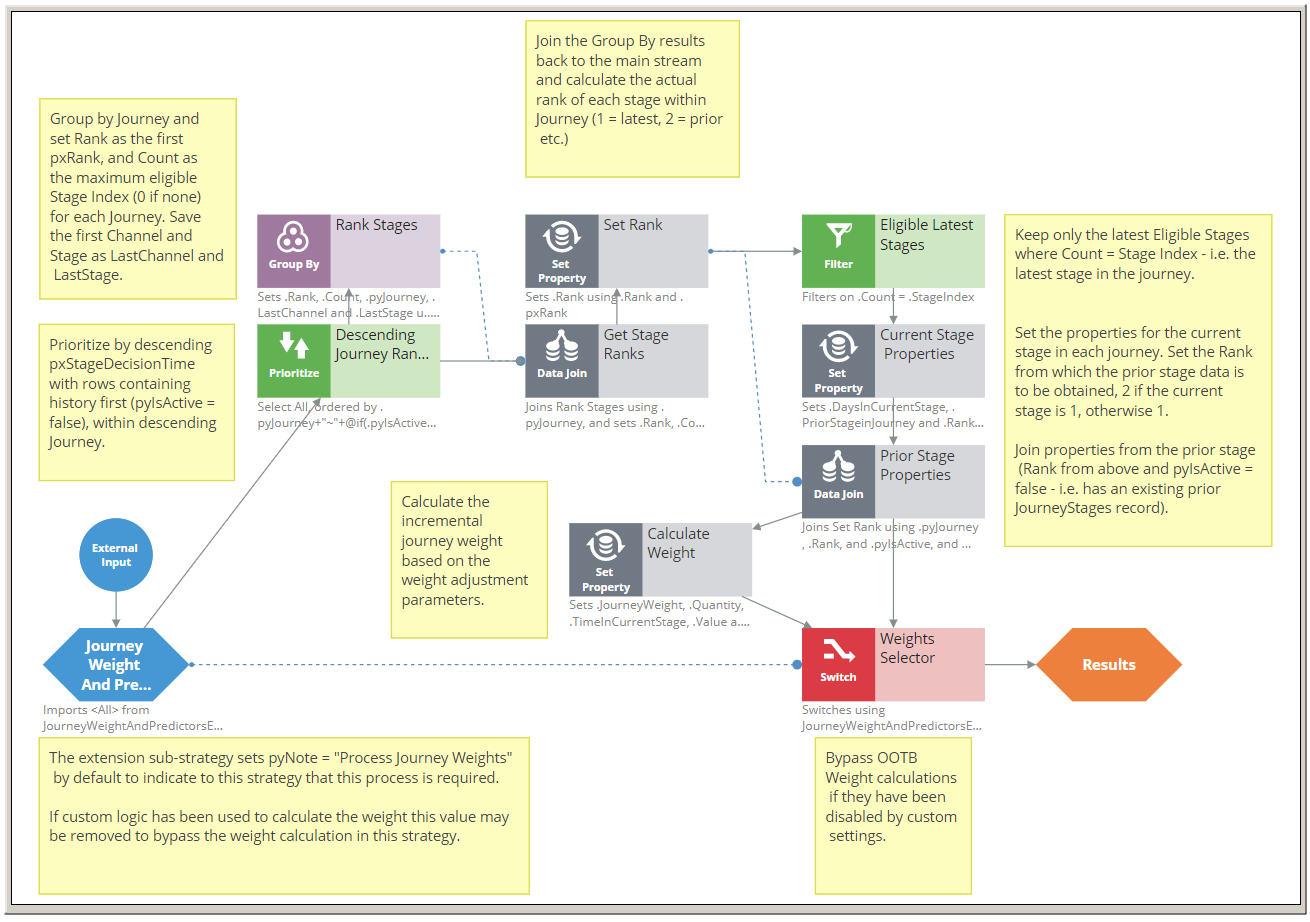
The data import shape obtains the records from the Journey Stages dataset for all relevant contacts, and the Switch rule and First Page filter ensure that only the records for the contact currently being processed are selected.

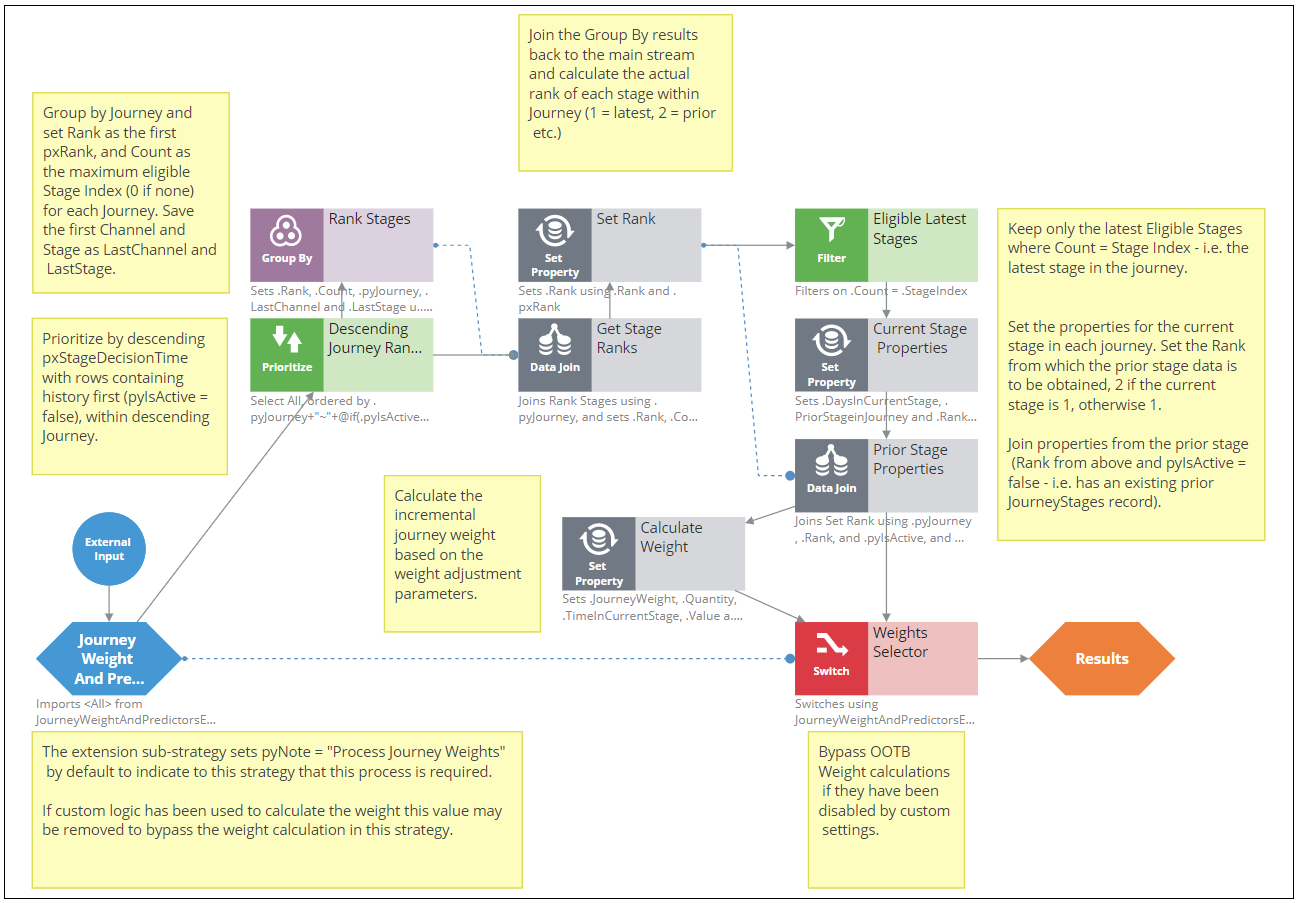
The Journey Eligibility proposition filter evaluates the entry conditions for each Journey Stage and only passes those that meet the criteria. The input conditions for the embedded context sub-strategy ensure that the filter is only evaluated for the current contact.

Both logic streams are merged and passed back to the Customer Journeys strategy.

Journey Weight And Predictors

This strategy sets journey properties and calculates the journey weights from the eligible journey metadata.

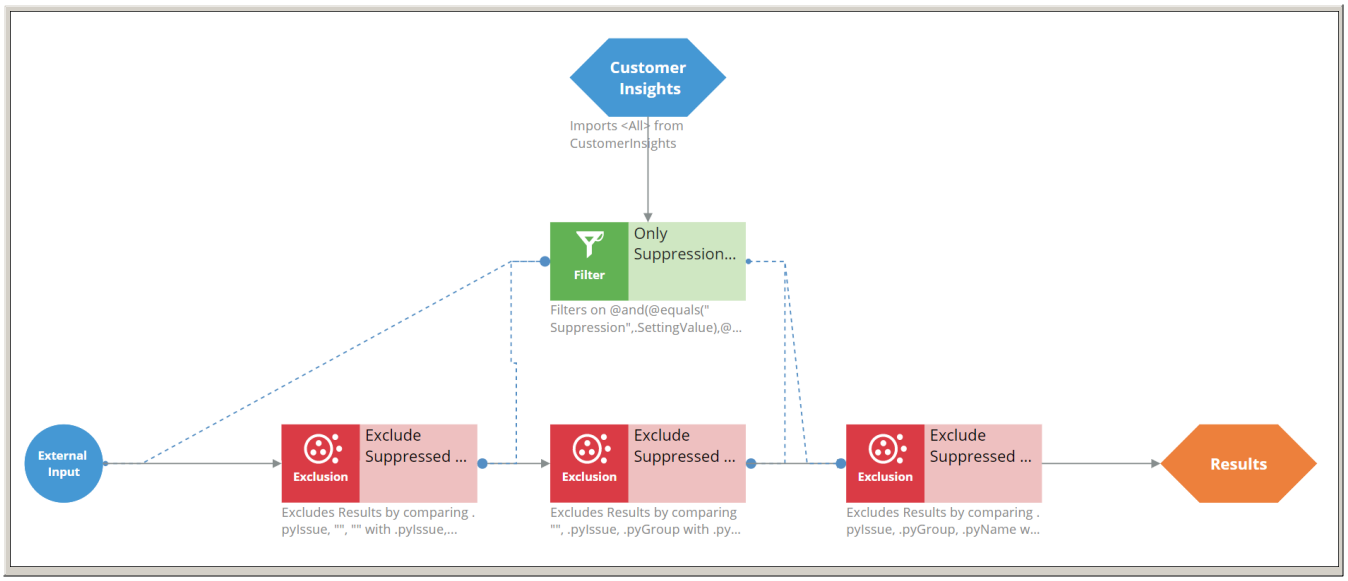


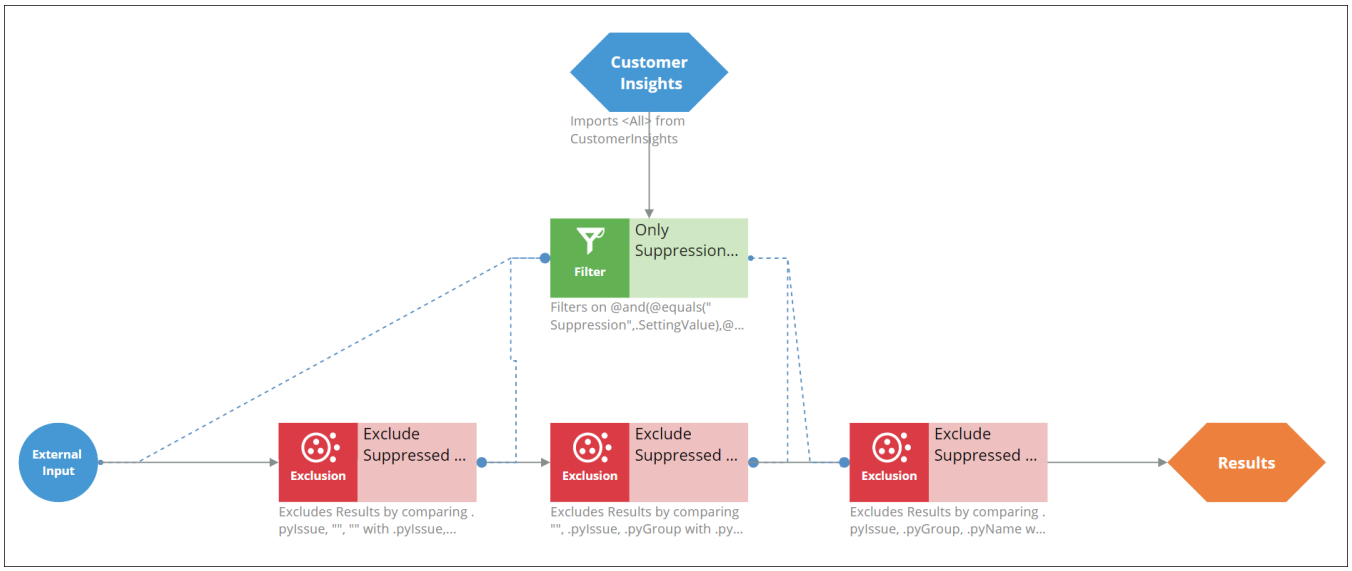


For a detailed description of the logic, refer to the annotations on the strategy.

Constraints

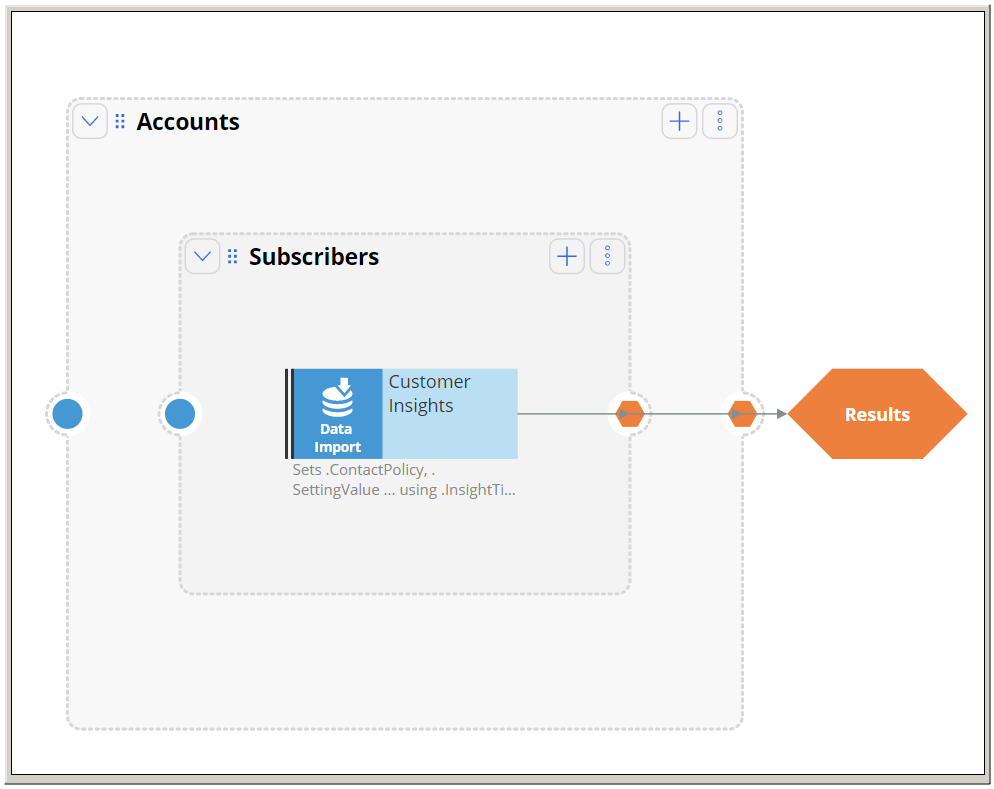
The Constraints strategy enforces any actions which need to be suppressed based on the associated contact policies. The strategy imports the suppression records, known as Action Insights, and filters out any issues, groups or actions which have matching suppression records.**Note:** Only the Action Insight records for the Original Contact ID – i.e. the contact currently being processed by the NBA Strategy Framework - are selected for suppression.

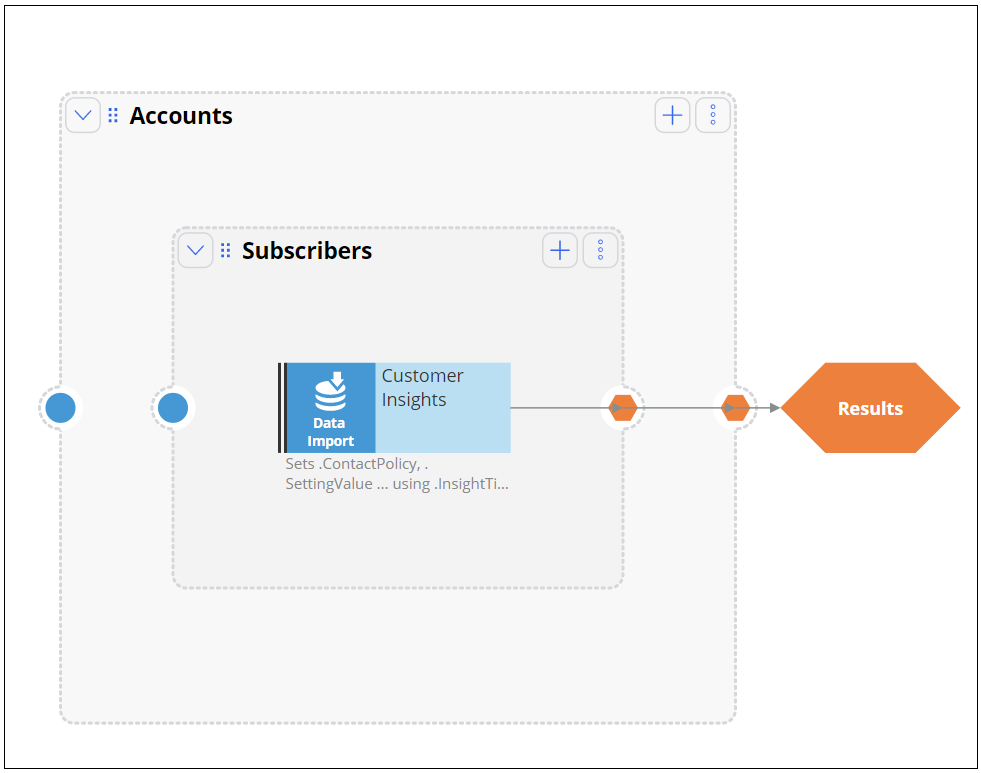




Customer Insights

The Customer Insights strategy is generated by Next-Best-Action Designer based on the context dictionary and will import records for all relevant contacts – it is not modifiable. The parent strategy is responsible for only processing the records related to the current contact, that is, the Original Contact ID.

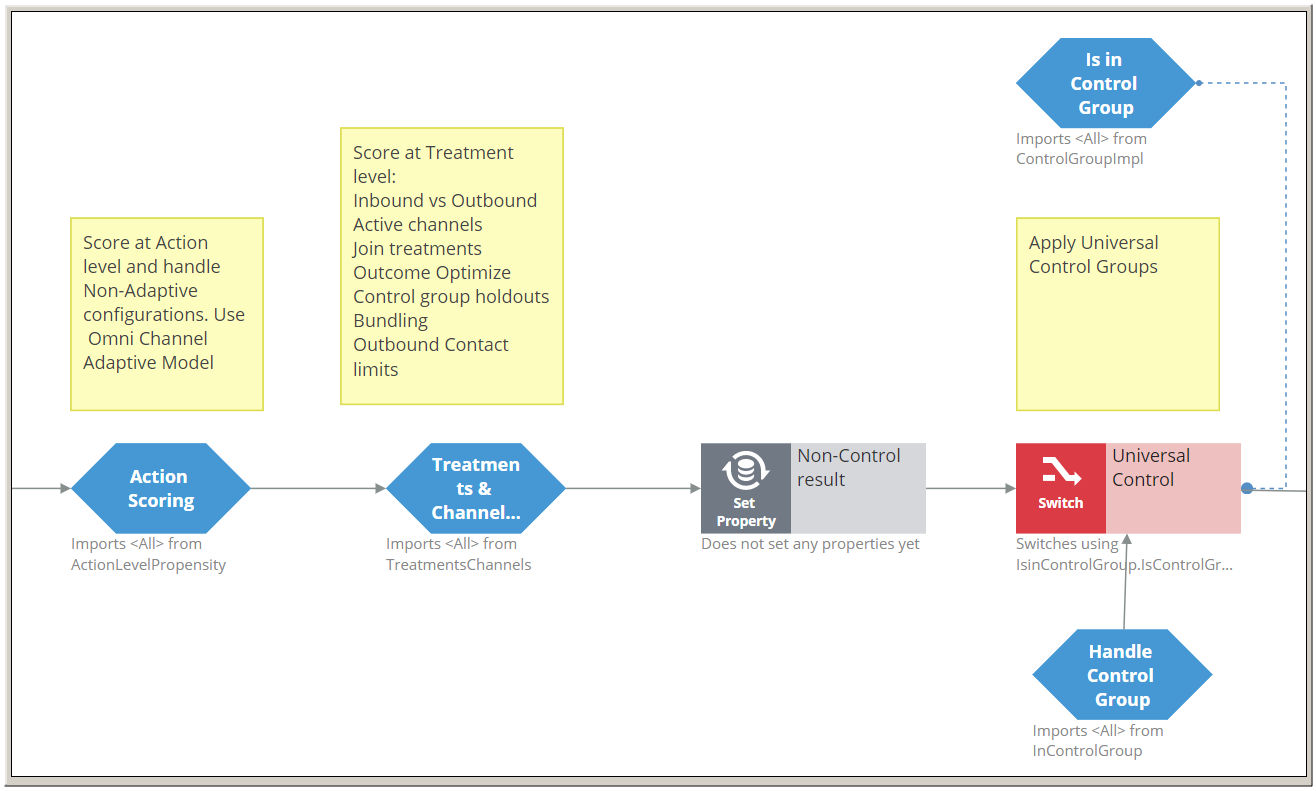


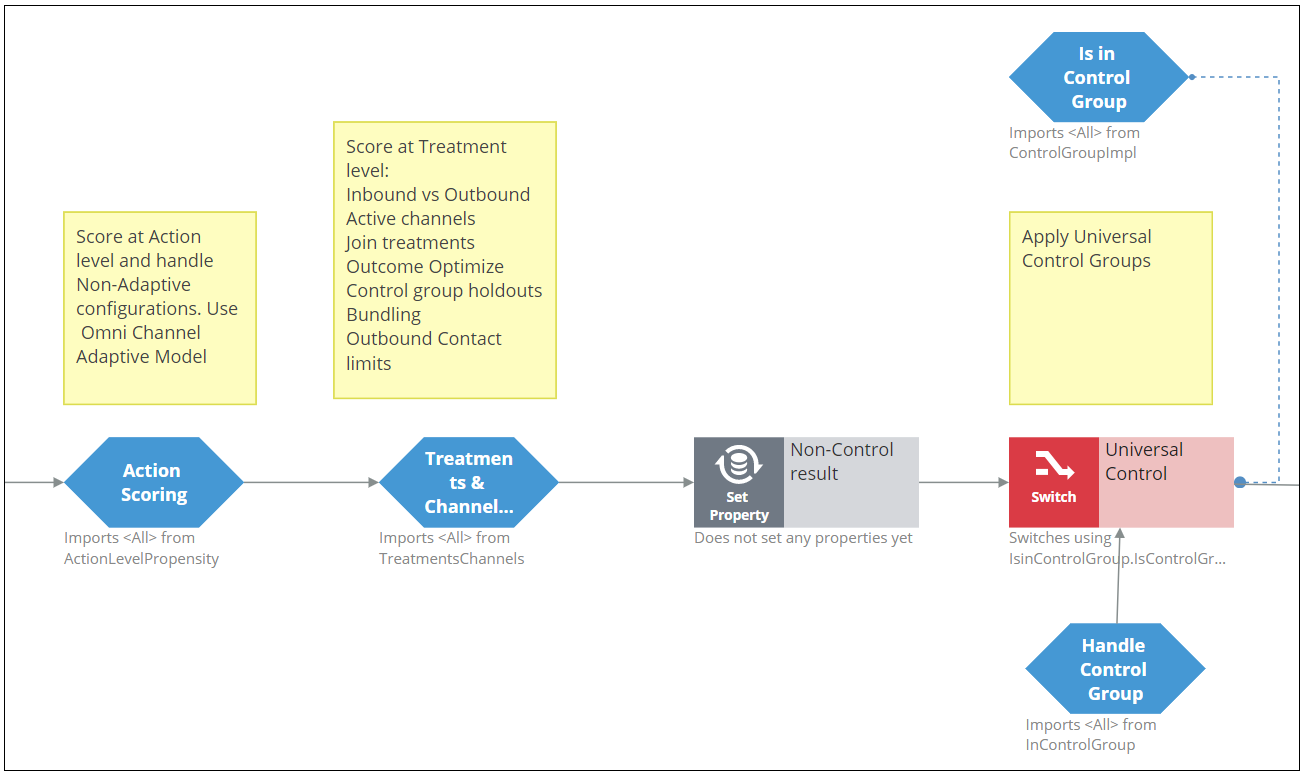


NBA Strategy framework components: Action scoring through to control group handling

This part of the Next-Best-Action strategy expands actions across active and eligible channels, applies the adaptive models, handles treatments and channels, and processes control groups.

Pega Customer Decision Hub



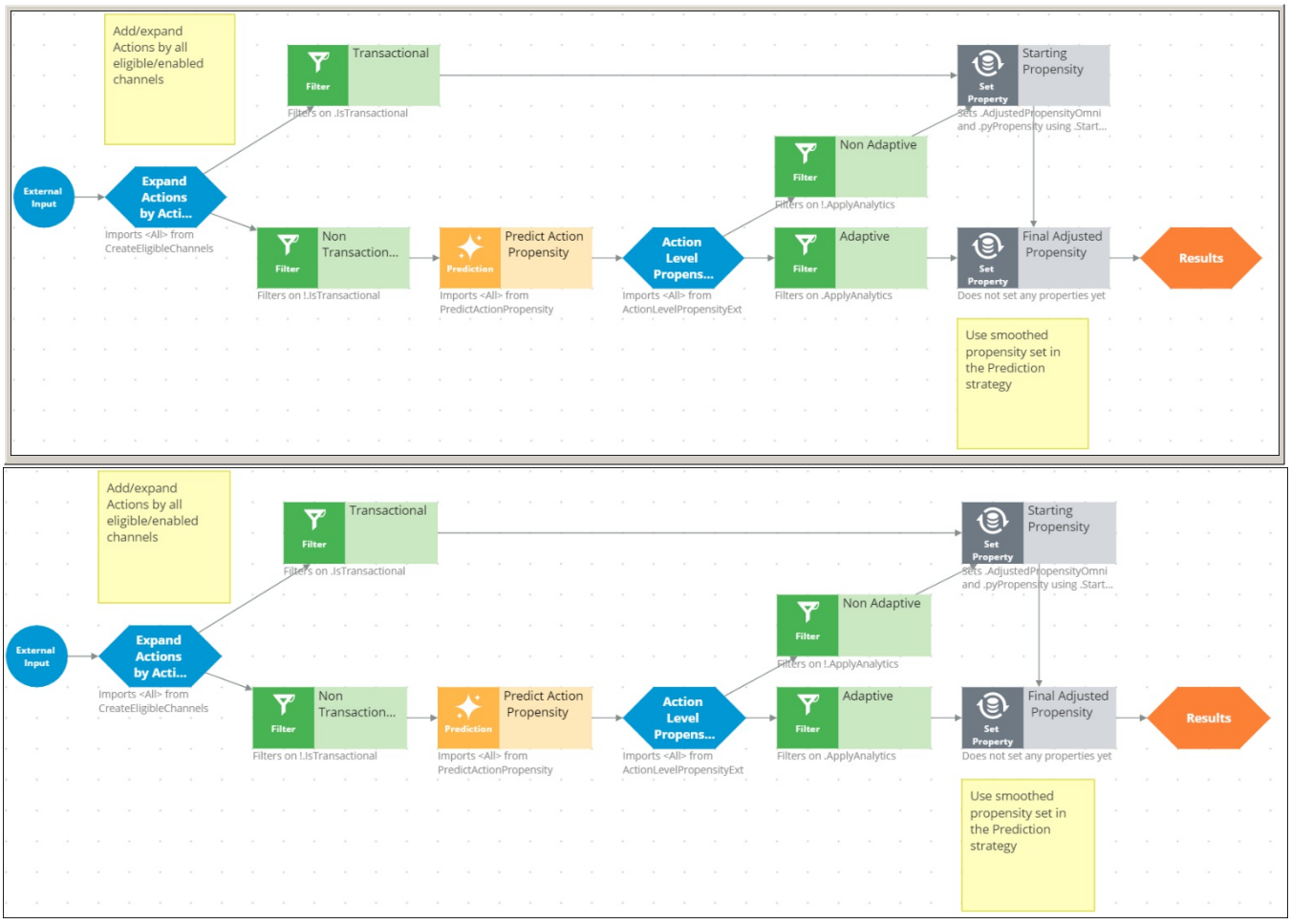


- Action Scoring (Action Level Propensity) strategy
- Treatments and channels
- Control Group Processing

Action Scoring (Action Level Propensity) strategy

The Action Level Propensity strategy is responsible for applying an adaptive model for each action and calculating the propensity across eligible channels.

Pega Customer Decision Hub



This strategy first expands the number of actions by the number of active and eligible channels using the Create Eligibile Channels strategy.

Actions can be designated as transactional in nature, i.e. they are always intended to be delivered and do not require any arbitration. Transactional actions are filtered out and bypass analytics processing.

Non-transactional actions are directed to the Predict Action Propensity Prediction strategy, and then to the Action Level Propensity Ext extension point sub-strategy.

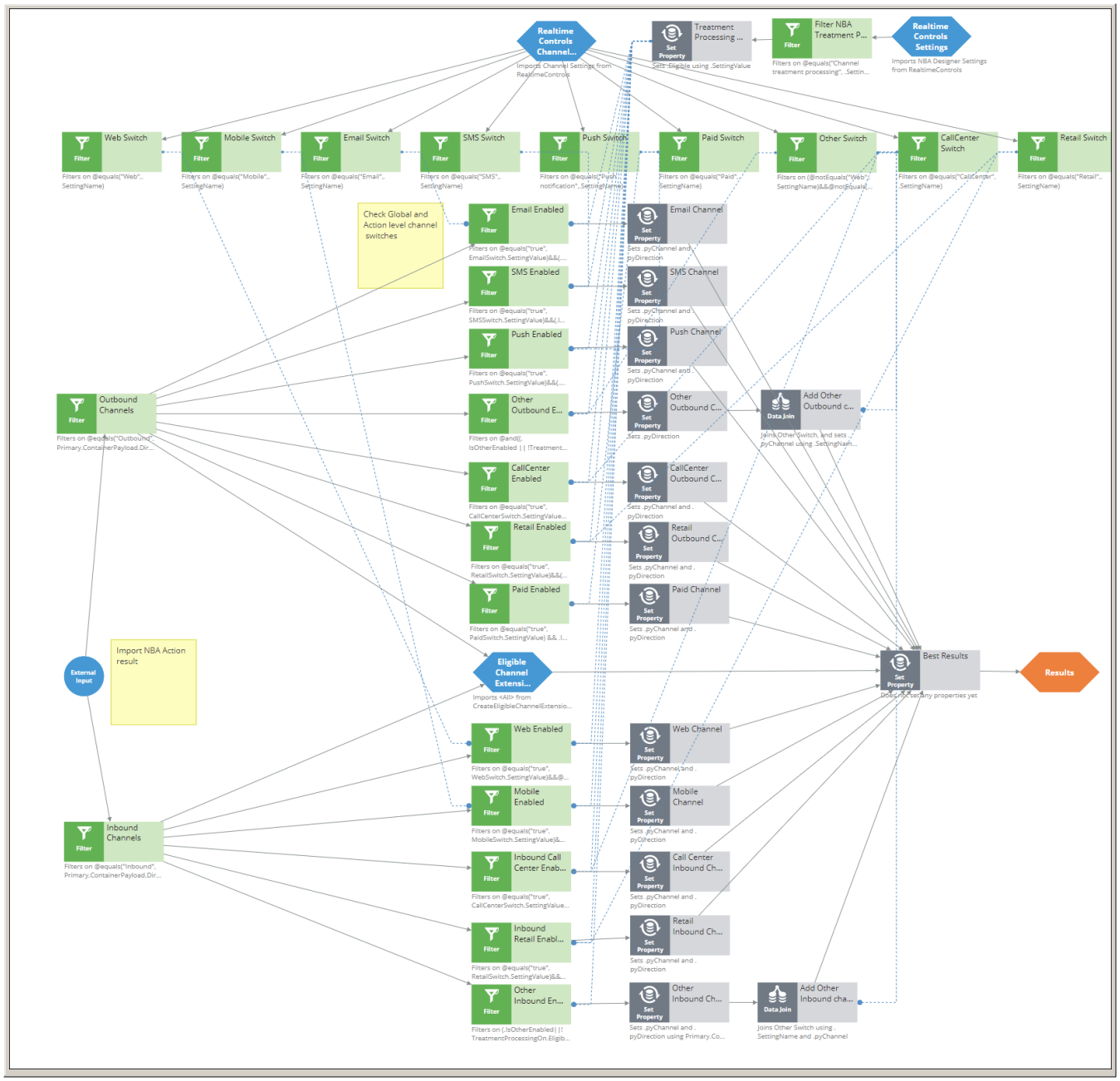
Finally, the actions are divided into adaptive and non-adaptive streams based on the ApplyAnalytics action property, and the non-adaptive stream sets the propensity based on the action starting propensity, whereas for the adaptive stream these values are calculated by the Predict Action Propensity strategy.

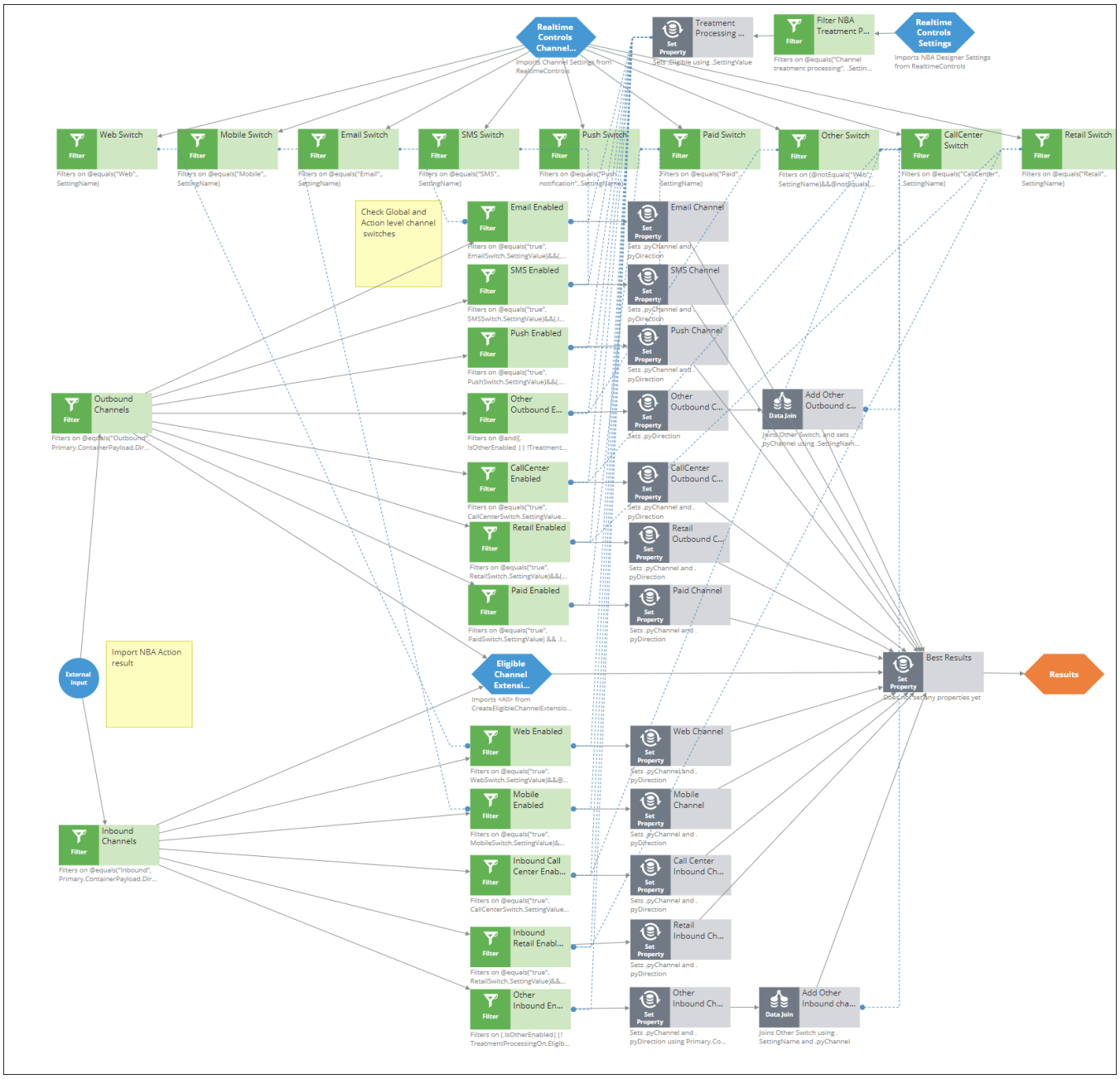
- Expand Actions by Active Channels (Create Eligibile Channels) strategy
- Predict Action Propensity prediction strategy

Expand Actions by Active Channels (Create Eligibile Channels) strategy

This strategy determines the active channels as specified in Next-Best-Action Designer, as well as the eligible channels designated on the treatment tab of the action. If the trigger is from an inbound channel, then only that specific channel is eligible and processed.

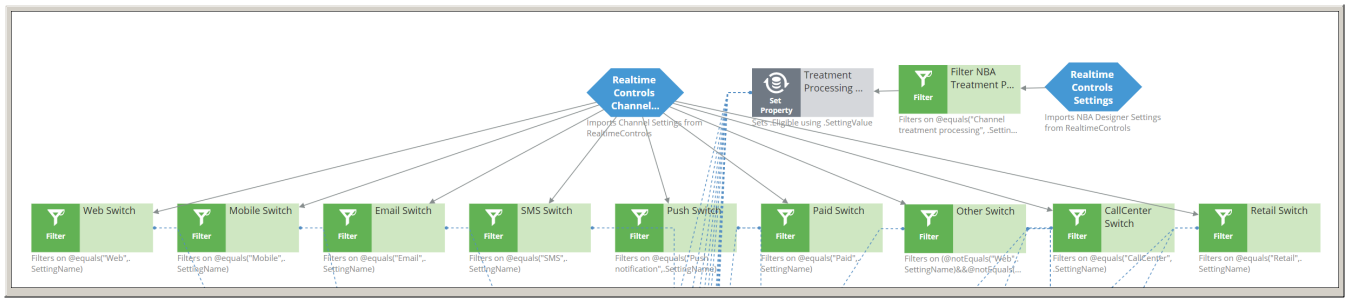
Pega Customer Decision Hub

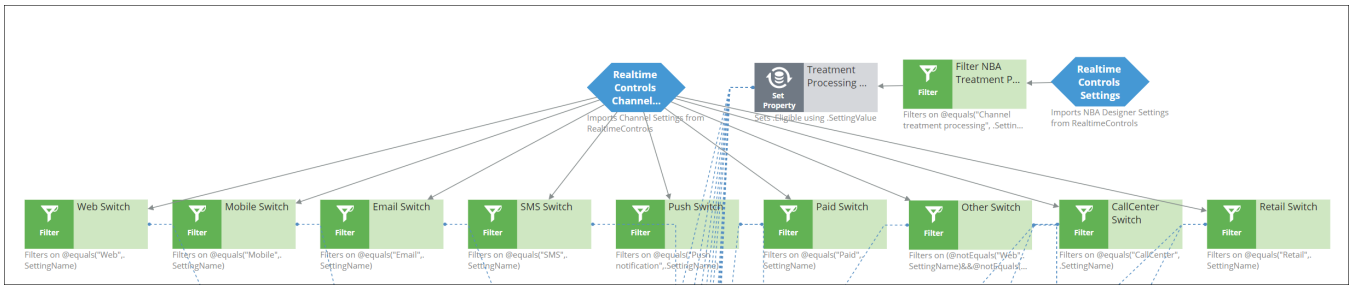




The top section of the strategy evaluates each of the real-time controls that determine which channels are active. The Filter shapes that select these settings are then referenced by other components in the strategy to evaluate which actions are eligible for each active channel.

The Treatment Processing On shape determines whether the Channel treatment processing Next-Best-Action Designer Settings parameter is set. This is referenced in each of the channel eligibility conditions (except for Paid).

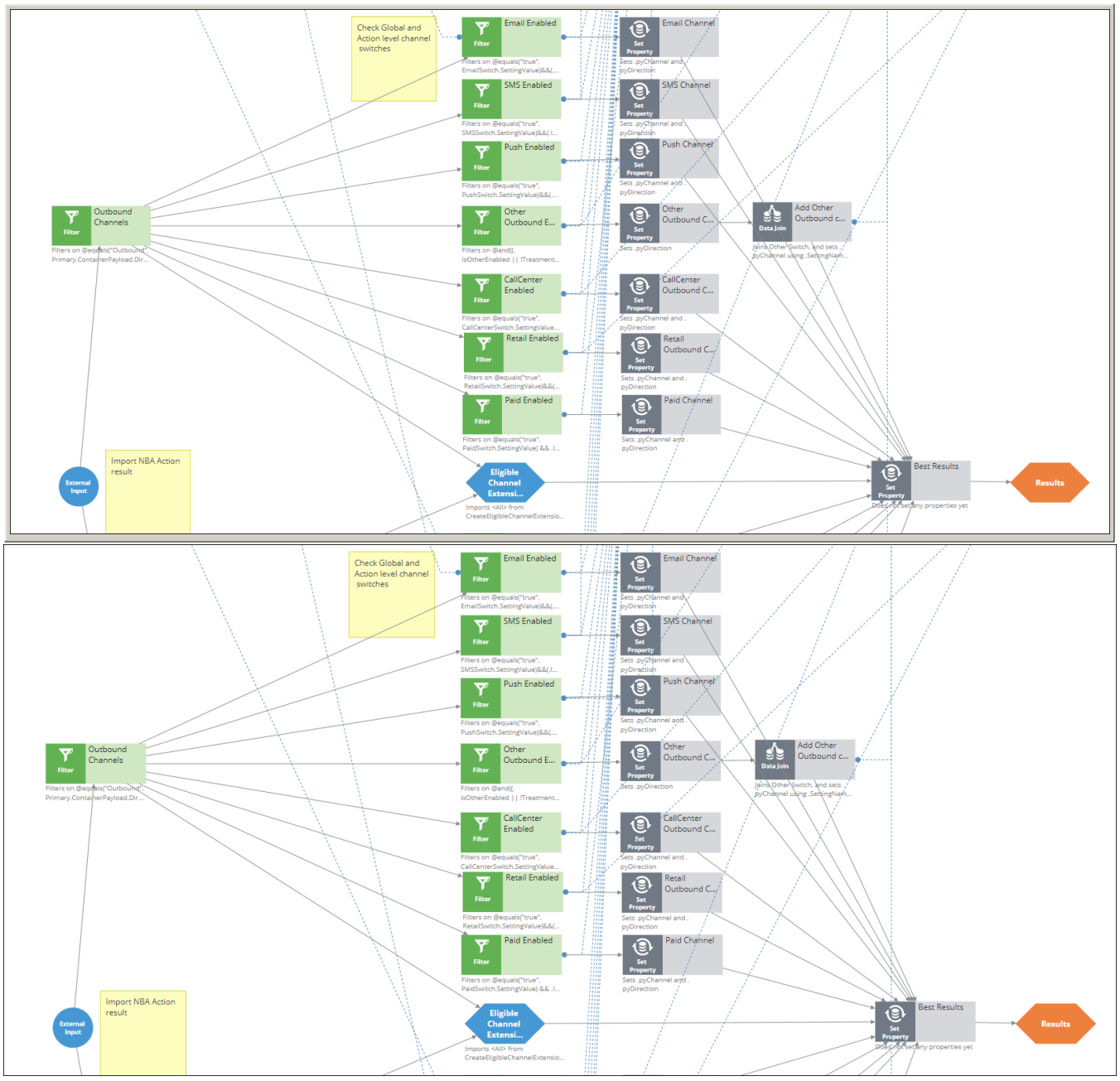




Next best actions are fed in through the External Input shape and are split into Outbound and Inbound streams for further processing based on the value of *Primary.ContainerPayload.Direction. pyChannel* and *pyDirection* are then set for each eligible channel.

An extension point is provided by the Create Eligible Channel Extension sub-strategy, in which additional eligible channels may be added.

The Outbound section is shown below, and the inbound section follows a similar pattern. Note that there is provision for including any non-standard channels (**Add Other Outbound Channels**) so long as they have been defined within the real-time controls.



For outbound, each action is evaluated against each channel to determine eligibility using logic similar to the following example for the Email channel:

```
@equals("true", EmailSwitch.SettingValue)
&& (.IsEmailEnabled || !TreatmentProcessingOn.Eligible)
```

The first part of the expression determines whether the channel is actually enabled in Next-Best-Action Designer, and if it is the action will be eligible for that channel if it has a treatment for the channel (IsEmailEnabled), or if treatment processing is not enabled in Next-Best-Action Designer. Note that since the Paid channel does not use treatments, this last check is not required.

For inbound, the logic is slightly different as shown in the following example for the Web channel:

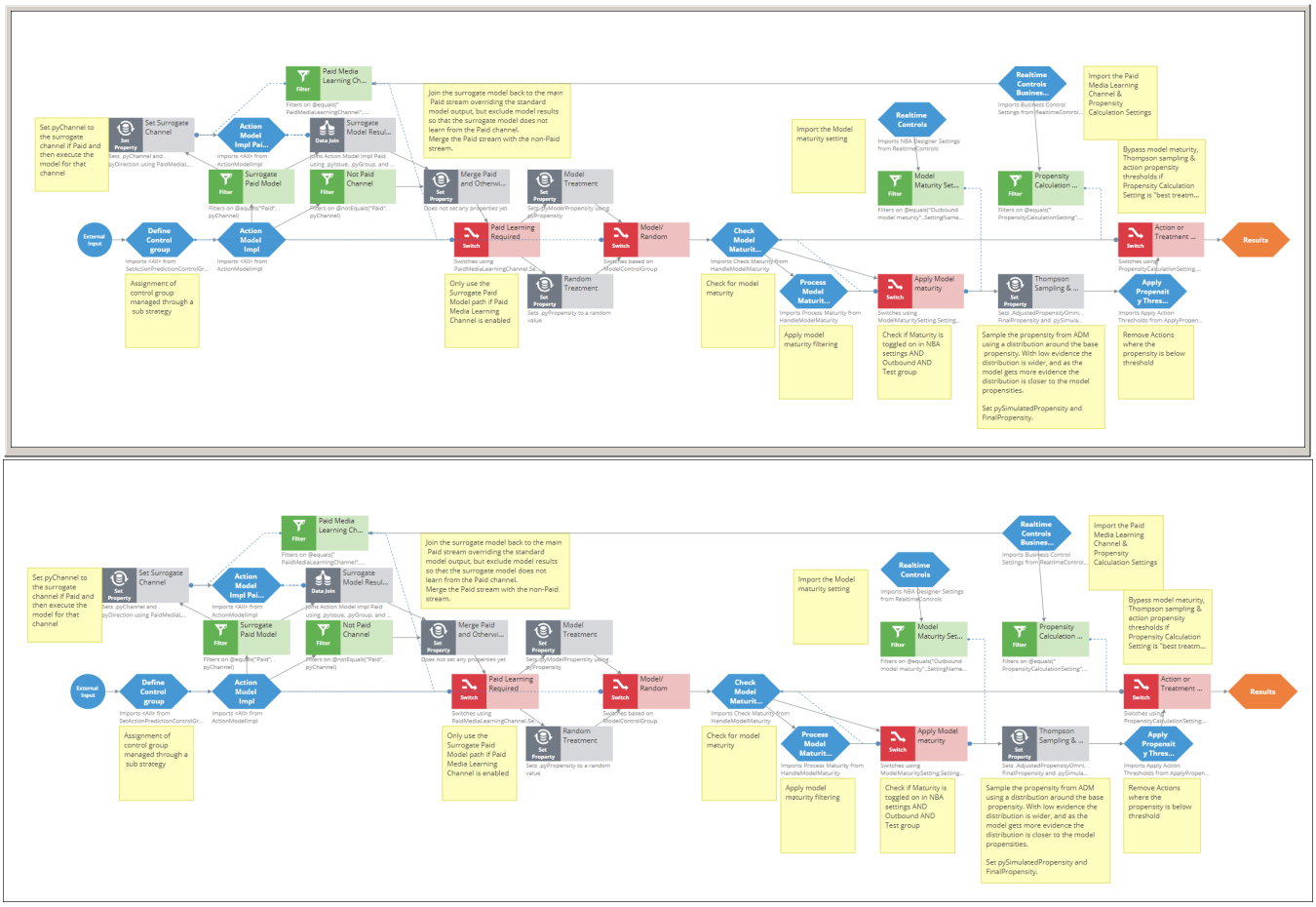
@equals("true", WebSwitch.SettingValue)
&& @equals("Web", Primary.ContainerPayload.Channel)
&& (.IsWebEnabled || !TreatmentProcessingOn.Eligible)

The first part of the expression determines whether the channel is actually enabled in Next-Best-Action Designer, and if it is the second expression determines whether the current inbound request is for the channel, and if so the action will be eligible for that channel if it has a treatment for the channel (*IsWebEnabled*), or if treatment processing is not enabled in Next-Best-Action Designer.

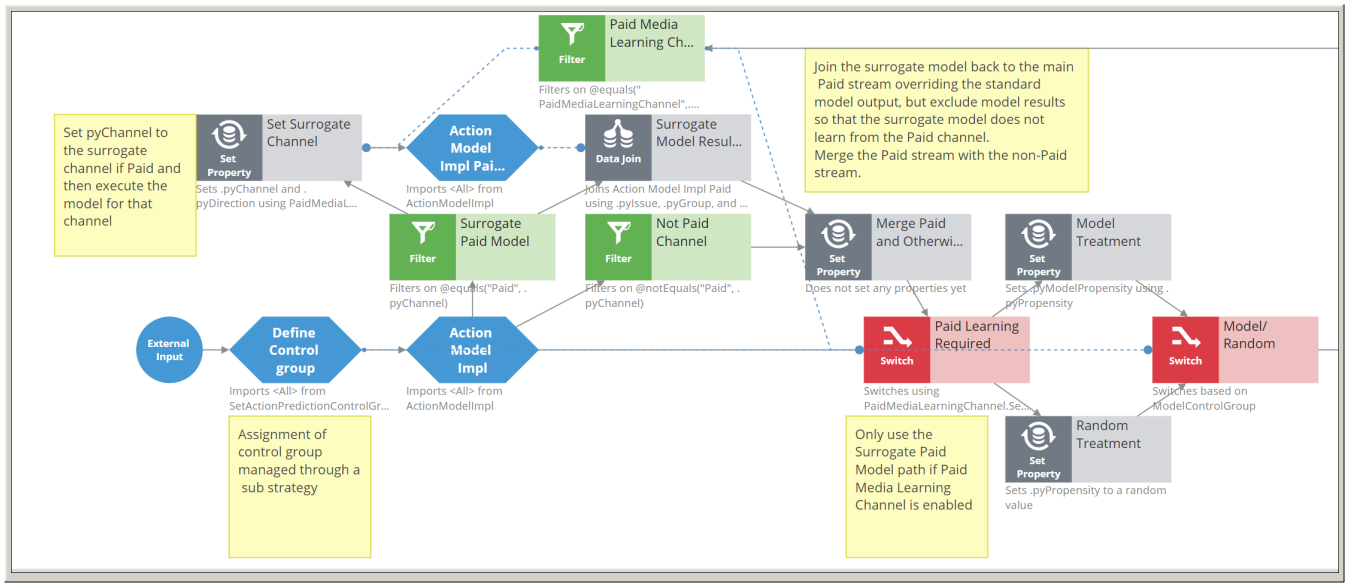
Predict Action Propensity prediction strategy

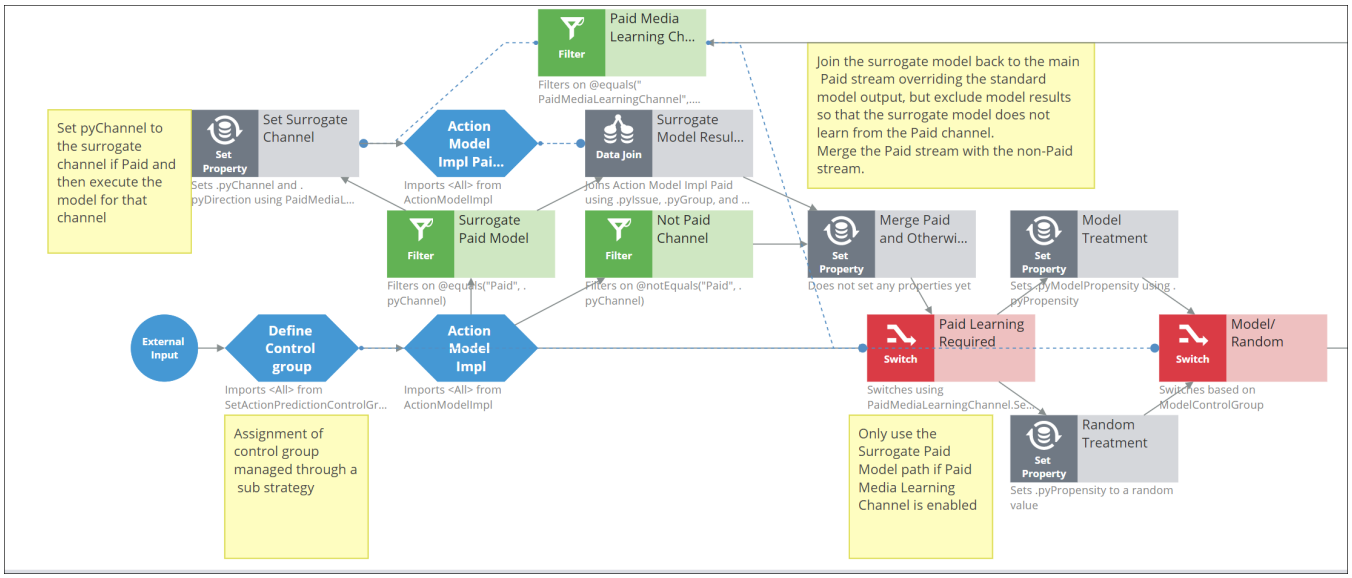
PredictActionPropensity is a prediction strategy that is accessible for modification from Prediction Studio. The full strategy is shown below.

Pega Customer Decision Hub



The leftmost section starts by defining the model control group properties and executing the Paid channel surrogate model if required.



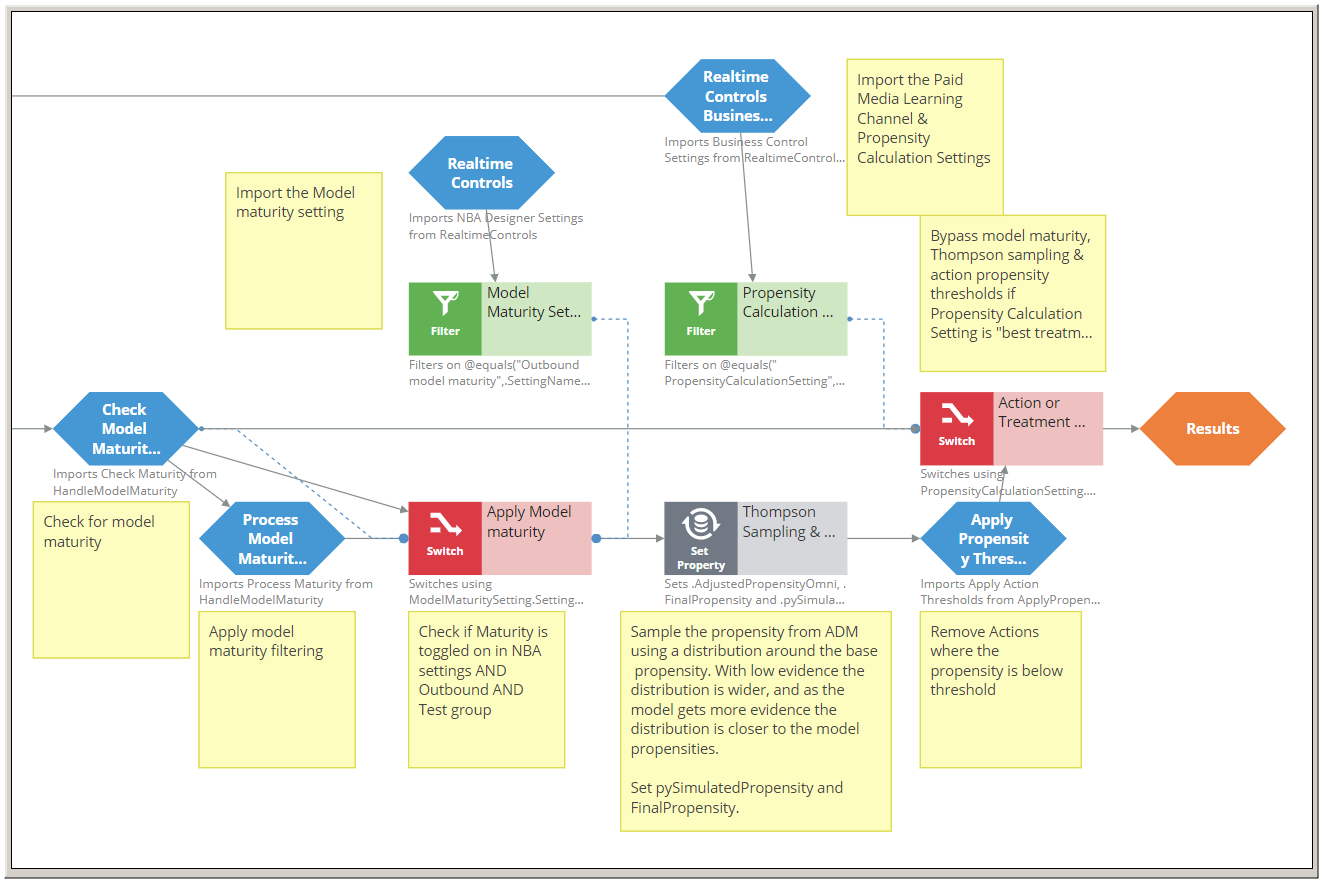


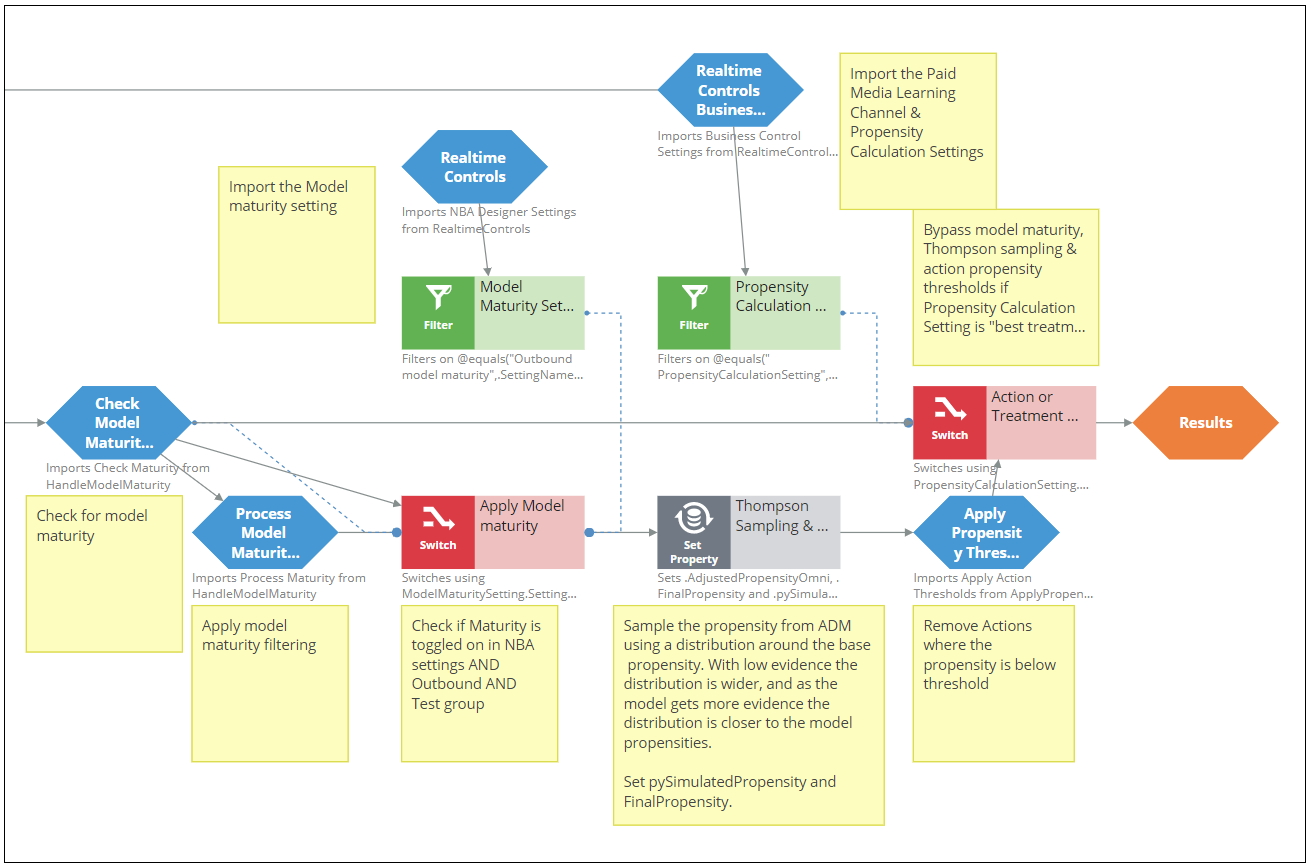
A surrogate Paid Media channel may be defined via the *PaidMediaLearningChannel* setting in the **Business Control Settings** DDR, and if defined, the model for this channel will be used instead of the Paid channel model for actions where Paid is enabled. Typically, the reason for choosing a surrogate channel would be that the Paid channel is receiving very little feedback and so the model has insufficient evidence from which to learn.

The logic for this section is described below.

- A control group is assigned through the Define Control Group sub strategy.
- The Action Model Impl strategy executes the adaptive model for the current contact and context for each action and enabled channel.
- Actions are split into Paid and Not Paid streams based on whether the action has Paid enabled.
- For Paid actions, the channel is changed to the surrogate, the model for that channel is executed, and the results are joined back to the Paid action stream overriding the model results for the Paid media channel. It is important to note that the Exclude model results from Action Model Impl Paid is checked in the data join so that only the original Paid model learns from any response received, and the surrogate model does not learn from the Paid channel.
- Non-Paid actions bypass the surrogate model path.
- Paid and Non-Paid actions are merged.
- The Paid Learning Required switch rule bypasses the whole paid surrogate logic path if a surrogate channel has not been defined.
- The Model/Random Switch rule then chooses between the model results and a random propensity (between 0 and 1) based on the control group setting.

The next section of the strategy applies outbound model maturity, Thompson sampling, and propensity thresholds.





For a detailed description of the logic, refer to the annotations on the strategy.

Define Control group

By default, 2% of the actions receive a random propensity between 0 and 1. A certain amount of randomness is always important to ensure the models continue to learn new patterns. The percentage may be changed within Prediction Studio.

This is achieved by calculating a *CustomerHash* value between 0 and 100 based on the customer ID (*pySubjectId*) and the month name, then selecting 2% for a random Control group, and the remaining 98% for the Test group (using the *Select Treatment Switch* rule). Including the month name in the hash string provides each customer with a value that will remain constant for one month, so that the customer remains part of the same control group for that month.

Action Model Impl

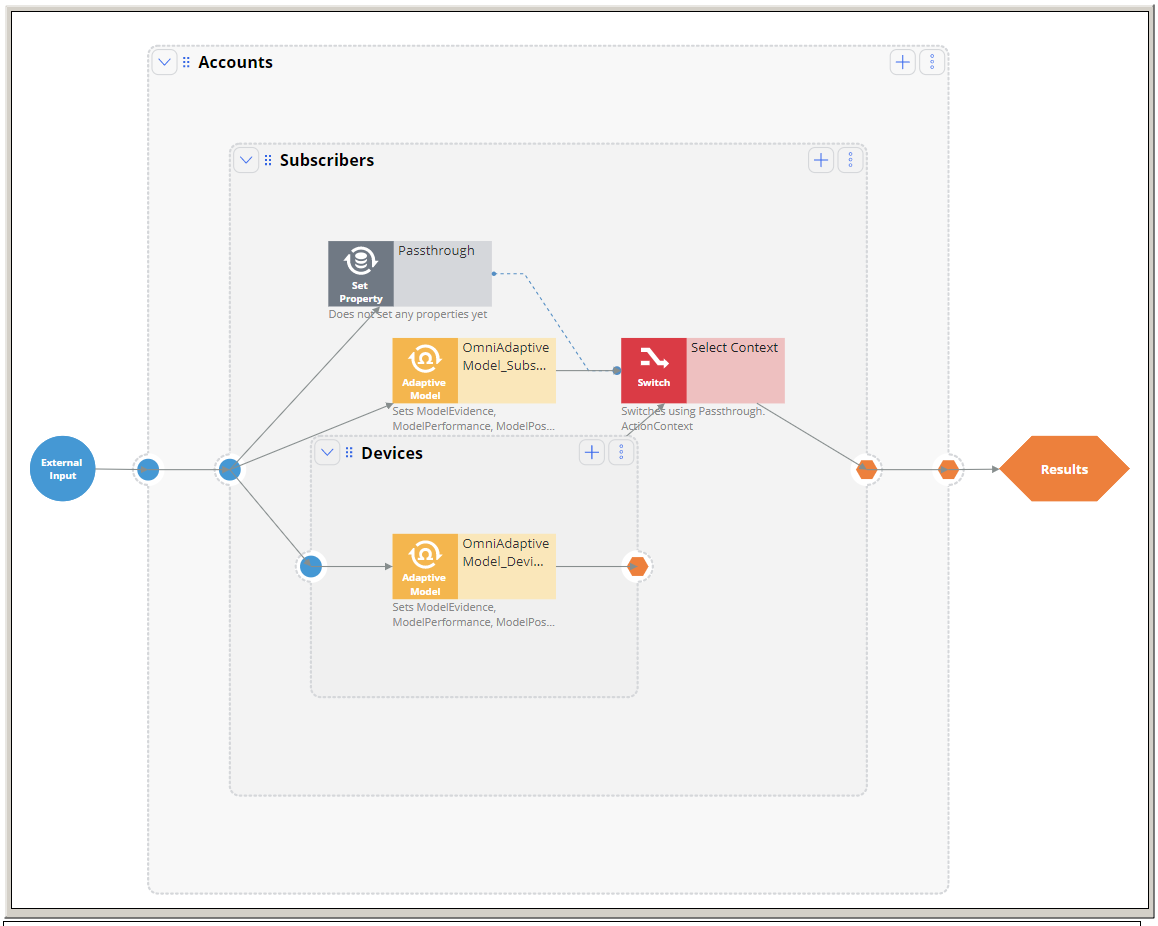
This strategy is generated by Next-Best-Action Designer based on the context dictionary configuration. It may be overtaken and modified if required, but if so any future changes to the context dictionary may need to be applied to the strategy manually.

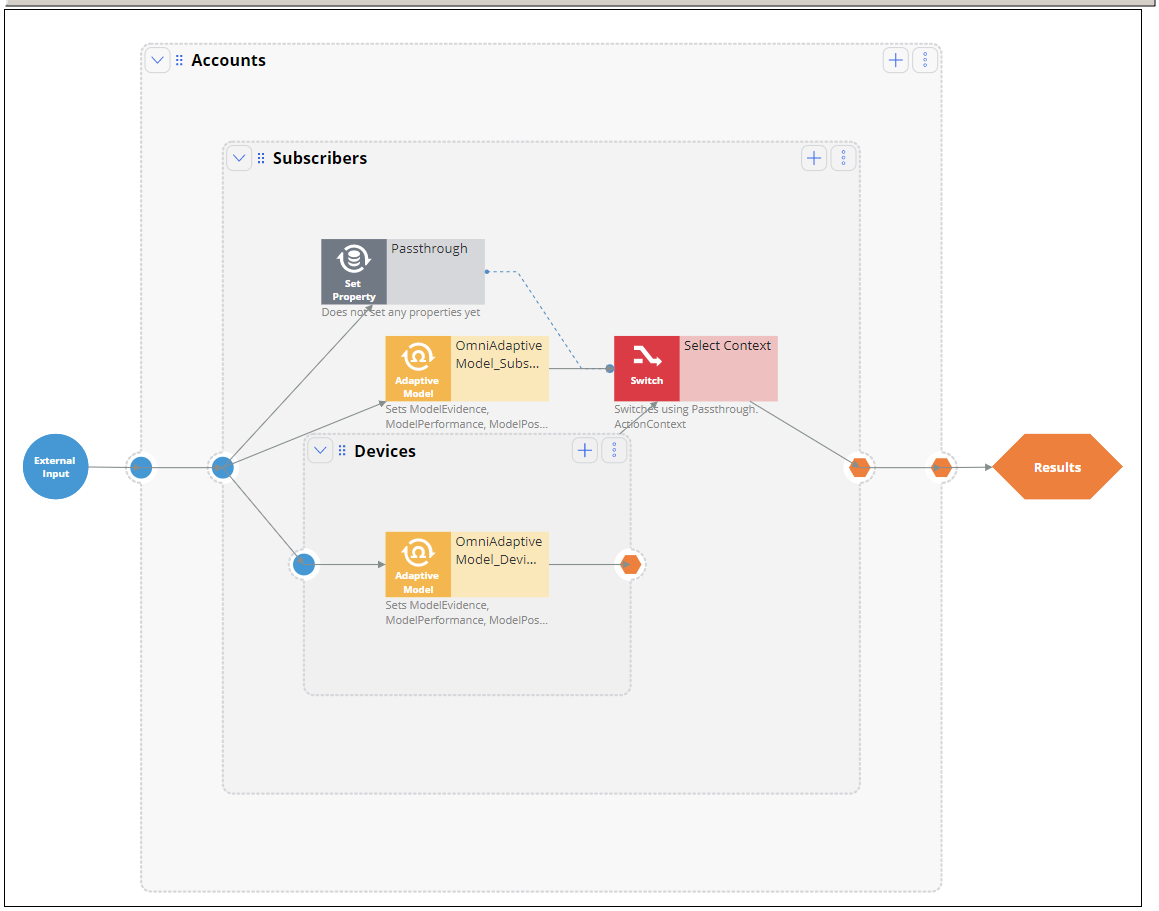
The **Passthrough** shape is an empty component connected to the input point of the primary context shape; it is needed so that the Select Context switch rule can test the account context to determine which context model is required, that is, whether a Subscriber or Device model is needed depending on the action context being processed. Without this switch, the logic will iterate over all contexts for all actions which will give incorrect results and execute unnecessary models.

The primary context (Subscribers) shape includes input conditions to ensure that only the current contact (*OriginalContactID*) is processed.

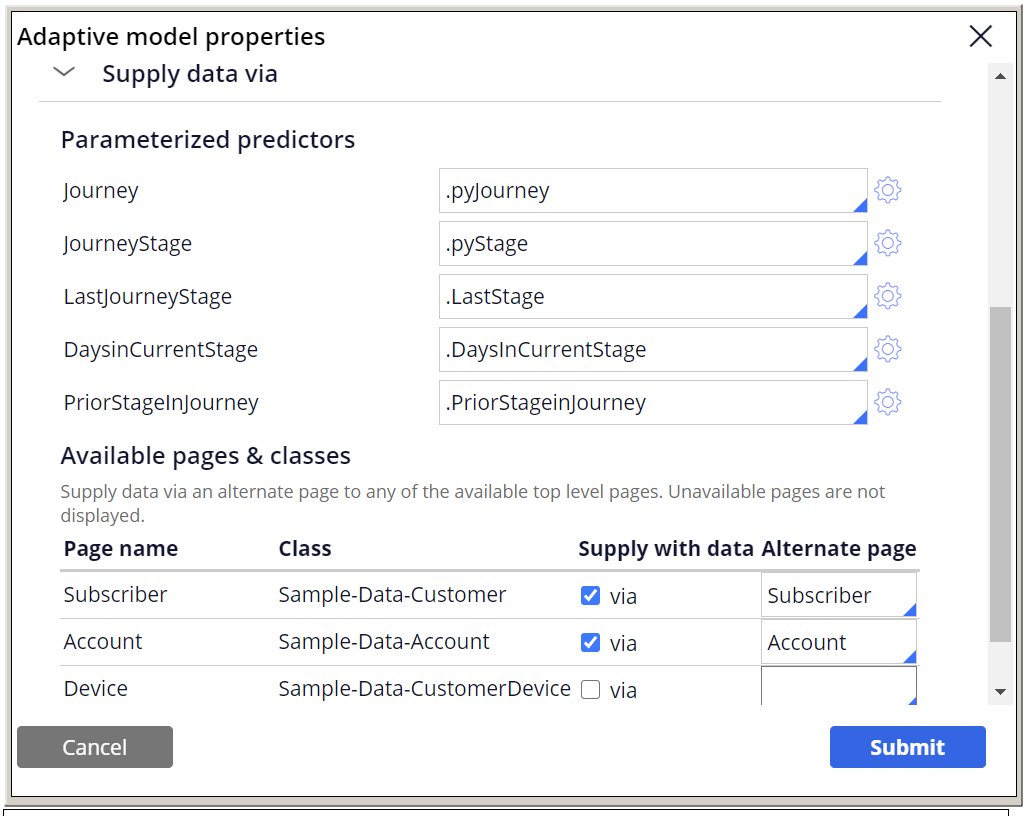
The Devices shape includes input conditions to ensure that only the current Device (*OriginalSubjectID*) is processed.

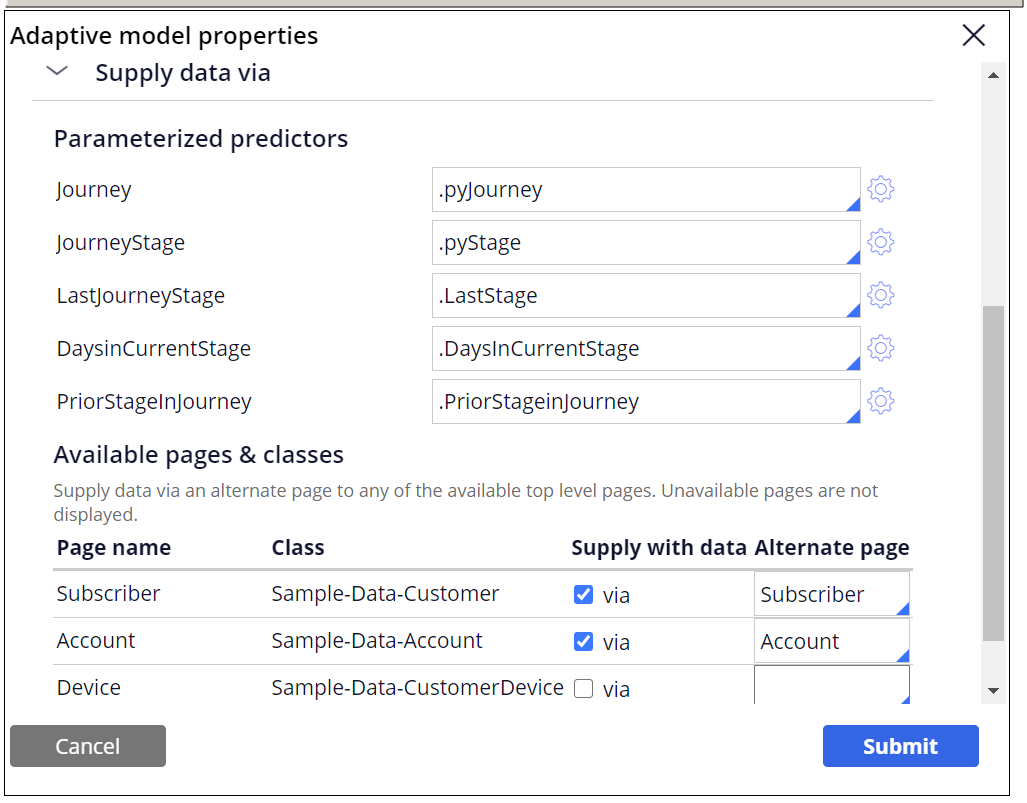
The same OmniAdaptiveModel is used for all action contexts and is applicable to all actions and channels, and the context for this model is issue, group, name, channel, and direction.



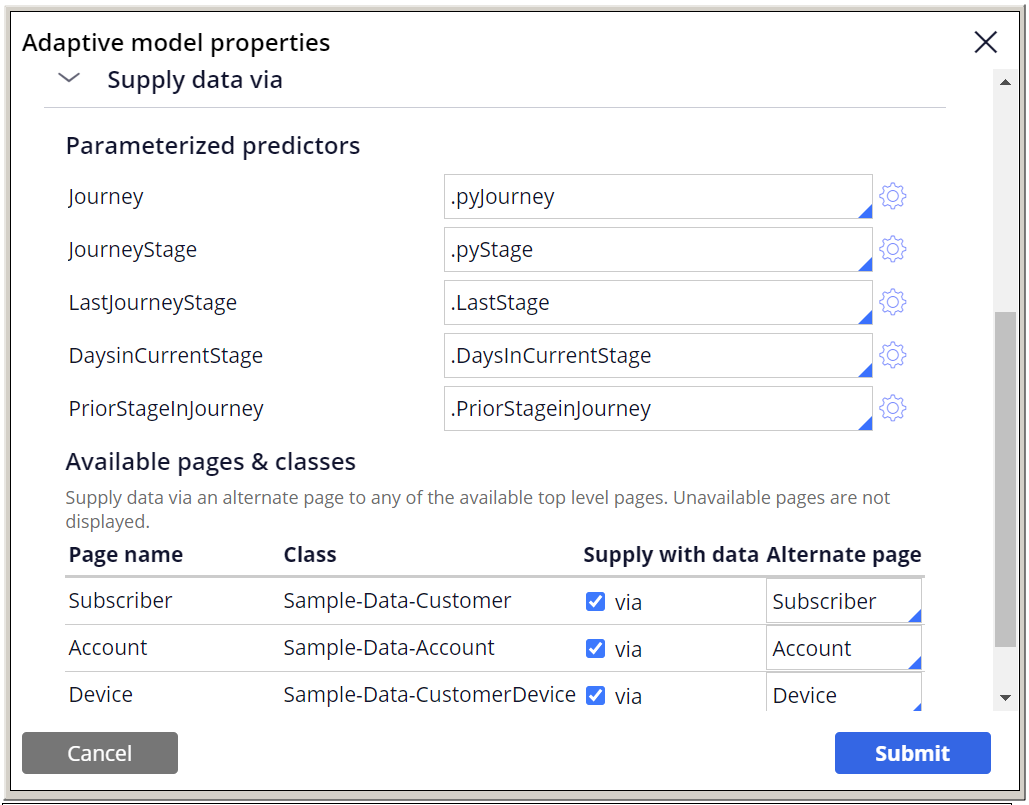


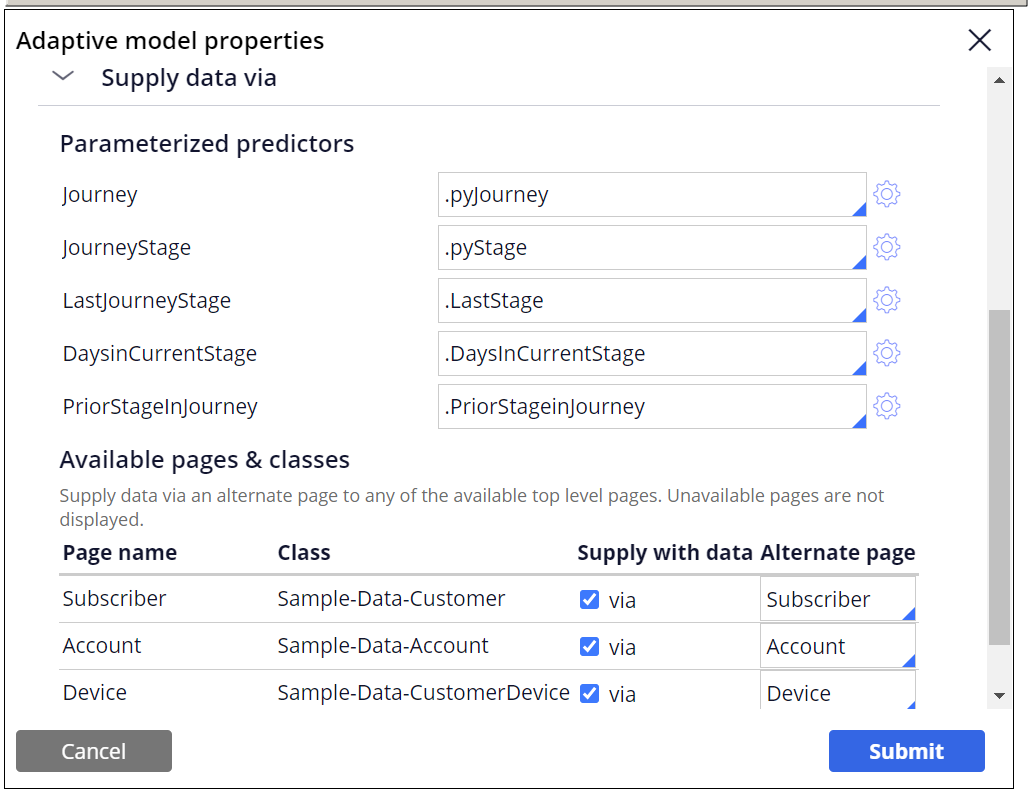
The model definition contains some parameterized predictors for Journey Stage data and Pages & Classes entries for each context entity. The shape in the Subscribers context passes the Account and Subscriber pages to the model, as in the following figure:





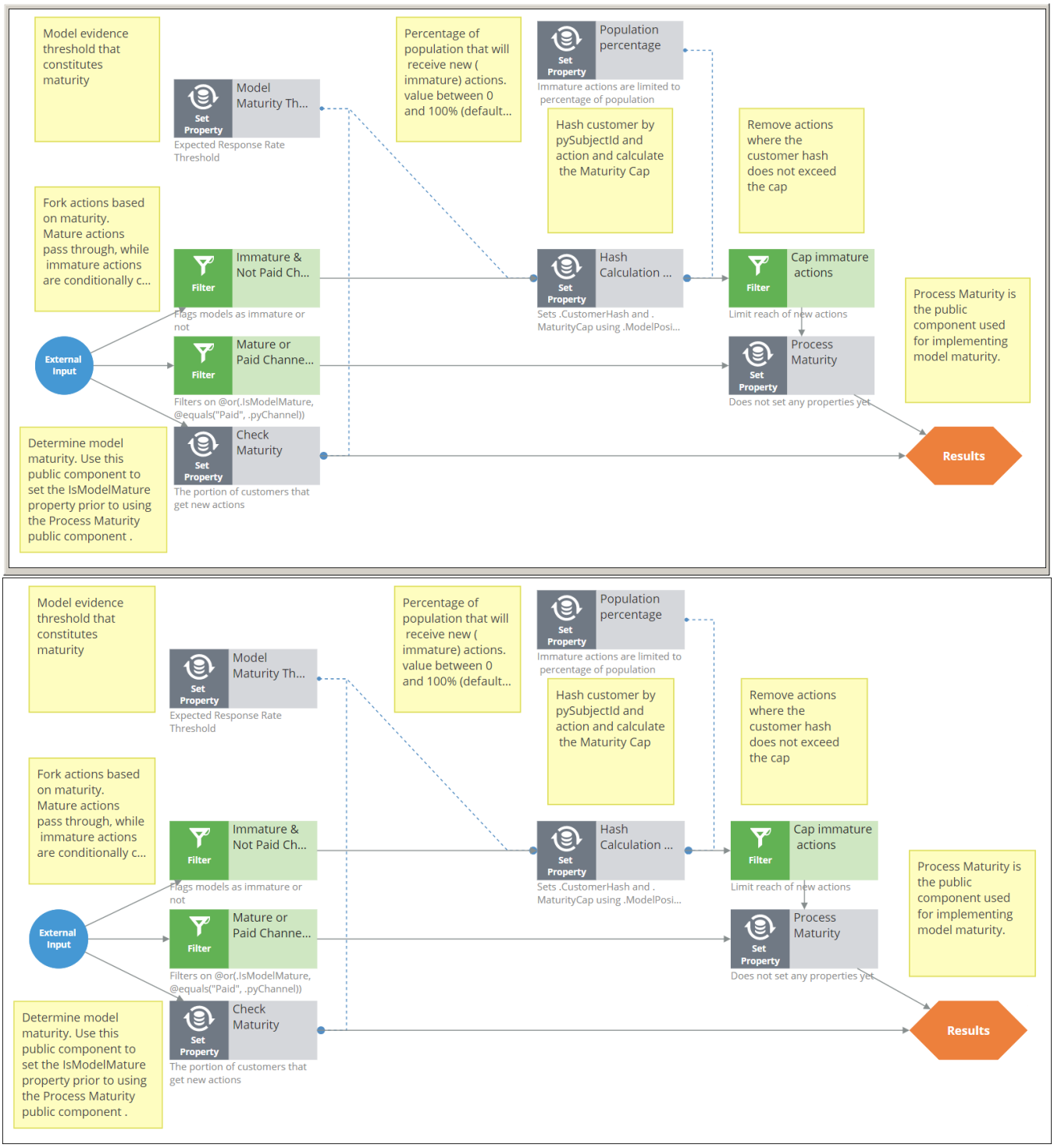
The shape in the Devices context passes Account, Subscriber and Device pages, as in the following figure:





Handle Model Maturity

The purpose of this strategy is to exclude actions (and treatments at a later stage) that have recently been introduced and for which there is little evidence on which the models can be trained.



This strategy has two public components:

Check Maturity
: Determine model maturity. Use this public component to set the IsModelMature property prior to using the Process Maturity public component.

Process Maturity
: Is the public component used for implementing model maturity.

Actions that are deemed to be already mature (or are enabled for Paid Media) bypass this process through the Mature or Paid Channel Actions filter. For actions that are less mature (and not enabled for Paid Media), an analytical method is used to filter out a portion based on their maturity (the less mature, the higher the proportion).

Apply Propensity Thresholds

This strategy imports propensity thresholds from the Model Propensity Thresholds DDR and matches them to Actions by Direction, Channel, All Issues, Issue and All Groups, Group and All Actions, Action Name and All Treatments or by Treatment. More specific threshold definitions take precedence over less specific ones, that is, a threshold set at the group level will override one set at the issue level.

The DDR Direction may be Inbound, Outbound or blank for either.

The DDR Channel may be blank for all channels, but if any specific channels are defined, then ALL channels must be defined, since blank channels will be ignored.

Actions with *FinalPropensity* less than the threshold will be excluded from the output (hence *FinalPropensity* must be set prior to calling this strategy).



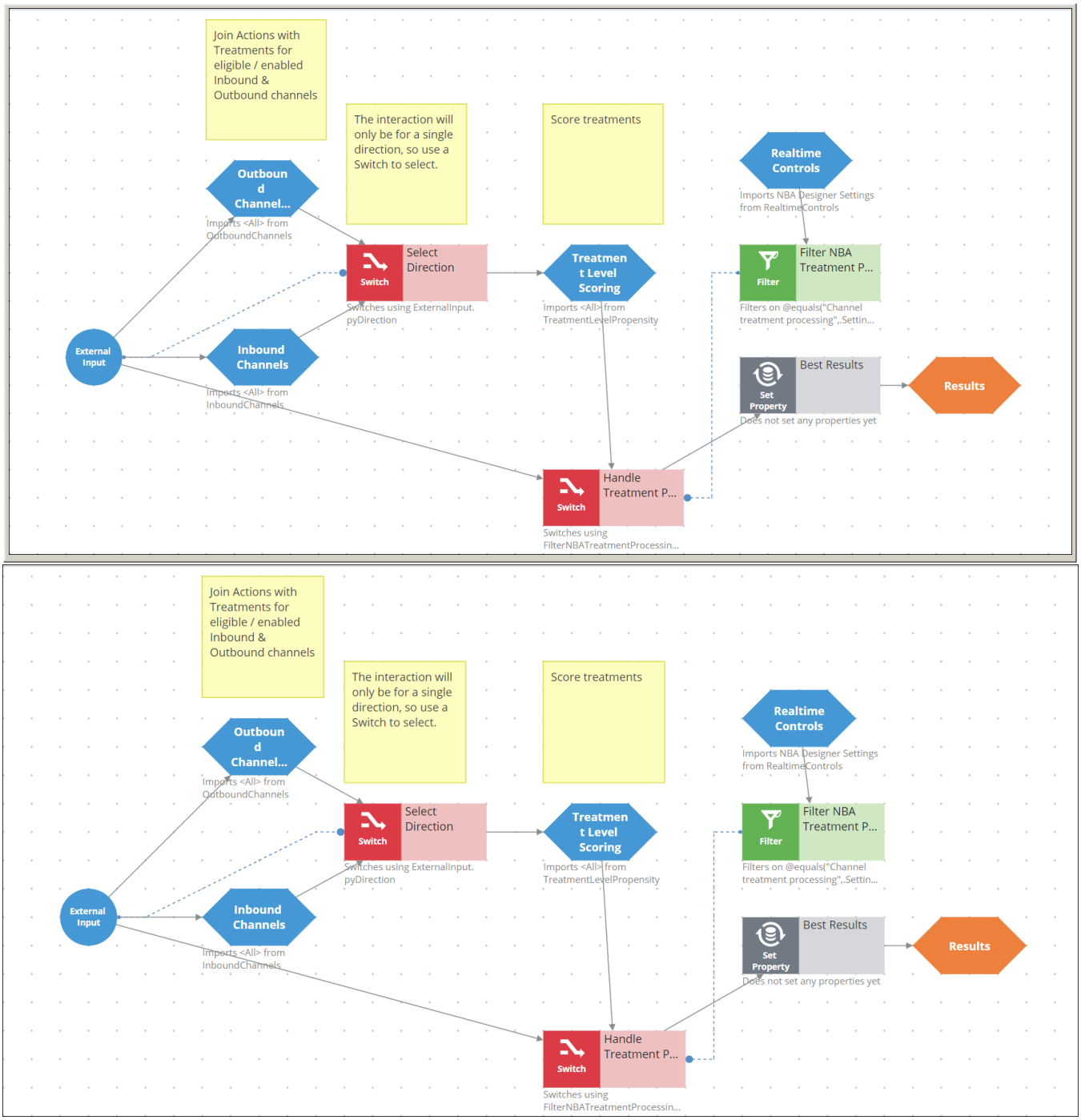
For a detailed description of the logic, refer to the annotations on the strategy.

Treatments and channels

The Treatments & Channels strategy first checks if treatment processing is enabled based on the **Channel treatment processing** real-time control setting in the *Handle Treatment Processing Switch* rule. If treatment processing is disabled, actions are not joined with their associated treatments, and treatment level analytics and other processes are bypassed.

Pega Customer Decision Hub

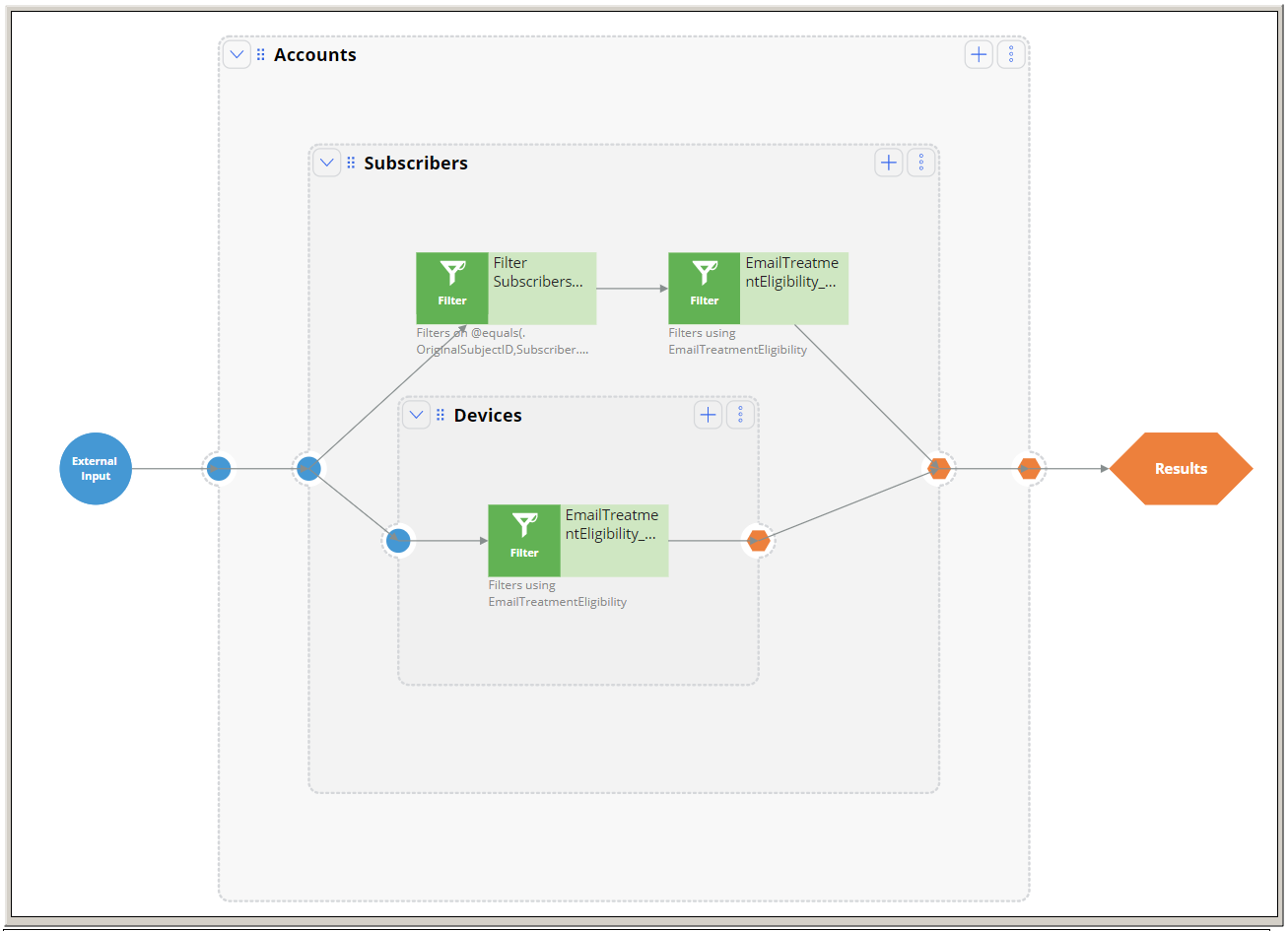
If Treatment Processing is enabled, a Switch rule selects whether to prepare outbound or inbound channels prior to treatment-level scoring (only a single direction will be processed at a time).

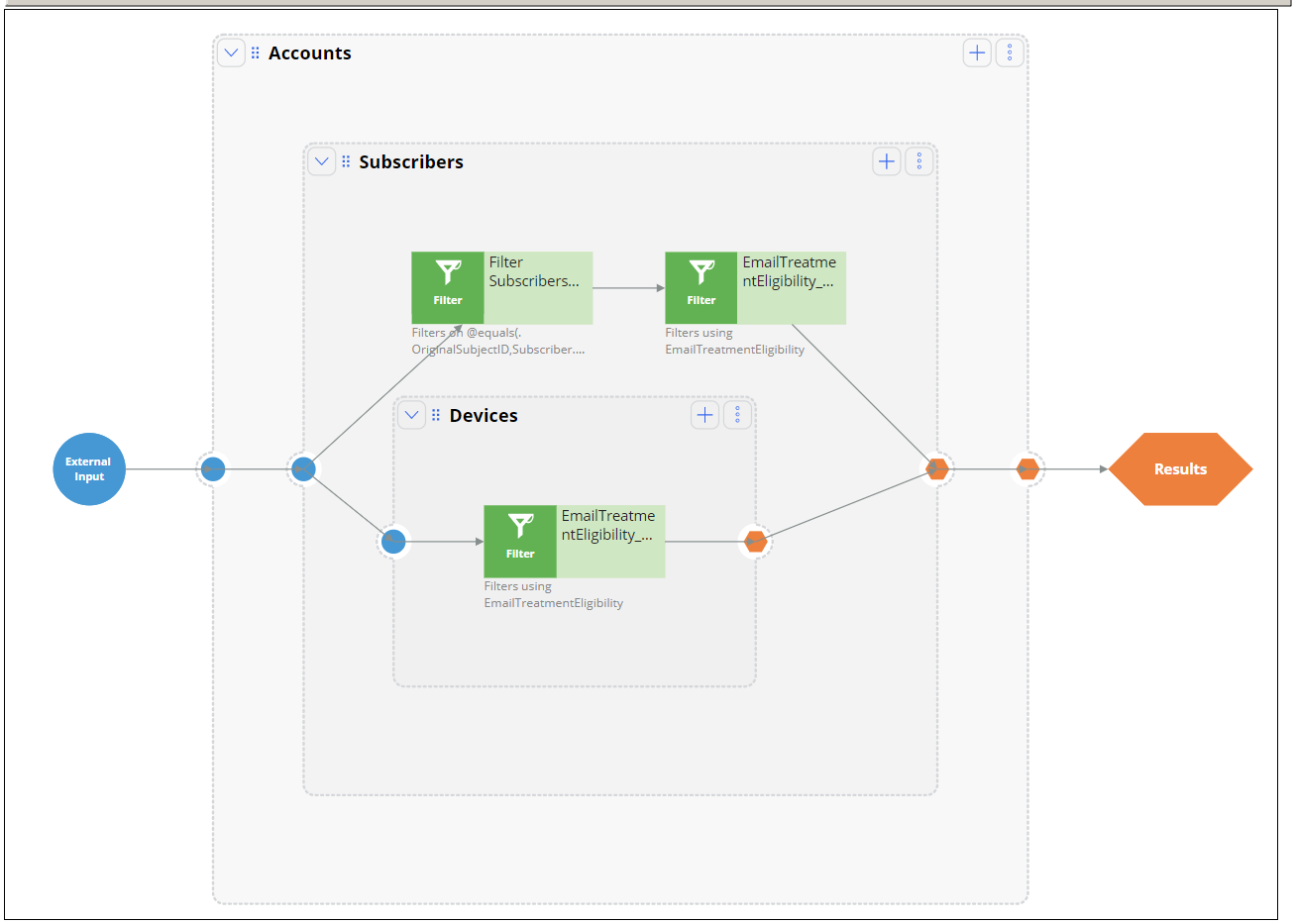


Channel Treatment Eligibility

A new feature from version 8.6 onwards is the introduction of treatment eligibility rules, which are implemented through Proposition Filters.

Each of the individual outbound and inbound channel strategies calls a separate sub-strategy to evaluate these. The sub-strategies are generated by Next-Best-Action Designer based on the context dictionary configuration and cannot be modified.





The embedded context sub-strategies have input conditions that ensure that the filters are only executed for the current contact and context.

- Outbound Channels
- Inbound Channels

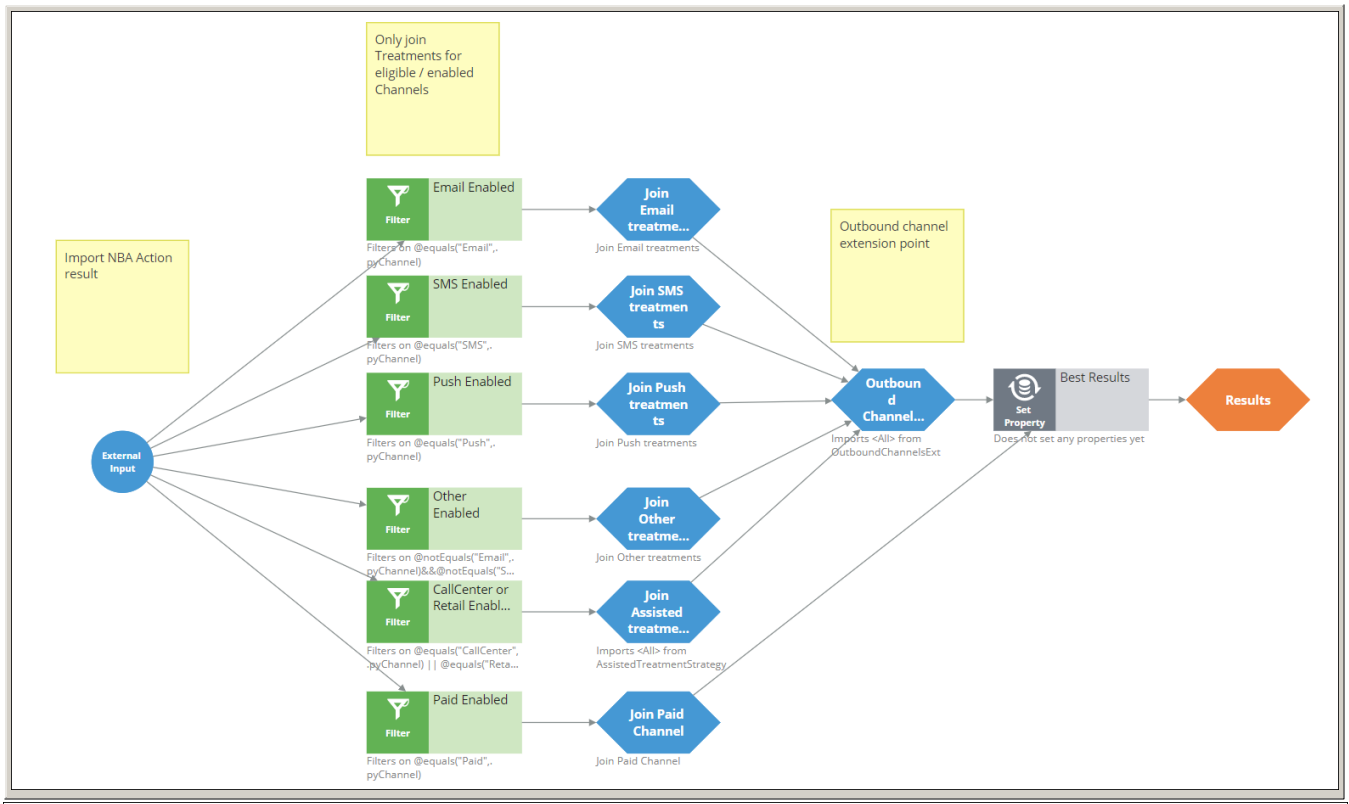
Outbound Channels

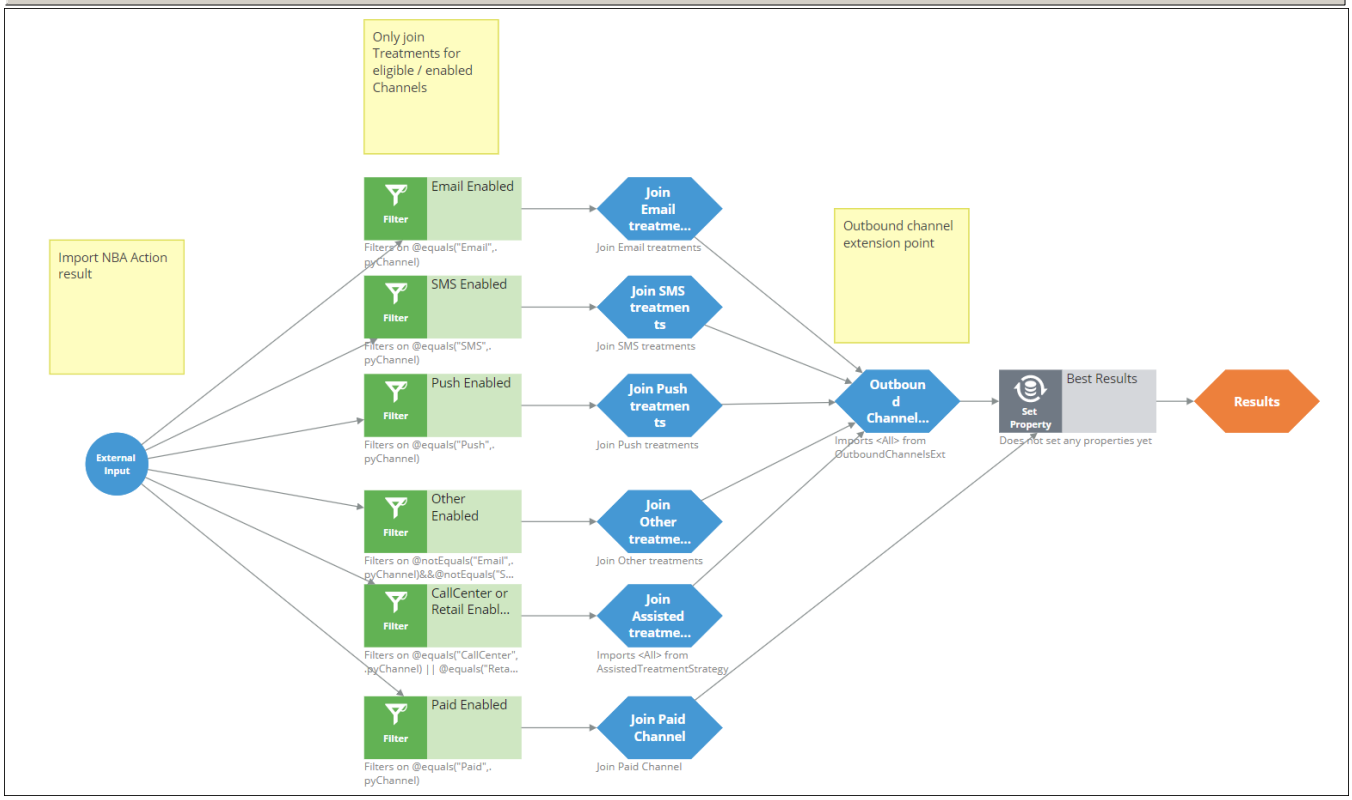
The Outbound Channels sub-strategy links actions with the treatments for all of the outbound channels; a separate sub-strategy is called for each of Email, SMS, Push, Paid, Assisted and Other (custom channels).

Pega Customer Decision Hub

The extension strategy *OutboundPreferencesExtension* is available to implement any customer preference filtering, for example, that the customer does not wish to be contacted through SMS. This is a placeholder strategy and needs to be implemented if you wish to take advantage of this.

This strategy only applies if the primary direction is Outbound or if the direction has not been specified (which implies it is a batch run). The individual channels must also be active and enabled on each action.





All of the individual channel sub-strategies – other than Paid – use the same method for assigning treatments to actions and augmenting the treatment data. They link actions with their channel specific treatments from the *<Channel>TreatmentsJoin* DDR (where *<Channel>* represents the name of the channel). As a result, the *pyTreatment* value is set on the action; if there are multiple treatments a separate action page is generated for each treatment. The *pyTreatment* property can be used in the action flow canvas to dynamically set the treatment name. This allows you to define a template flow and apply it to the action.

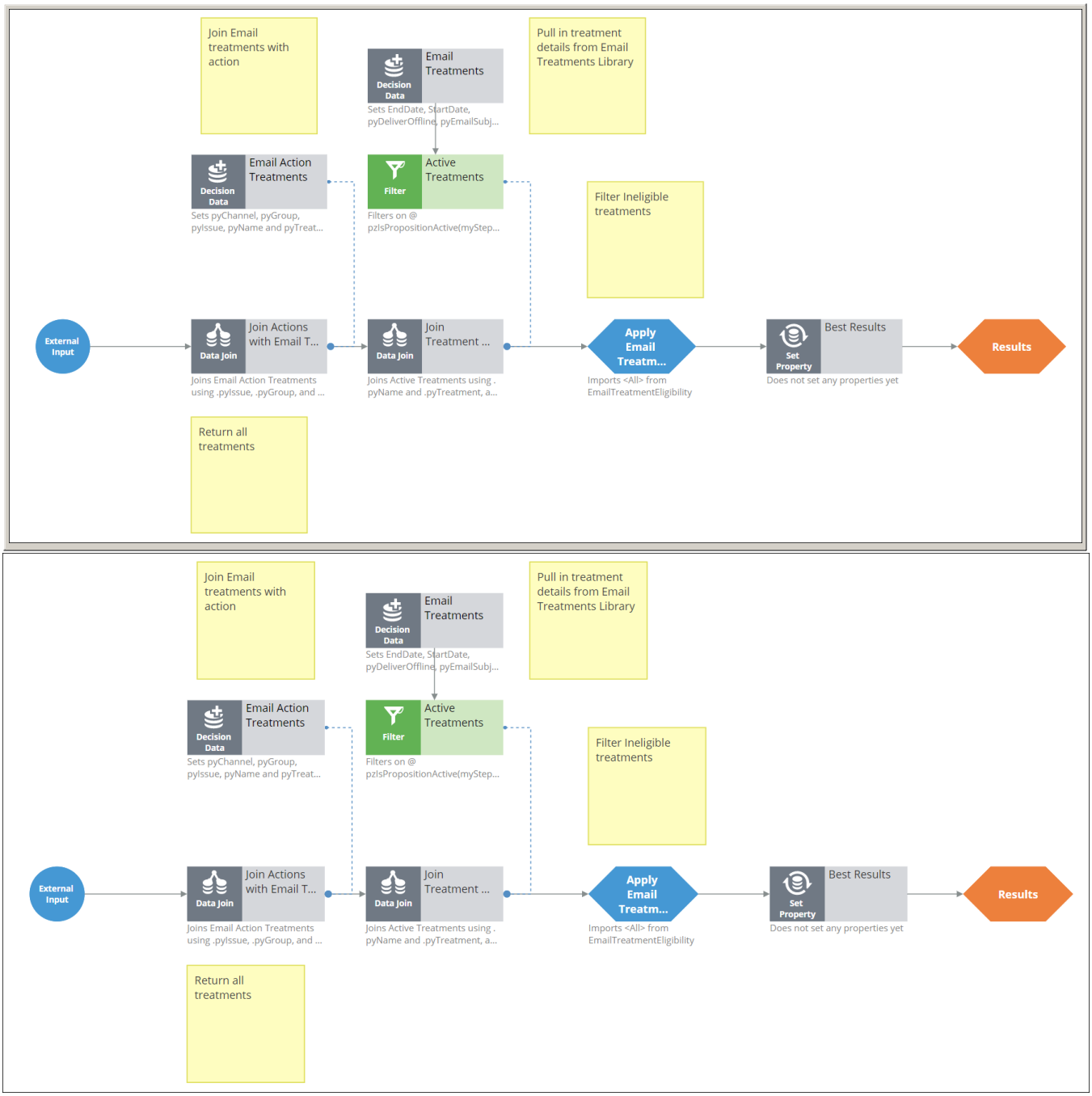
Additional Treatment properties are obtained from the *<Channel>Treatments* DDR and joined to the actions, and then *Apply <Channel> Treatment Eligibility* calls the channel treatment eligibility rules sub-strategy.

These strategies may be overtaken and customized for local requirements if required, and the *<Channel>Treatments* DDR may be extended to augment the treatment data. For more information, see Managing Decision Data rules in Pega Customer Decision Hub.

The individual channel strategies are described in more detail below.

Email, SMS and Push channel strategies

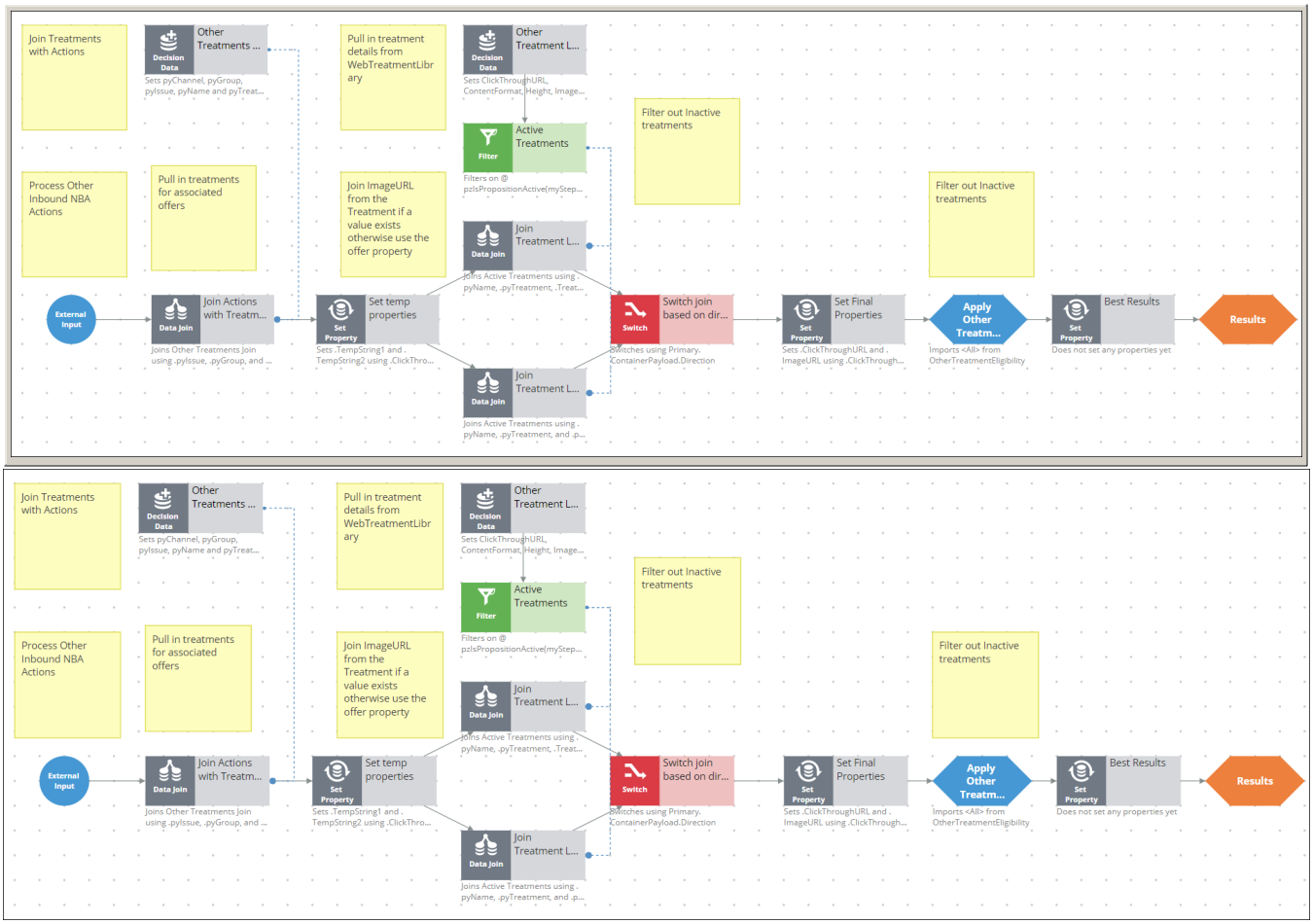
The *EmailChannel* strategy links actions with email treatments from the *EmailTreatmentsJoin* DDR. As a result, the *pyTreatment* value is set on the action. The *pyTreatment* property can be used in the action flow canvas to dynamically set the treatment name. This allows you to define a template flow and apply it to the action.



Other treatment strategy

This strategy performs the basic logic for assigning treatments, augmenting treatment data and evaluating treatment eligibility, but with some minor additions.

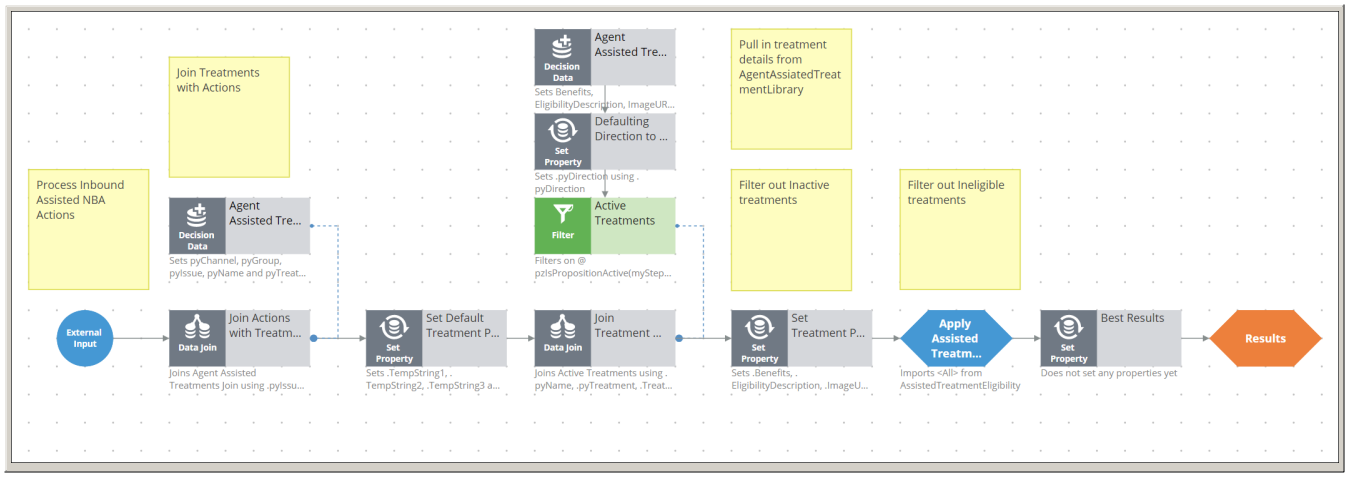
Some of the individual metadata values that are defined at the treatment level will override their action-level equivalents. Also, since this strategy is used for both Outbound and Inbound interactions, the join conditions for the DDR are slightly different for each and are selected via the Switch join based on direction Switch rule.

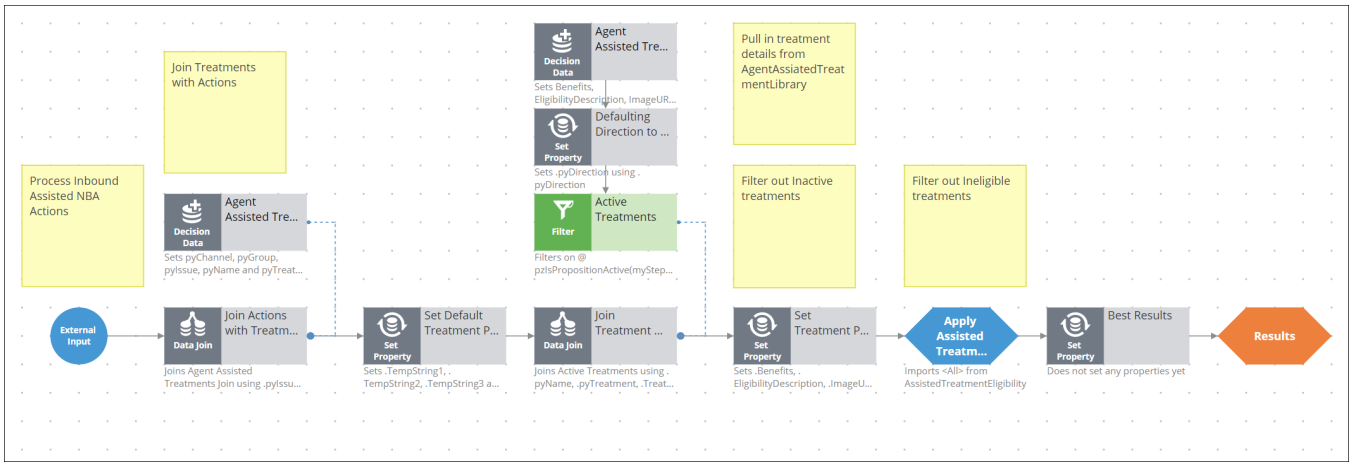


Assisted channel strategy

This strategy performs the basic logic for assigning treatments, augmenting treatment data and evaluating treatment eligibility, but with some minor additions.

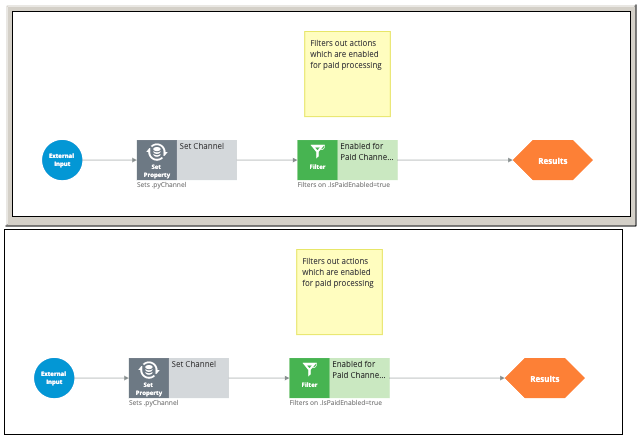
This strategy is used for both Outbound and Inbound interactions, and pyDirection on the Assist Treatments DDR is defaulted to Inbound if not already set. Also, some of the individual metadata values that are defined at the treatment level will override their action-level equivalents.





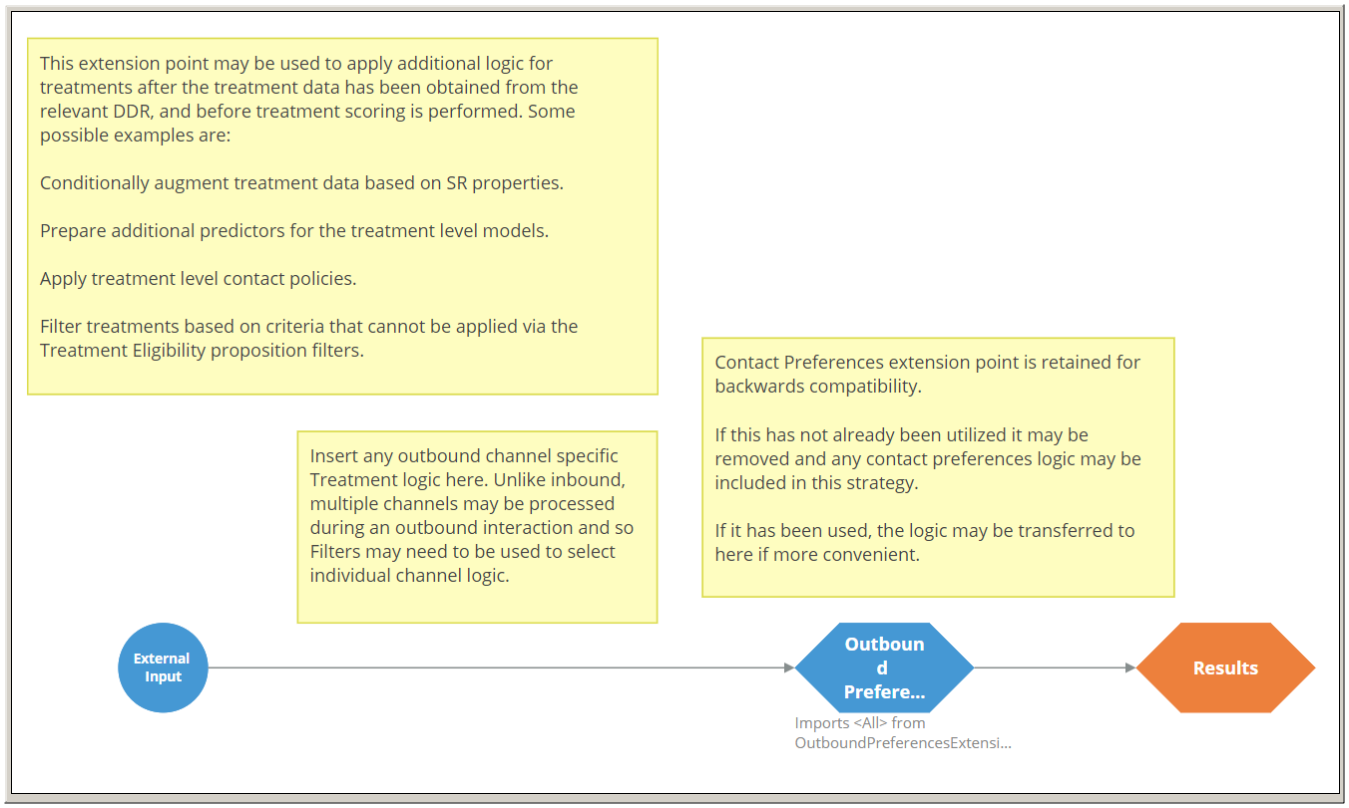
Paid channel strategy

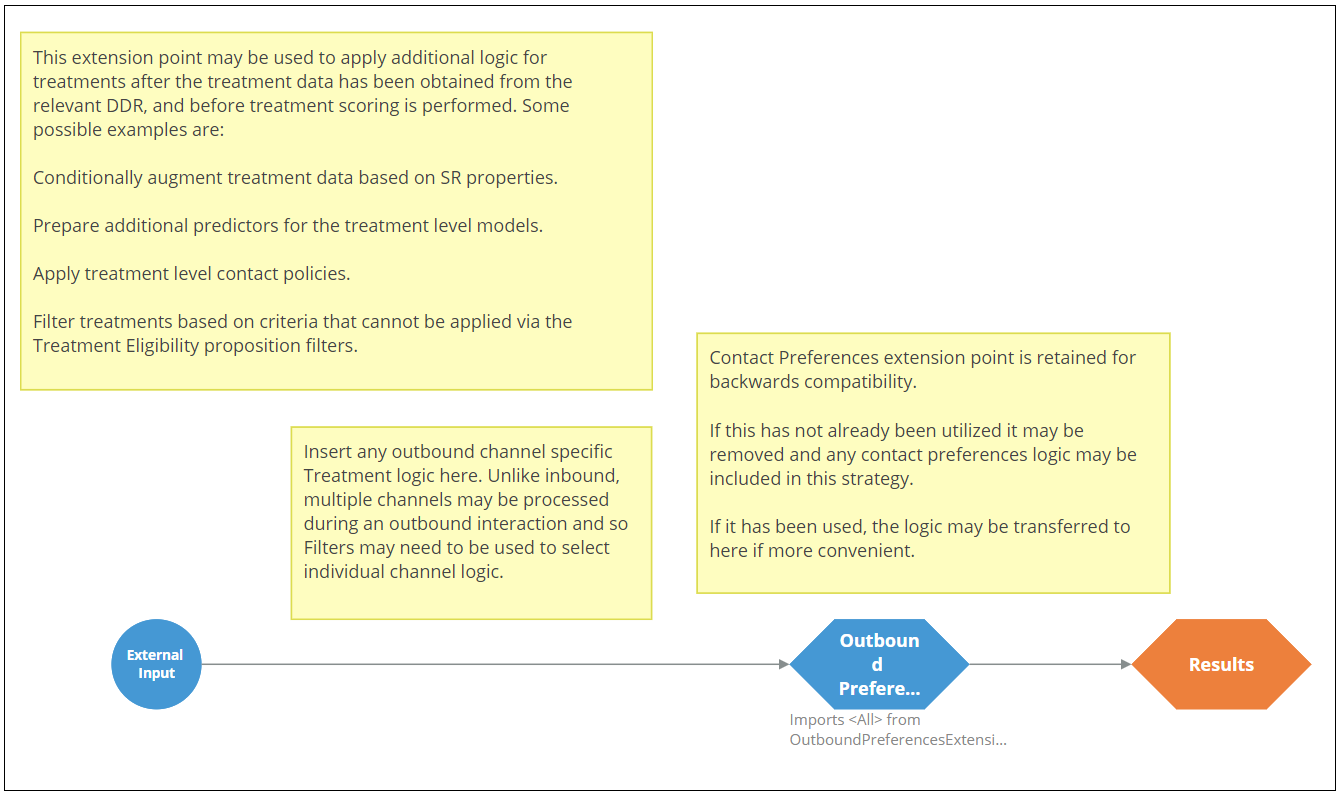
This strategy is an empty canvas and allows you to implement additional channel-specific logic as required.



Outbound Channel Extension

This sub-strategy is a placeholder strategy to implement any customer preferences. It contains a single extension point sub-strategy *OutboundPreferencesExtension* which contains only a blank canvas.



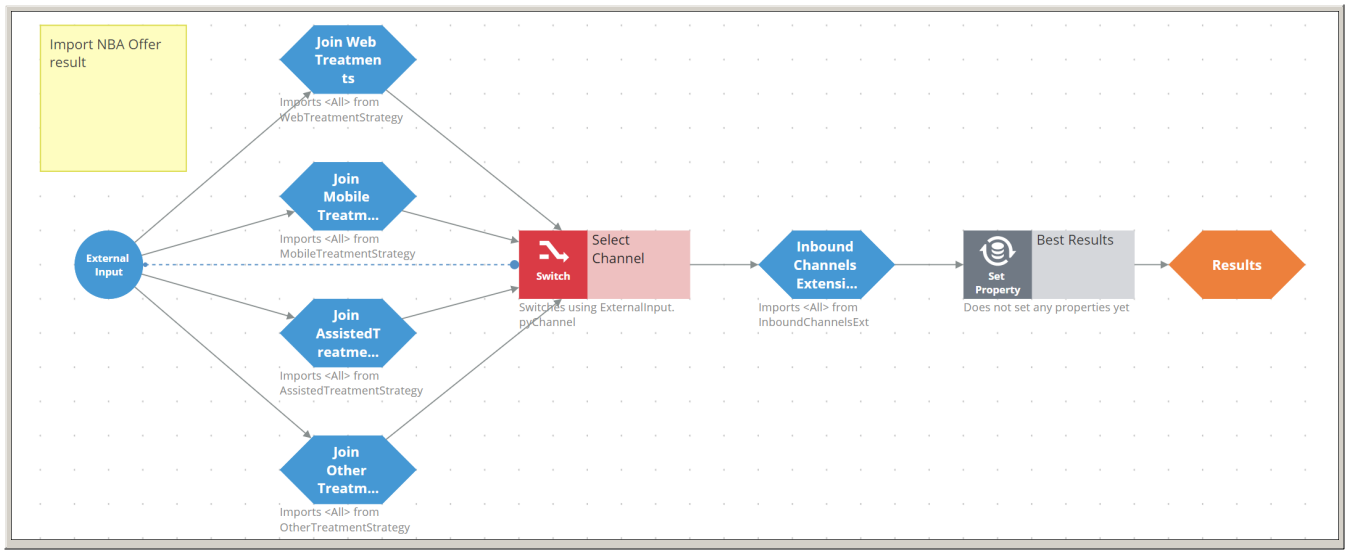


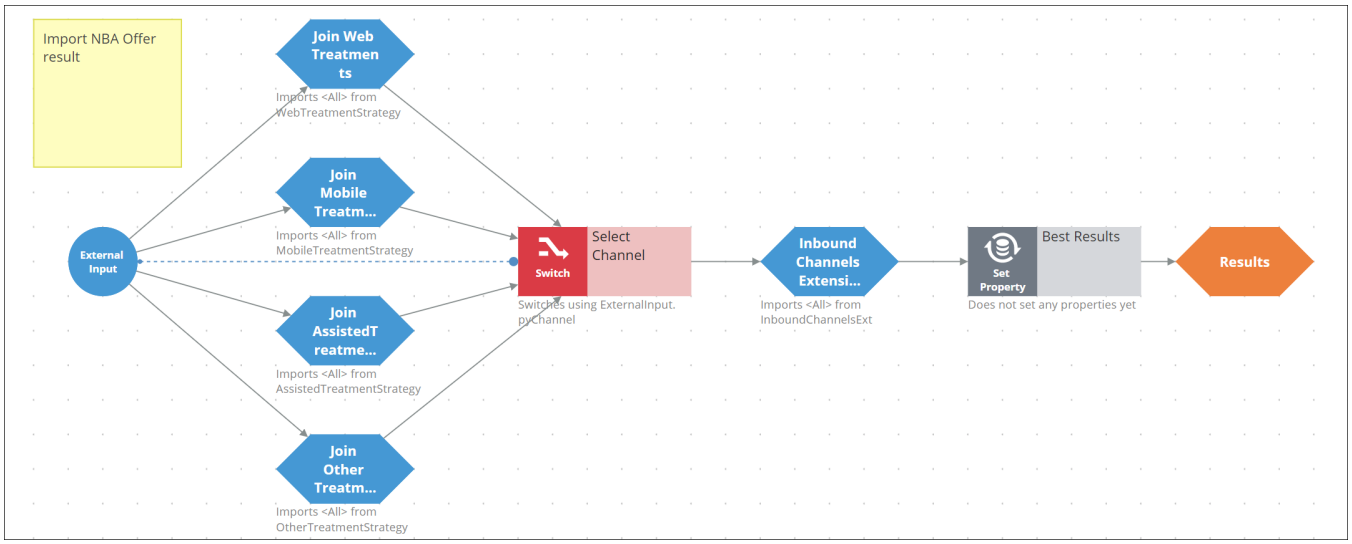
Inbound Channels

The *InboundChannels* sub-strategy links actions with the treatments for all of the inbound channels; a separate sub-strategy is called for each of Web, Mobile, Assisted (Call Center and Retail), and Other (custom channels).

Pega Customer Decision Hub

This strategy only applies when the primary direction is inbound. The individual channels must also be active and enabled on each action.





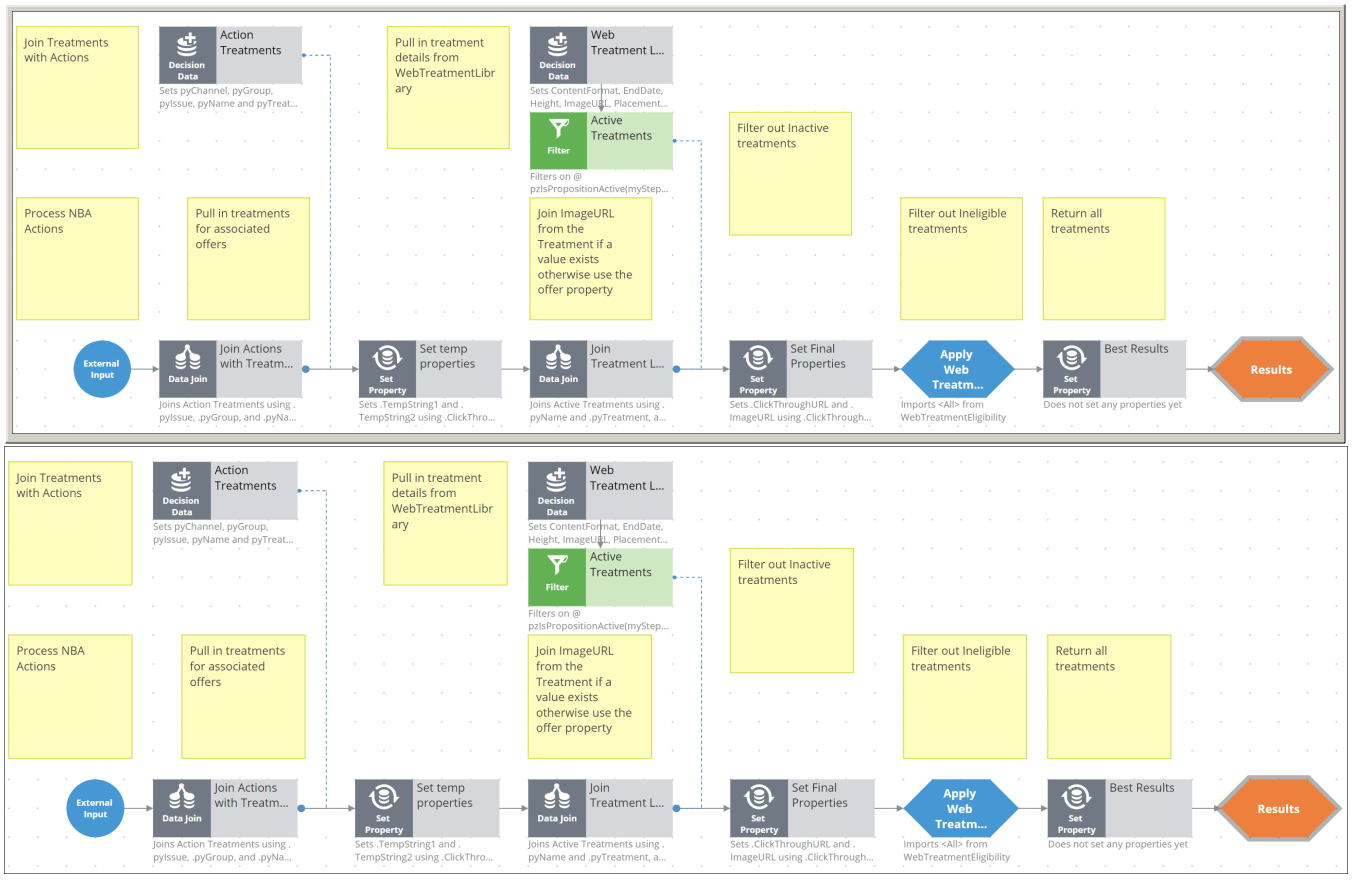
The strategies for processing Assisted and Other channels are the same for both oubound and inbound interactions and have already been described in Outbound Channels.

Web and Mobile treatment strategies

Both of these sub-strategies – named *<Channel> Treatment Strategy* (where *<Channel>* represents the name of the channel) - use the same method for assigning treatments to actions and augmenting the treatment data. They link actions with their channel specific treatments from the *<Channel>TreatmentsJoin* DDR and as a result, the *pyTreatment* value is set on the action; if there are multiple treatments a separate action page is generated for each treatment.

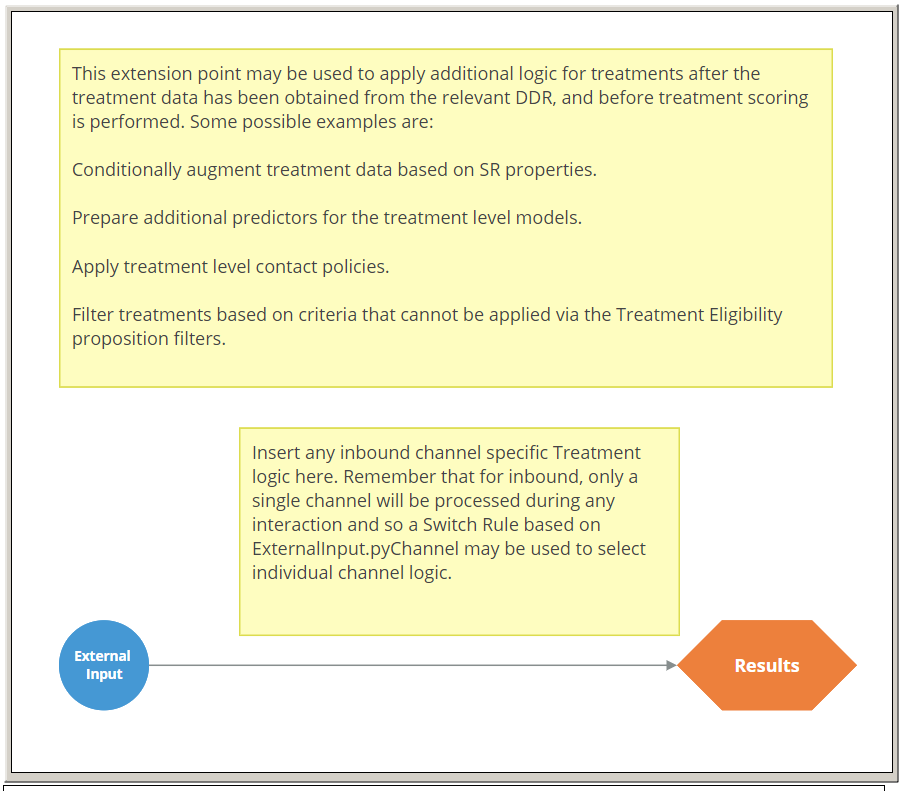
Additional Treatment properties are obtained from the *<Channel>Treatments* DDR and joined to the actions; individual metadata values that are defined at the treatment level will override their action-level equivalents. Finally, *Apply <Channel> Treatment Eligibility* calls the channel treatment eligibility rules sub-strategy.

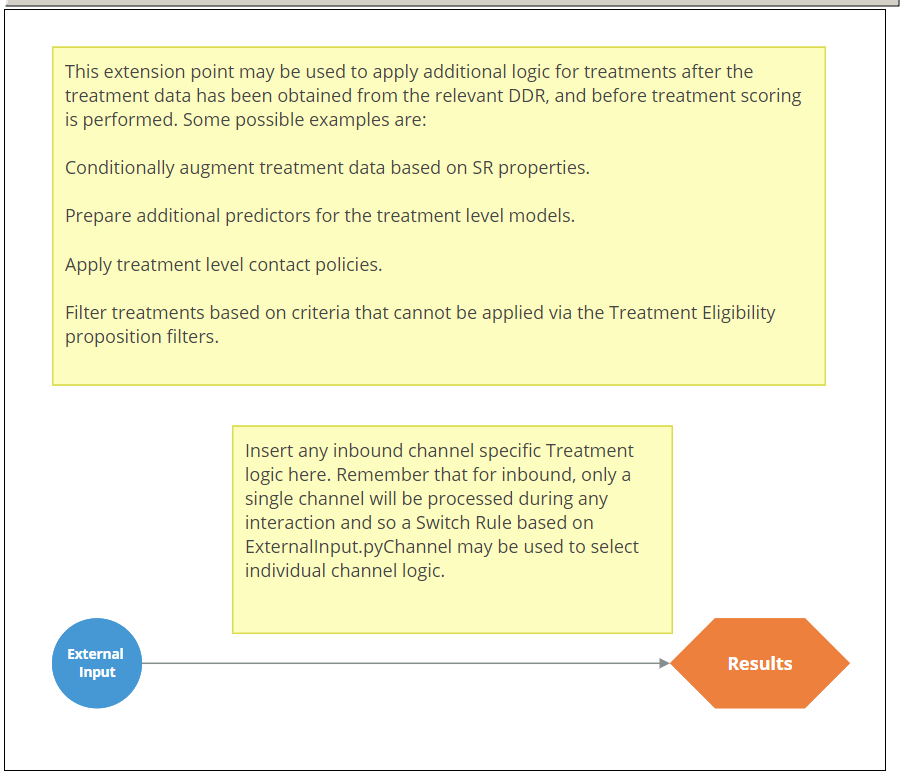
These strategies may be overtaken and customized for local requirements if required, and the *<Channel>Treatments* DDR may be extended to augment the treatment data. For more information, see Managing Decision Data rules in Pega Customer Decision Hub.



Inbound Channels Extension

This sub-strategy is a placeholder strategy to implement any inbound customer preferences. It contains only an annotated blank canvas.





For a detailed description of the logic, refer to the annotations on the strategy.

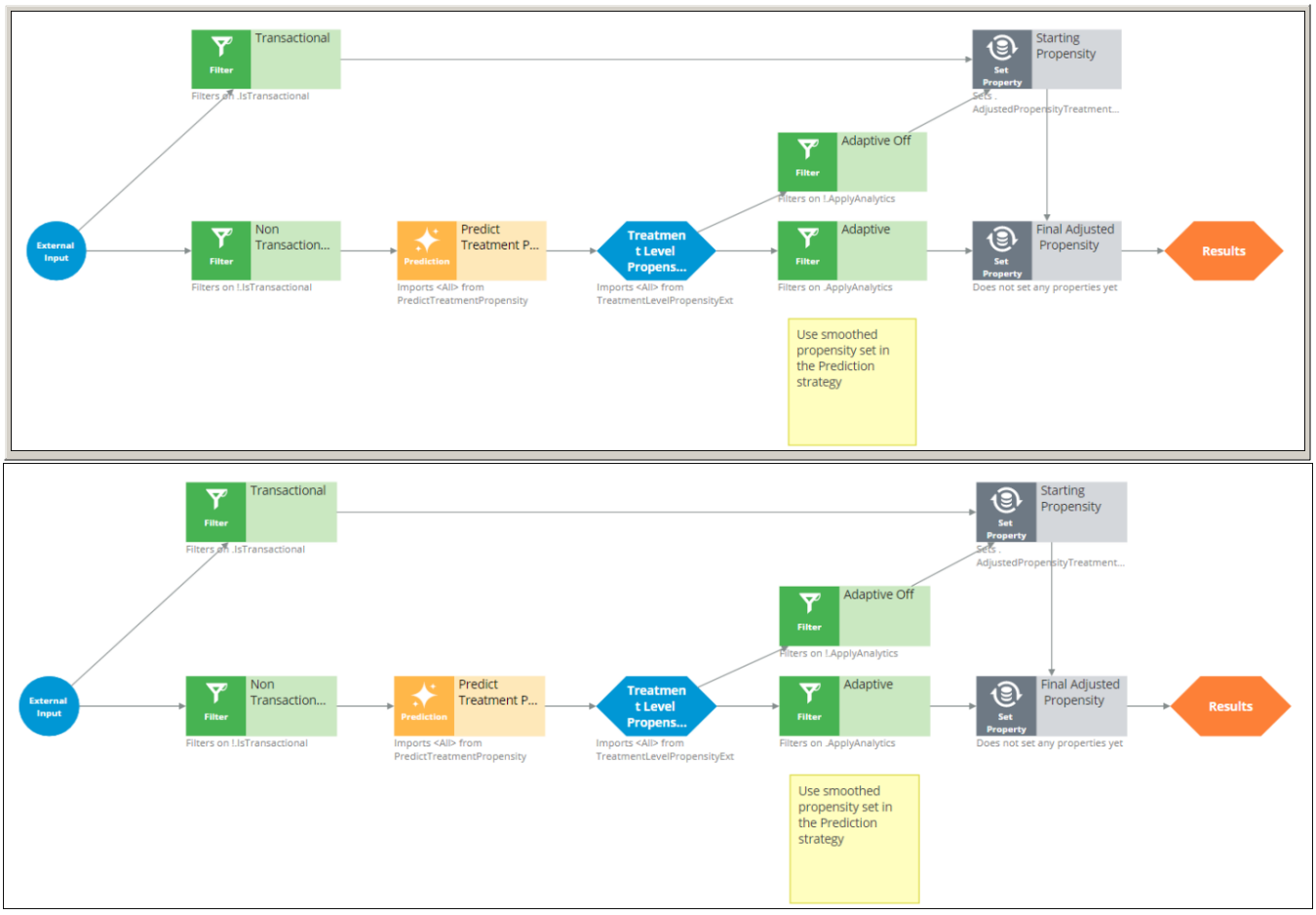
Treatment-level scoring strategy

The *TreatmentLevelPropensity* strategy is responsible for applying an adaptive model for each treatment and calculating the propensity across eligible channels.

Actions can be designated as transactional in nature, that is, they are always intended to be delivered and do not require any arbitration. Treatments for transactional actions are filtered out and bypass analytics processing.

Treatments for non-transactional actions are directed to the *Predict Treatment Propensity* Prediction shape, and then to the *Treatment Level Propensity Extension* extension point sub-strategy.

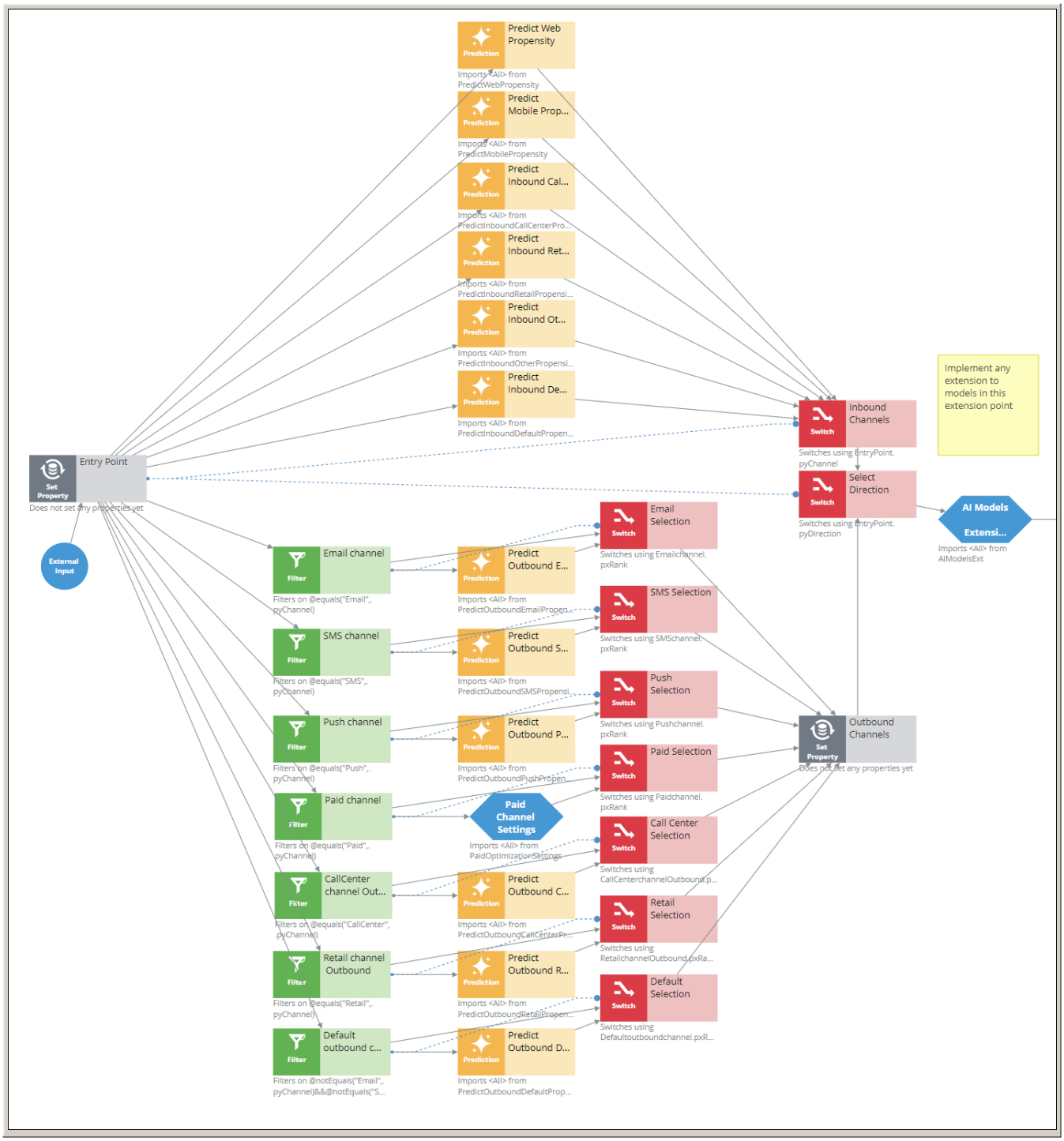
Finally, the actions are divided into adaptive and adaptive-off streams based on the *ApplyAnalytics* action property, and the adaptive-off stream sets the propensity based on the action starting propensity, whereas for the adaptive stream these values are calculated by the *Predict Treatment Propensity* strategy.

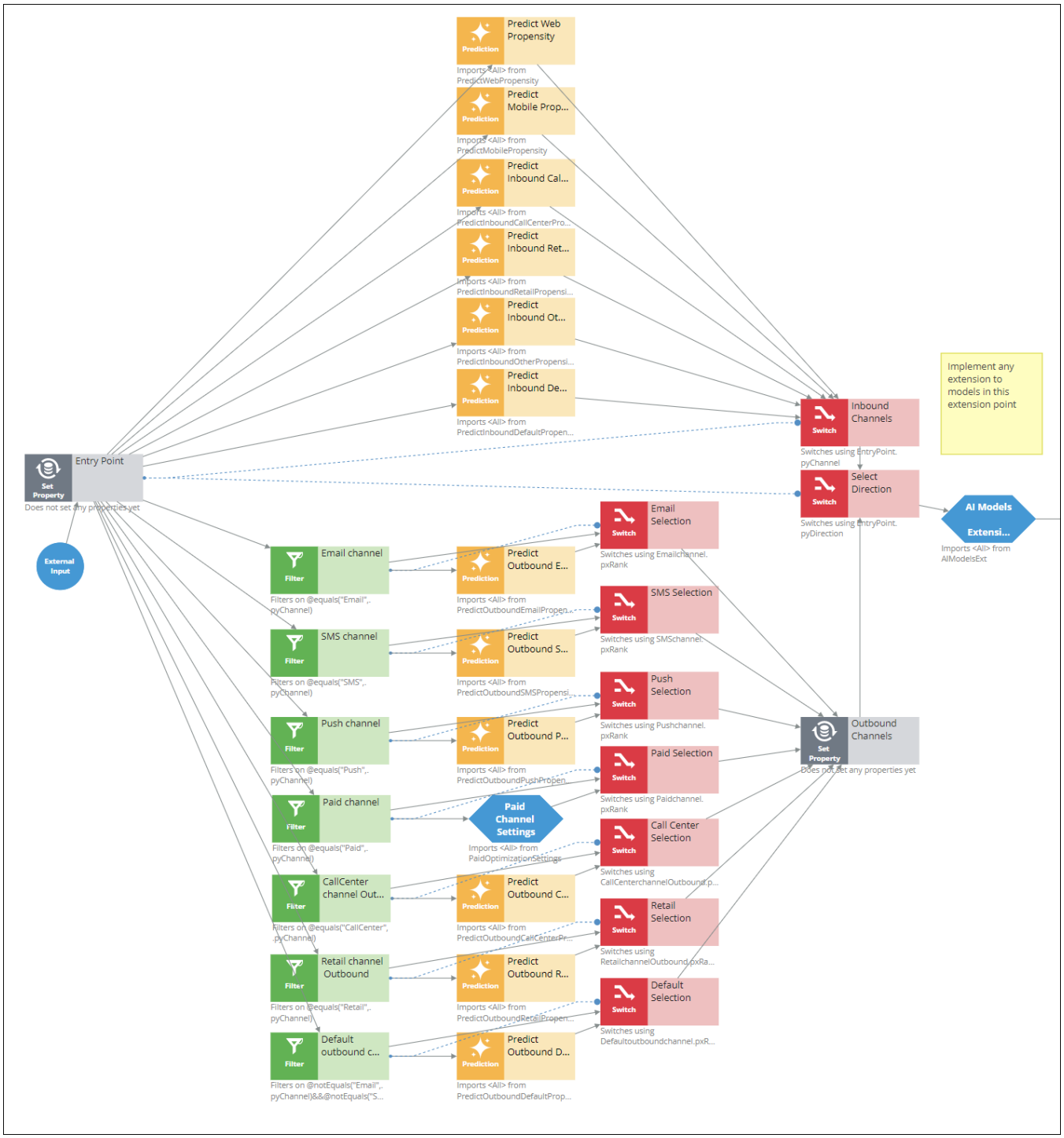


Predict Treatment Propensity

The *Predict Treatment Propensity* prediction strategy uses the outcome optimization approach which provides a different model configuration and outcome for each channel. This strategy separates treatments based on channel to ensure that the correct model is applied. As with the action level *Predict Action Propensity* strategy, Random Control Group, Outbound Model Maturity, and Thompson Sampling techniques are applied to optimize statistical robustness.

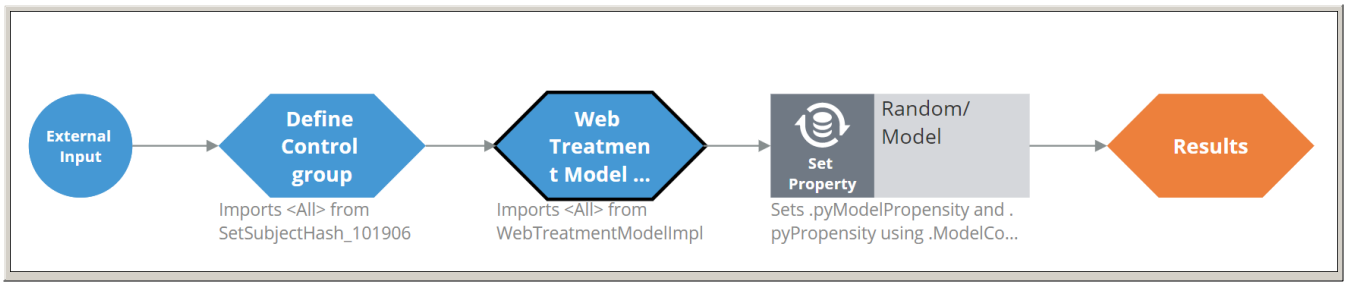
The strategy is shown below split into two sections for easier viewing; the first portion separates the treatments into inbound or outbound, and then further splits them based on channel before executing the appropriate Prediction strategy for the channel. They are then merged to the *AI Models Ext* extension point sub-strategy for any additional processing (if required) before proceeding further.

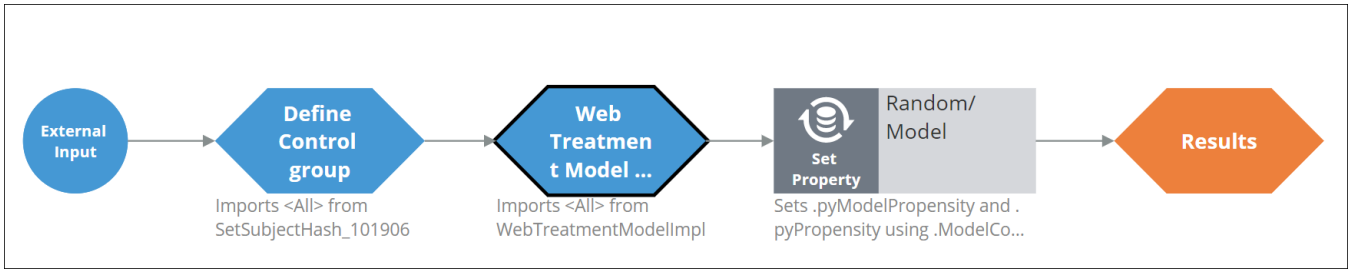

Predict Web Propensity
Imports <All> from PredictWebPropensity
Predict Mobile Prop...
Imports <All> from PredictMobilePropensity
Predict Inbound Cal...
Imports <All> from PredictInboundCallCenterPro...
Predict Inbound Ret...
Imports <All> from PredictInboundRetailPropensi...
Predict Inbound Ot...
Imports <All> from PredictInboundOtherPropensi...
Predict Inbound De...
Imports <All> from PredictInboundDefaultPropen...
Implement any extension to models in this extension point
Inbound Channels
Switches using EntryPoint.pyChannel
Select Direction
Switches using EntryPoint.pyDirection
AI Models Extensi...
Imports <All> from AIModelsExt
Entry Point
Does not set any properties yet
External Input
Email channel
Filters on @equals("Email",.pyChannel)
Predict Outbound E...
Imports <All> from PredictOutboundEmailPropen...
Email Selection
Switches using Emailchannel.pxRank
SMS channel
Filters on @equals("SMS",.pyChannel)
Predict Outbound S...
Imports <All> from PredictOutboundSMSPropensi...
SMS Selection
Switches using SMSchannel.pxRank
Push channel
Filters on @equals("Push",.pyChannel)
Predict Outbound P...
Imports <All> from PredictOutboundPushPropen...
Push Selection
Switches using Pushchannel.pxRank
Paid channel
Filters on @equals("Paid",.pyChannel)
Paid Channel Settings
Imports <All> from PaidOptimizationSettings
Paid Selection
Switches using Paidchannel.pxRank
Outbound Channels
Does not set any properties yet
CallCenter channel Out...
Filters on @equals("CallCenter",.pyChannel)
Predict Outbound C...
Imports <All> from PredictOutboundCallCenterPr...
Call Center Selection
Switches using CallCenterchannelOutbound.p...
Retail channel Outbound
Filters on @equals("Retail",.pyChannel)
Predict Outbound R...
Imports <All> from PredictOutboundRetailPropen...
Retail Selection
Switches using RetailchannelOutbound.pxRa...
Default outbound c...
Filters on @notEquals("Email",.pyChannel)&&@notEquals("S...
Predict Outbound D...
Imports <All> from PredictOutboundDefaultProp...
Default Selection
Switches using Defaultoutboundchannel.pxR...



A Switch rule selects whether inbound or outbound channels are required, and for inbound a further Switch rule selects the required channel. Remember that only one direction (inbound or outbound) is processed for any interaction, and for inbound interactions only a single channel is processed at a time, and so a Switch rule is the ideal selection mechanism for this.

Each channel except for Paid has a separate Prediction strategy assigned which sets the control group properties, calls the context level treatment model prediction for the channel, and then sets the propensity based on the control group settings. An example for the web channel is shown below (*Predict Web Propensity*).

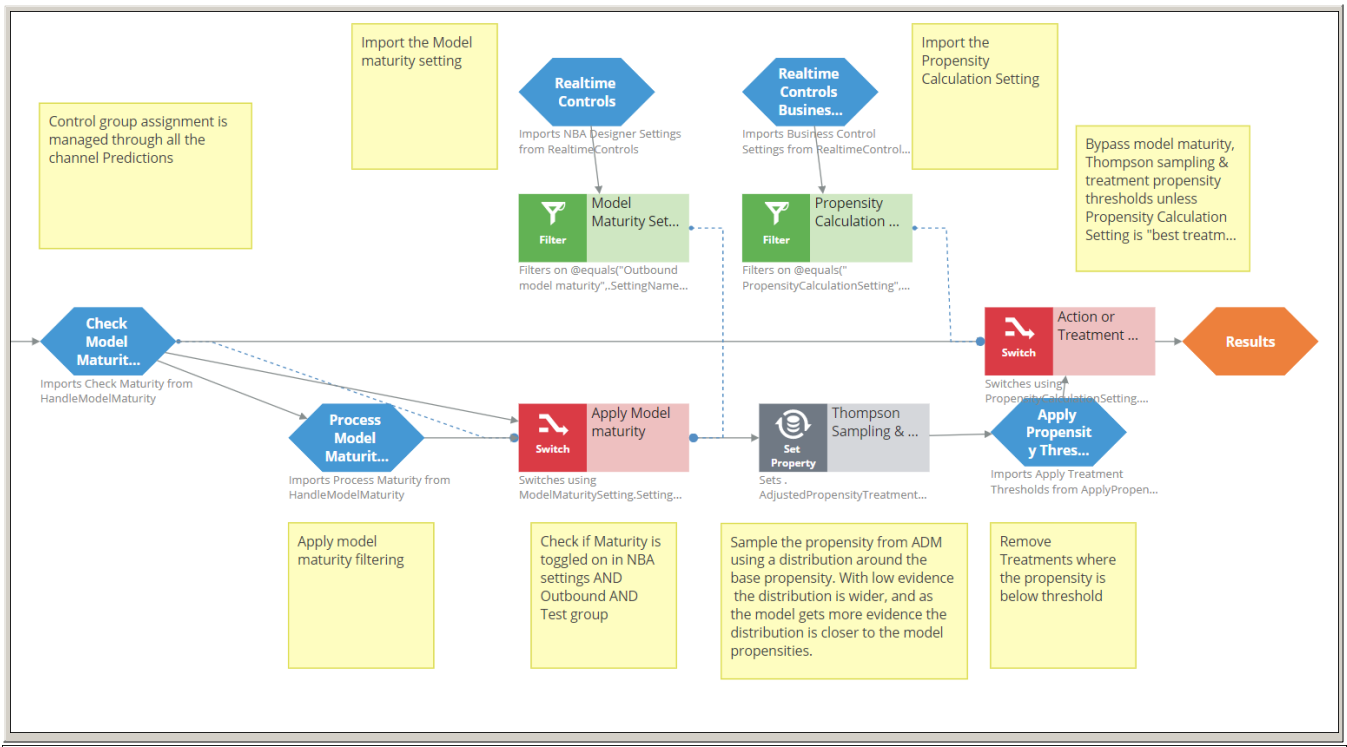


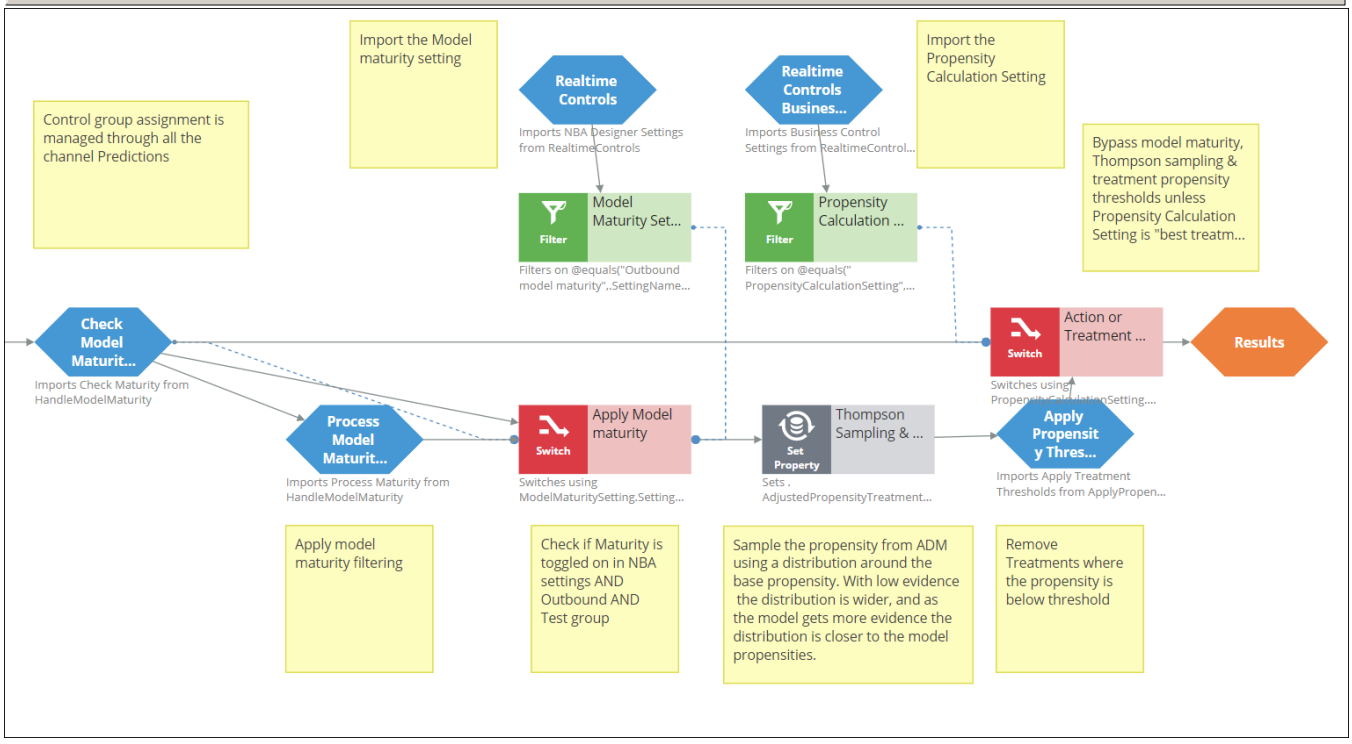


The Define Control Group and Web Treatment Model Impl sub-strategies are effectively the same as their Action level equivalents already described in Predict Action Propensity prediction strategy.

The Random / Model shape sets the following two properties:.pyModelPropensity = .pyPropensity .pyPropensity = @If(.ModelControlGroup=="Control",@random(0,1.0),.pyPropensity)

The second portion of this strategy is ostensibly the same as the right portion of its Action equivalent, i.e. the Predict Action Propensity Prediction Strategy, where the Outbound Model Maturity, Thompson Sampling and Apply Propensity Thresholds processes are applied.





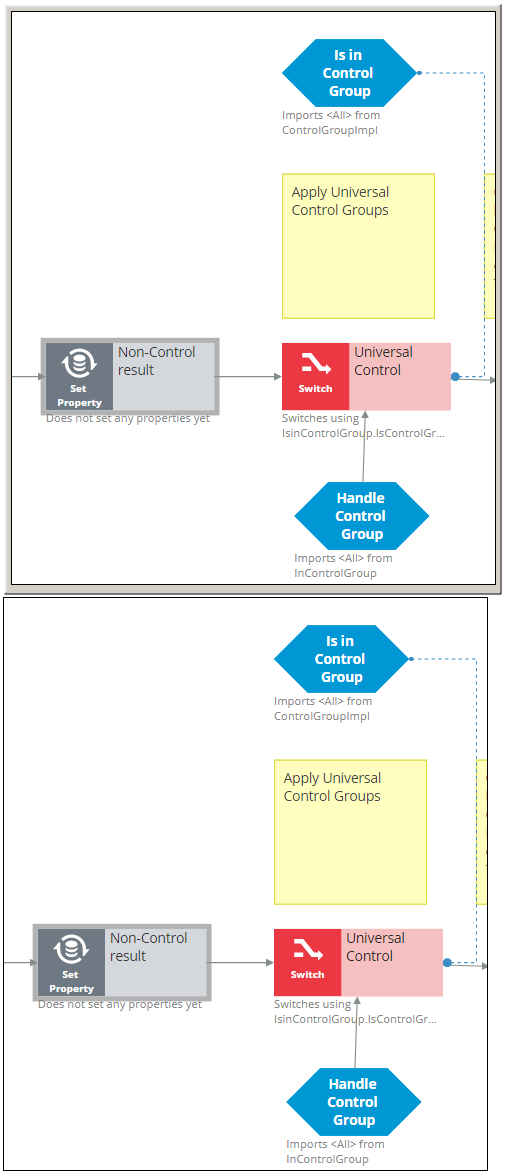
The main differences from the action processing are:

- Bypass model maturity, Thompson sampling and action propensity thresholds unless Propensity Calculation Setting is **best treatment**. These processes will already have been applied in the Predict Action Propensity strategy).
- Set *AdjustedPropensityTreatment* to the calculated propensity and set *pySimulatedPropensity* and *FinalPropensity*.
- Apply propensity thresholds and remove treatments where the propensity is below the treatment threshold.

Control Group Processing

This is the section in the Next-Best-Action strategy framework where a universal control group can be configured.

Pega Customer Decision Hub



The **Non-Control** result is the entry point to this section and at present does not set any properties.

The **Is in Control Group** (*ControlGroupImpl*) sub-strategy is where presence in a control group can be established – this could be using a Segment Filter or some other statistical method, but by default it simply returns false – that is, there is no control group.

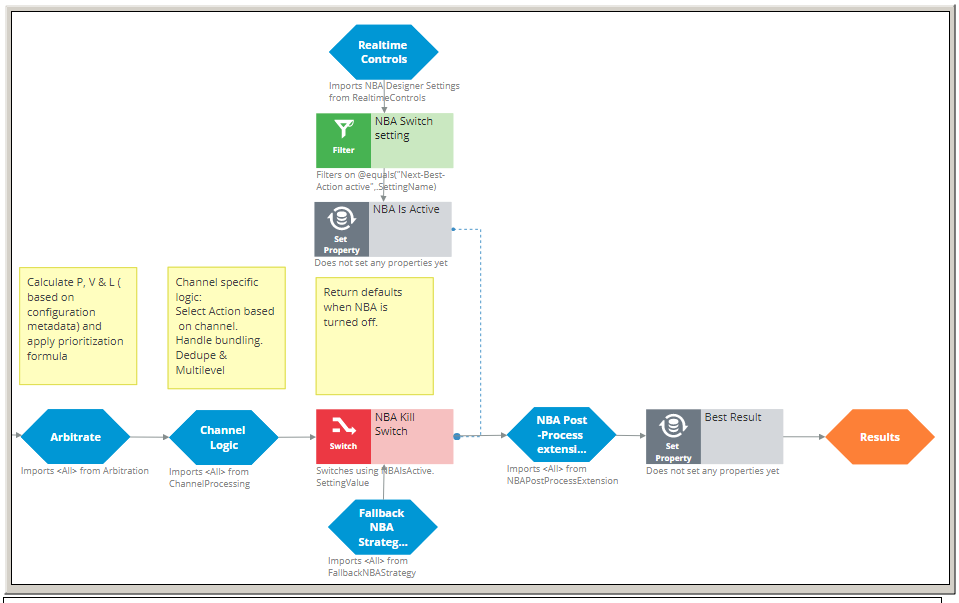
The **Handle Control Group** (*InControlGroup*) sub-strategy is where any control group processing should be defined, for example by defining properties that can be persisted to Interaction History or setting conditions such that no actions are delivered. By default, the strategy returns an empty page list.

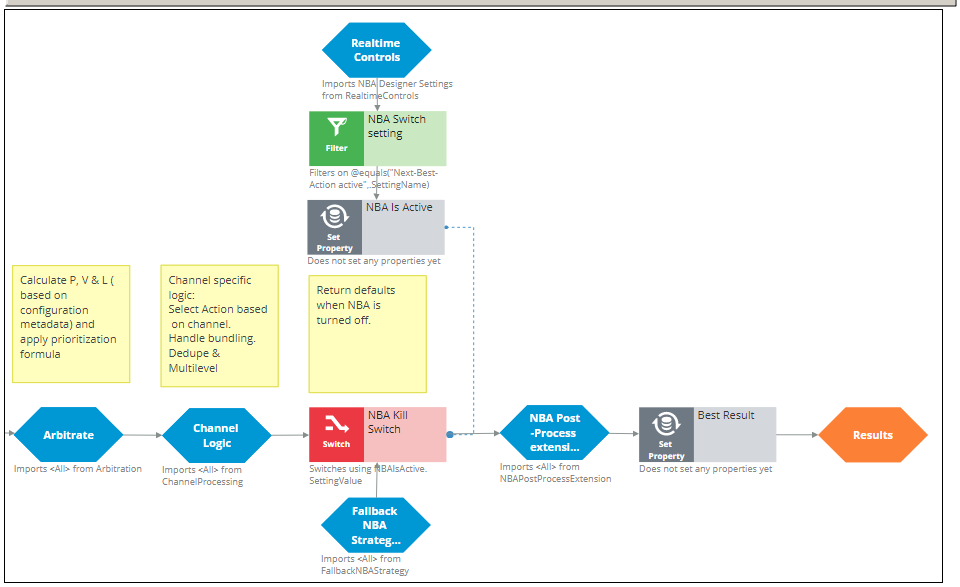
NBA Strategy framework components: Arbitrate through NBA Post-Process extension

This is the last portion of the Next-Best-Action strategy framework and includes arbitration and final channel processing, followed by an option to use an alternate strategy (the *NBA Kill Switch*) and the final extension point *NBA Post-Process Extension Point* sub-strategy.

Pega Customer Decision Hub

The *NBA Kill Switch* is actually the **Next-Best-Action active** setting in the *NBADesignerSettings* DDR; if this is set to false, the Fallback NBA Strategy is invoked instead of using the NBA framework. This fallback strategy is provided with no logic, but can be extended to include any default actions to be presented in case NBA is switched off.





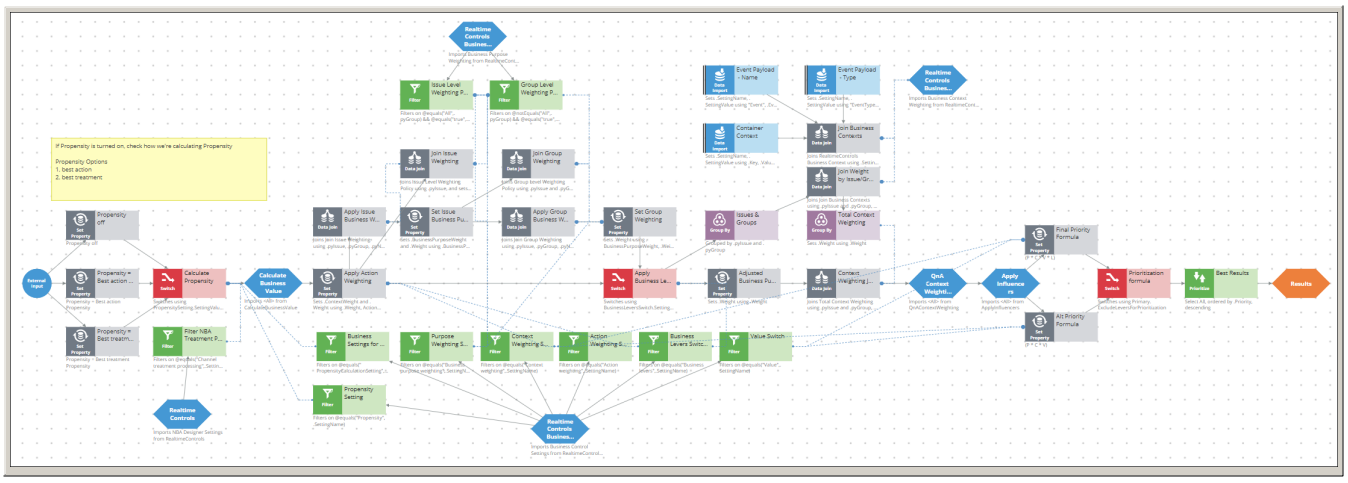
- Arbitration strategy
- Channel Processing strategy
- NBA Post-Process Extension Point strategy

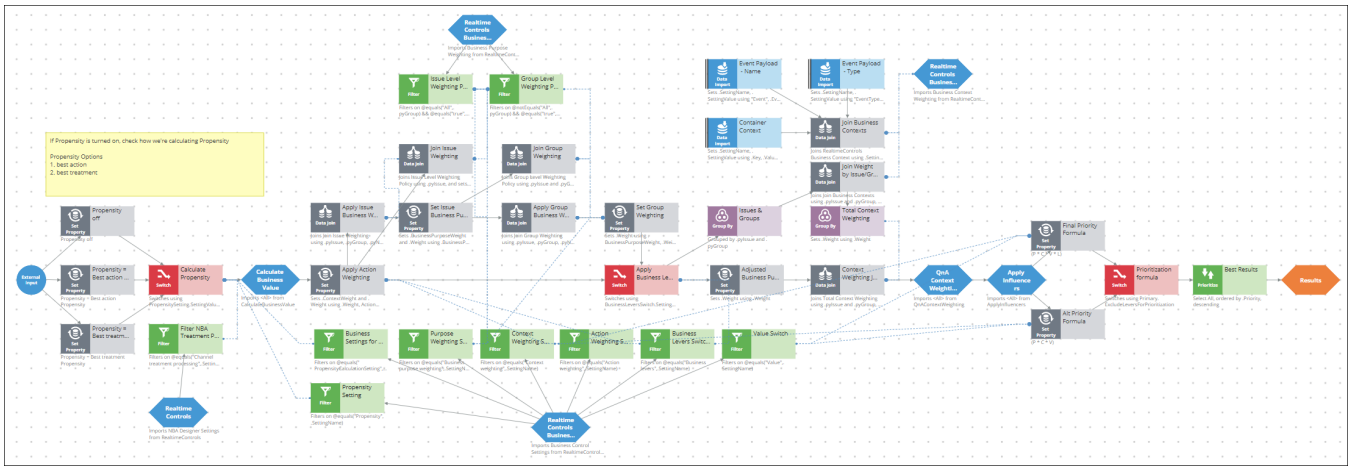
Arbitration strategy

The Arbitration strategy is responsible for setting up the attributes that will be used in the final priority calculation (P * C * V * L). These settings are managed within Next-Best-Action Designer and saved into decision data rules (DDRs). These DDRs are imported via the various Realtime Controls sub-strategies in the strategy, and filters are used to obtain individual setting values for some of these.

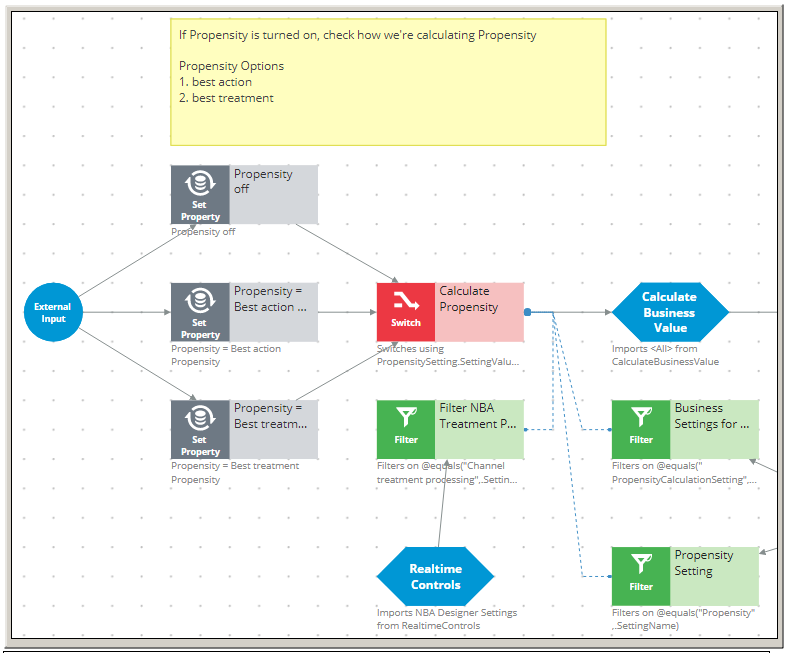
Pega Customer Decision Hub

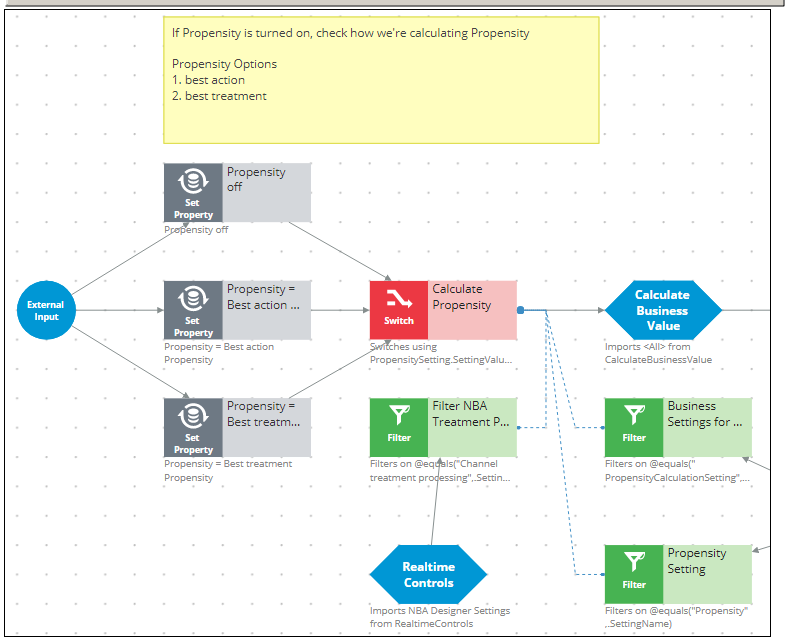
A high-level view of the strategy is shown first, and then each section will be described in more detail.



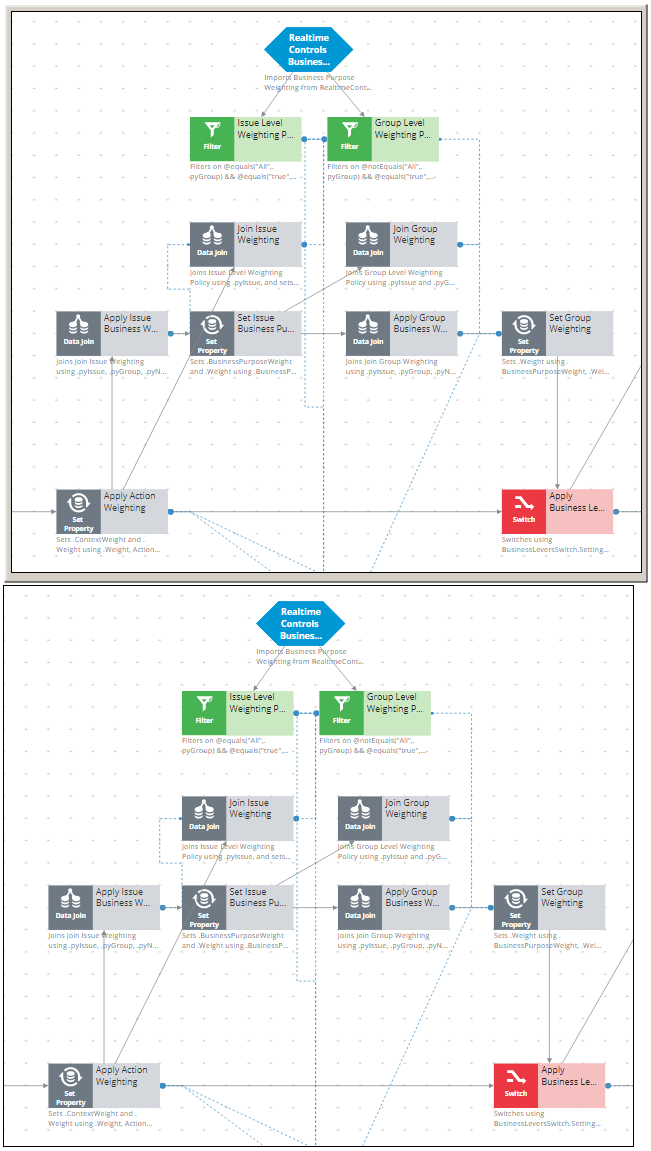


The strategy first sets the *FinalPropensity* property to be based on the action or treatment propensity, or a fixed value of 1 if propensities are not being used. Then the strategy sets the business value. By default, the action-level business value is used. Modify the Calculate Value Extension point strategy to change this.

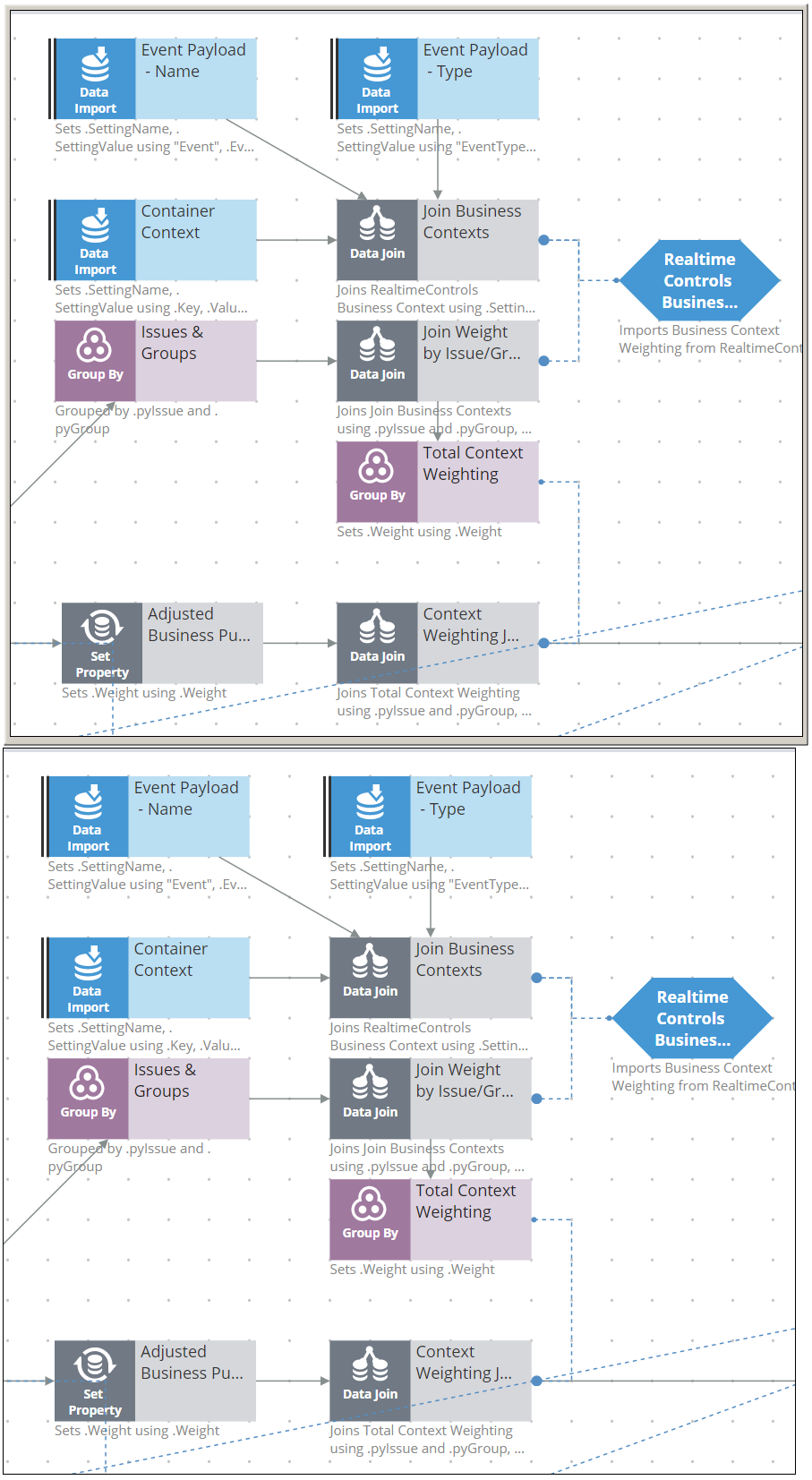




After that, if the Business levers are enabled, the strategy determines business purpose weighting, first at the issue level and then the group level. These weightings are both added to the action weight, divided by 100 and then added to 1, so a total weighting of 55 sets a final weighting of 1.55.



The strategy then applies context weighting by looking at the Container and Event payloads and searching for matching Key and Value pairs for each issue and group. If a match is found, the weighting is added to the weight for the specified issue and group. This weighting is also divided by 100 and then added to 1, so a total weighting of 55 sets a final weighting of 1.55.



The next step is to calculate Q & A (question and answer) related context – if any. By default, there is none, but it can be added via the *QnAContextWeightingExt* extension point.

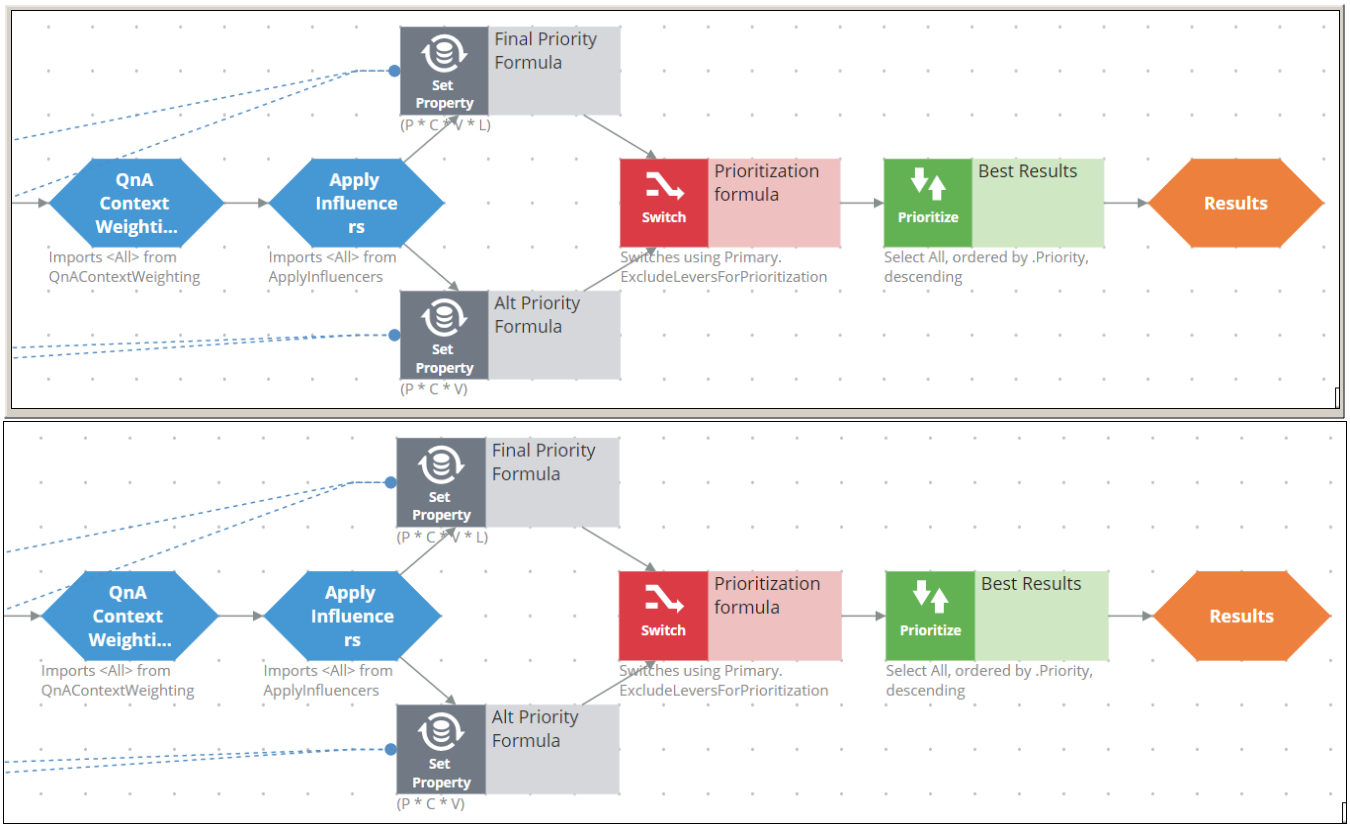
Then the ApplyInfluencers strategy is invoked to apply any upweighting required by the action influencer contact policy configuration.

After the properties are calculated and adjusted based on the settings configured in Next-Best-Action Designer, the Prioritization formula Switch determines which formula to use to calculate the Priority based on the ExcludeLeversForPrioritization setting. Note that this feature is only used during simulation and is not user configurable.

If false, the standard formula is used:`Priority = .FinalPropensity * .ContextWeight * .Value * .Weight`

Otherwise Weight is omitted and the alternate formula is used:`Priority = .FinalPropensity * .ContextWeight * .Value`

Then all actions and treatments are prioritized based on Priority in descending order.

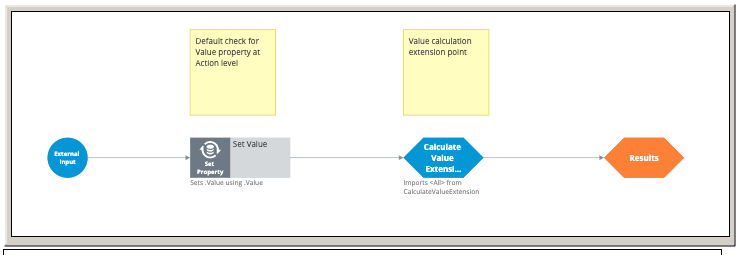


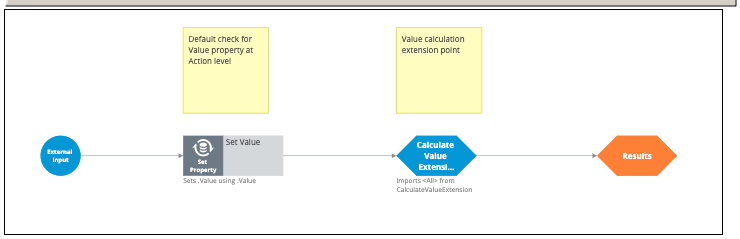
- Calculate Business Value strategy
- Question & Answer (QnA) Context Weighting strategy
- Apply Influencers strategy

Calculate Business Value strategy

The *CalculateBusinessValue* strategy uses the business value property of the action. An extension strategy *CalculateValueExtension* is provided for more sophisticated value calculations.

Pega Customer Decision Hub

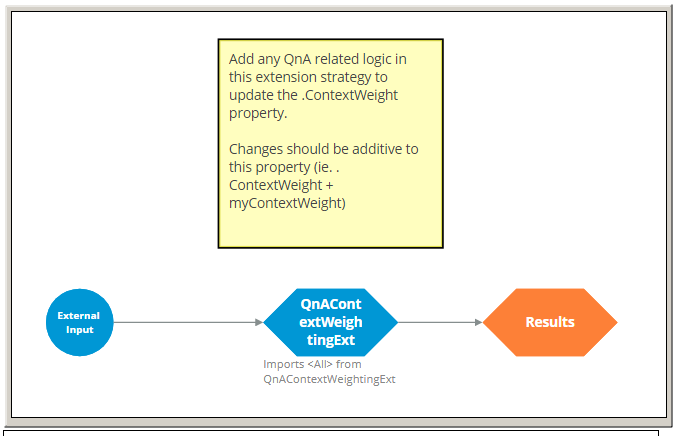


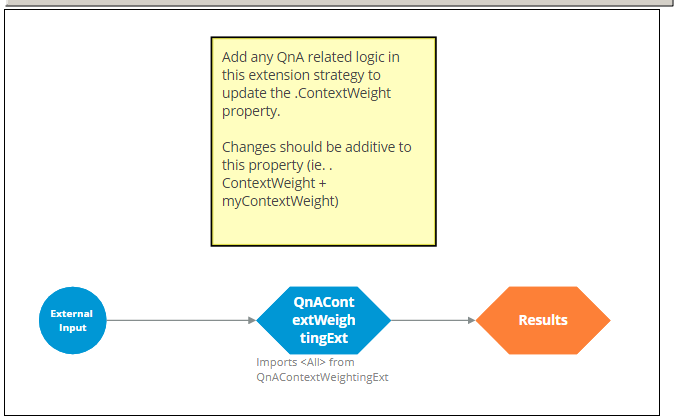


Question & Answer (QnA) Context Weighting strategy

The *QnAContextWeighting* strategy simply contains a call to the *QnAContextWeightingExt* extension point sub-strategy. This can be used to add any Q & A related context weighting. Any weighting calculated here should be added to the *ContextWeight* property.

Pega Customer Decision Hub

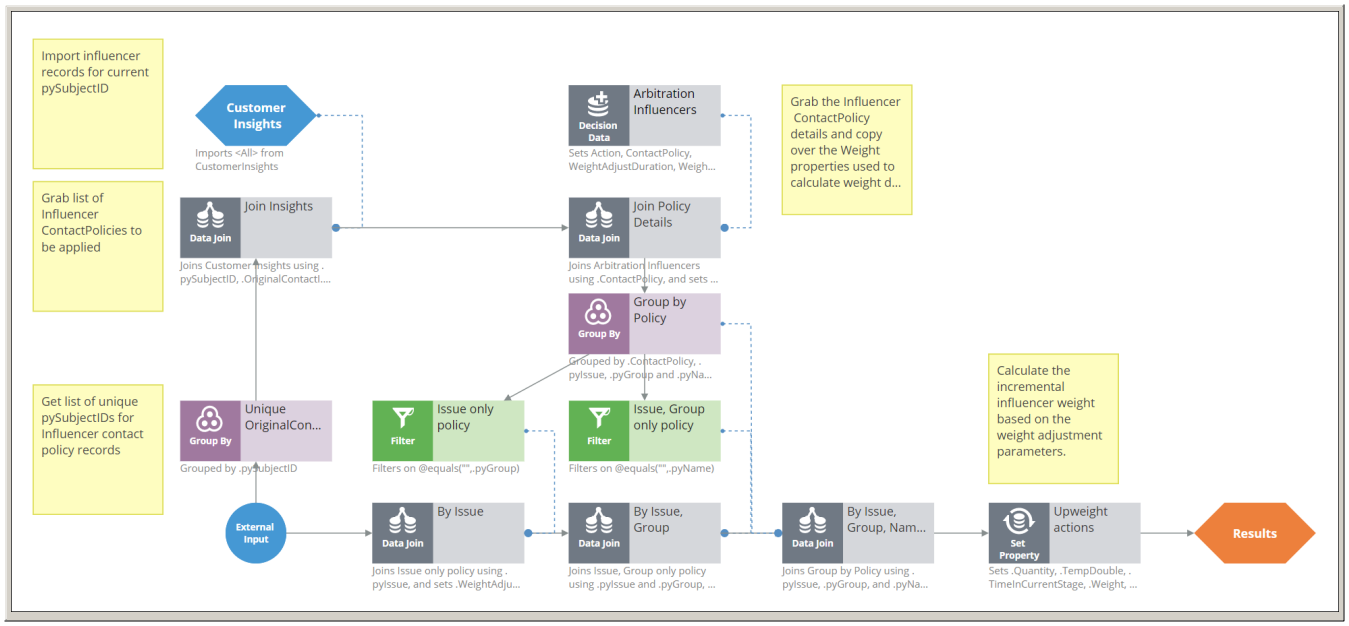


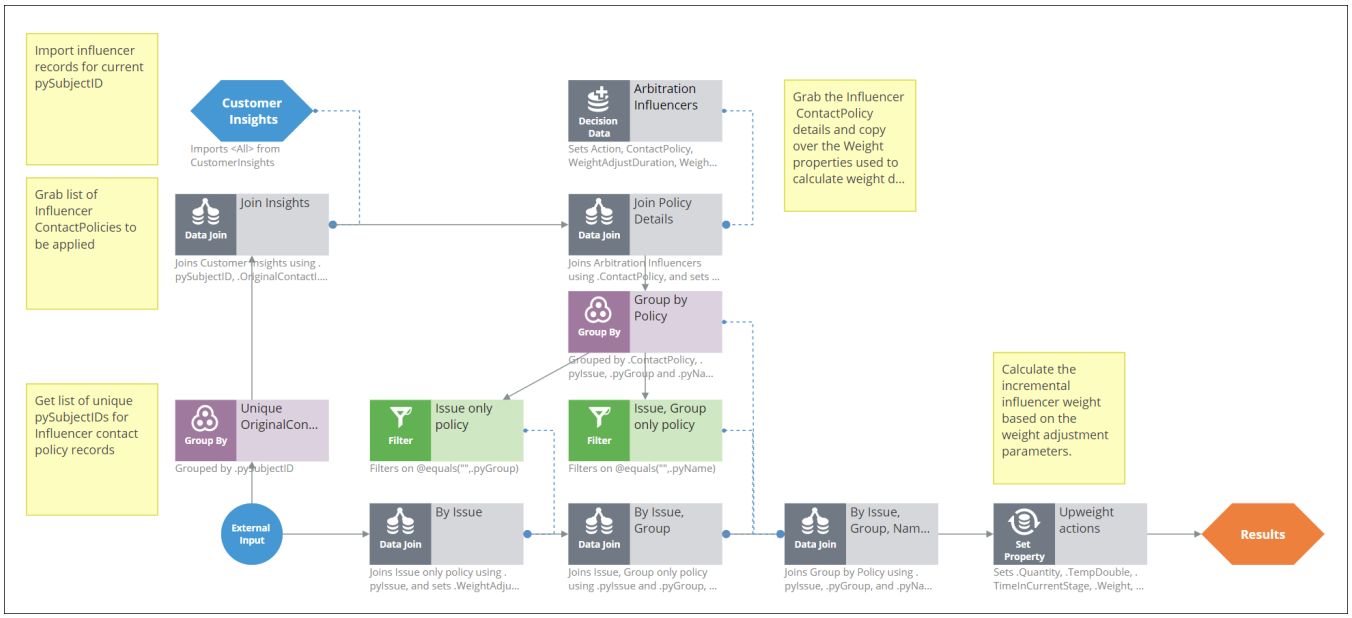


Apply Influencers strategy

This strategy uses action insights of the type *Influencer* for the customer. If there is a matching influencer action the weighting will be increased or decreased based on weighting adjustment definitions in the *Arbitration Influencers* DDR.

Pega Customer Decision Hub





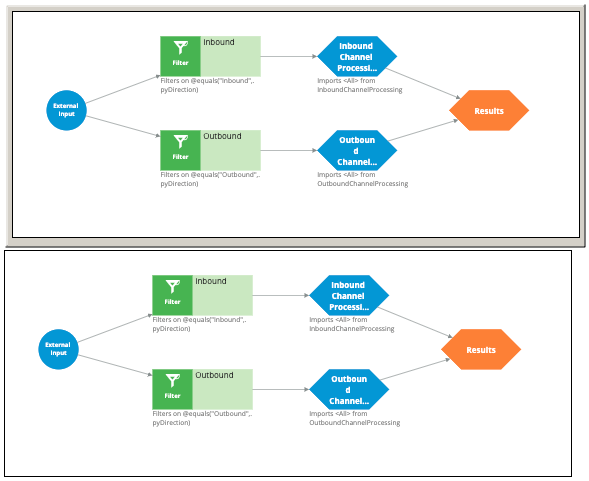
The logic branches into two streams, one is the set of input actions and treatments, while the other one obtains the action insights and arbitration influencers data.

For a detailed description of the logic, refer to the annotations on the strategy.

Channel Processing strategy

After priority has been assigned to the actions and treatments, and they have been ranked by priority, they need to be filtered based on the specific channel requirements. The channel processing strategy handles this for inbound and outbound channels.

Pega Customer Decision Hub



- Inbound Channel Processing
- Outbound channel processing

Inbound Channel Processing

The *Inbound Channel Processing* strategy handles all inbound channel processing; a separate sub-strategy is called for each of Web, Mobile, Agent Assisted (Call Center or Retail), and Other (everything else).

Pega Customer Decision Hub



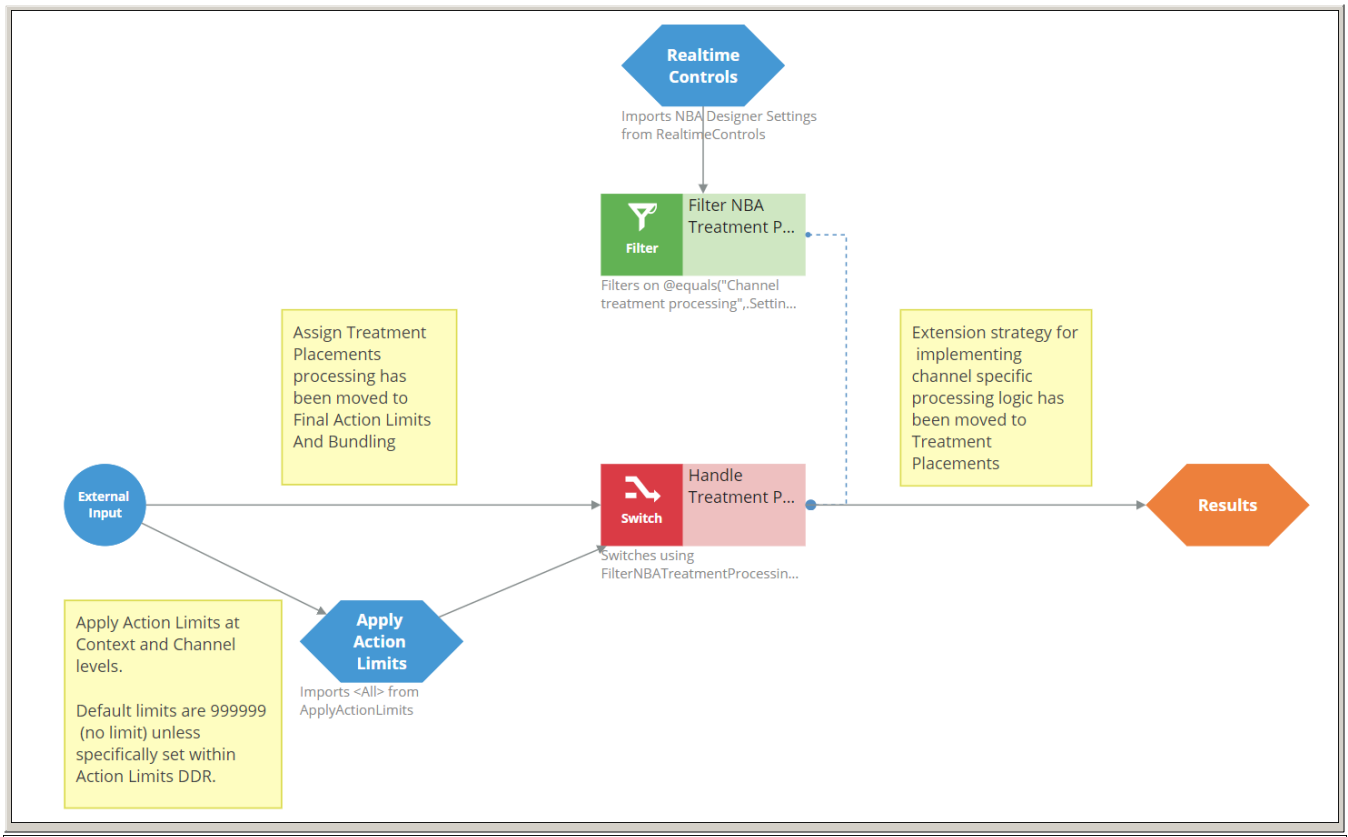
The Process Web and Mobile Channel strategies

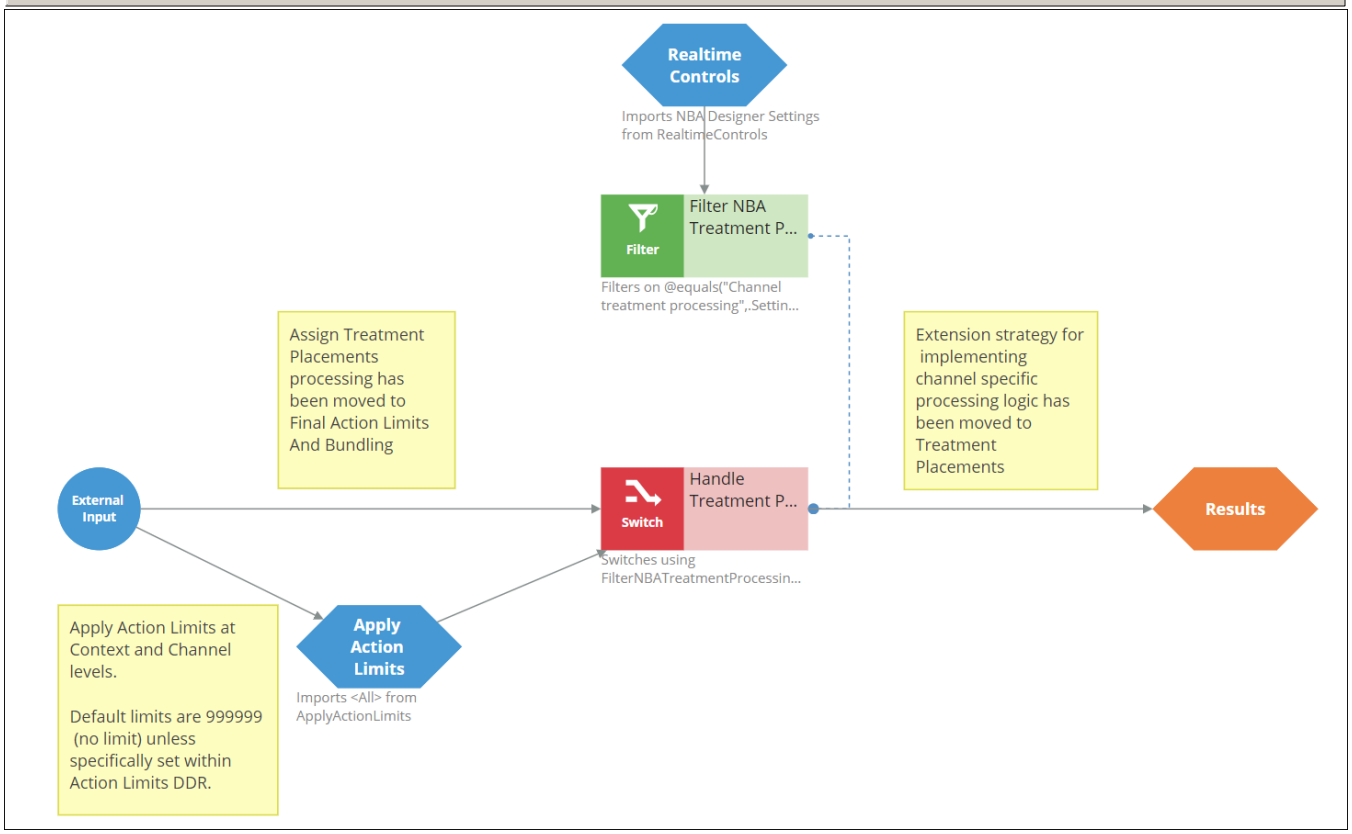
The Process Web Channel and Process Mobile Channel strategies are effectively identical. Each determines if the Channel treatment processing switch is on, and if so passes actions and treatment without alteration, otherwise the Apply Action Limits strategy is called to perform any channel specific filtering based on preset limits.

Note:

From version 8.5 onwards, Assign Treatment Placements processing has been moved to Final Action Limits And Bundling, and the extension strategies for implementing channel specific processing logic have been moved to Treatment Placements (within Final Action Limits And Bundling).

The Process Web Channel strategy is shown below.



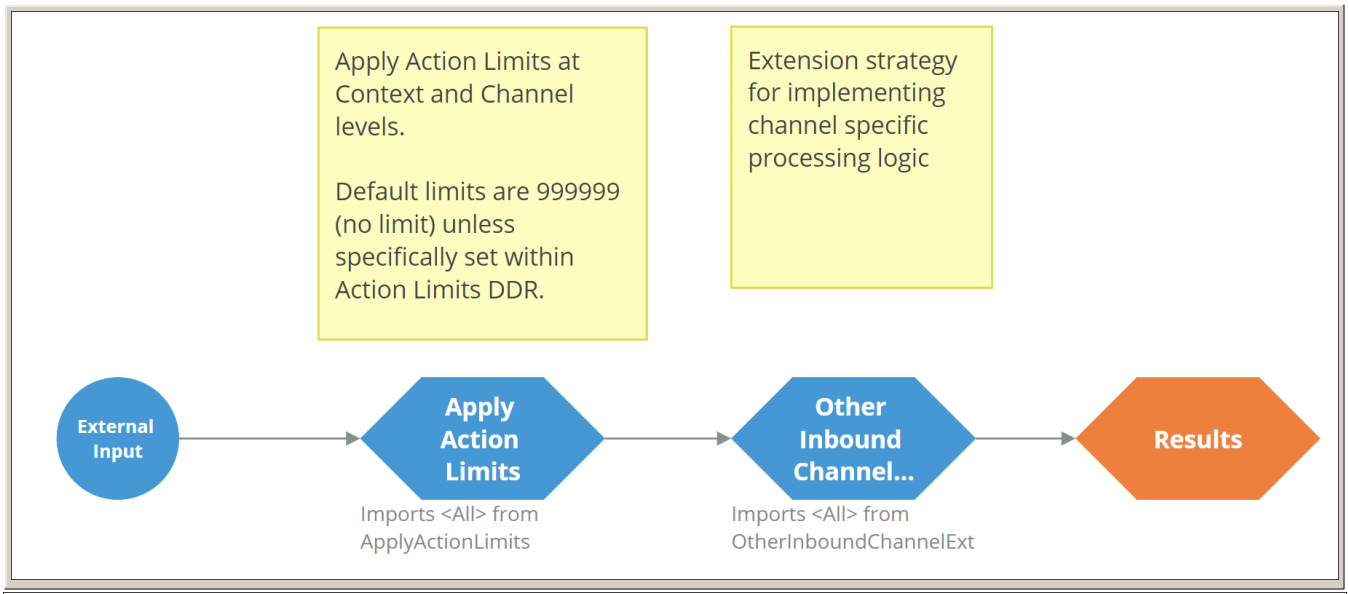


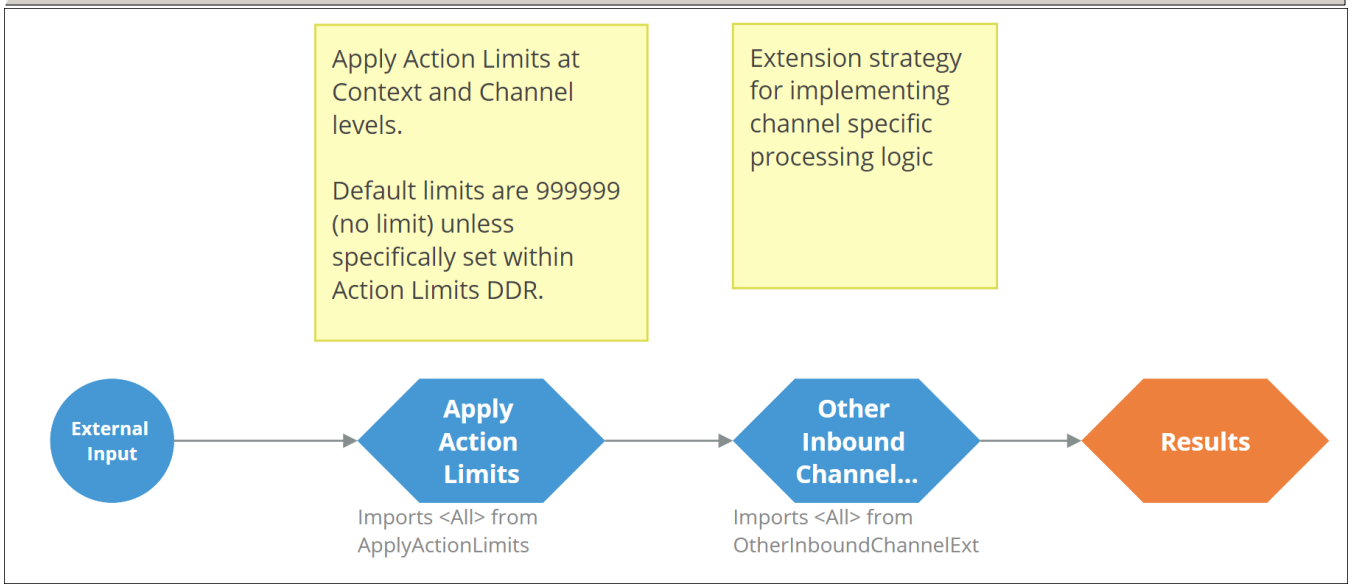
Assisted and Other Inbound Channel strategies

The *Assisted Inbound Channel* and *Other Inbound Channel* strategies follow the same pattern. Each calls the Apply Action Limits strategy to perform any channel specific filtering based on preset limits.

An extension point strategy called Assisted / Other Inbound Channel Extension is provided after action limits processing to allow for any additional processing for the channel.

The following example is for the Other channel.





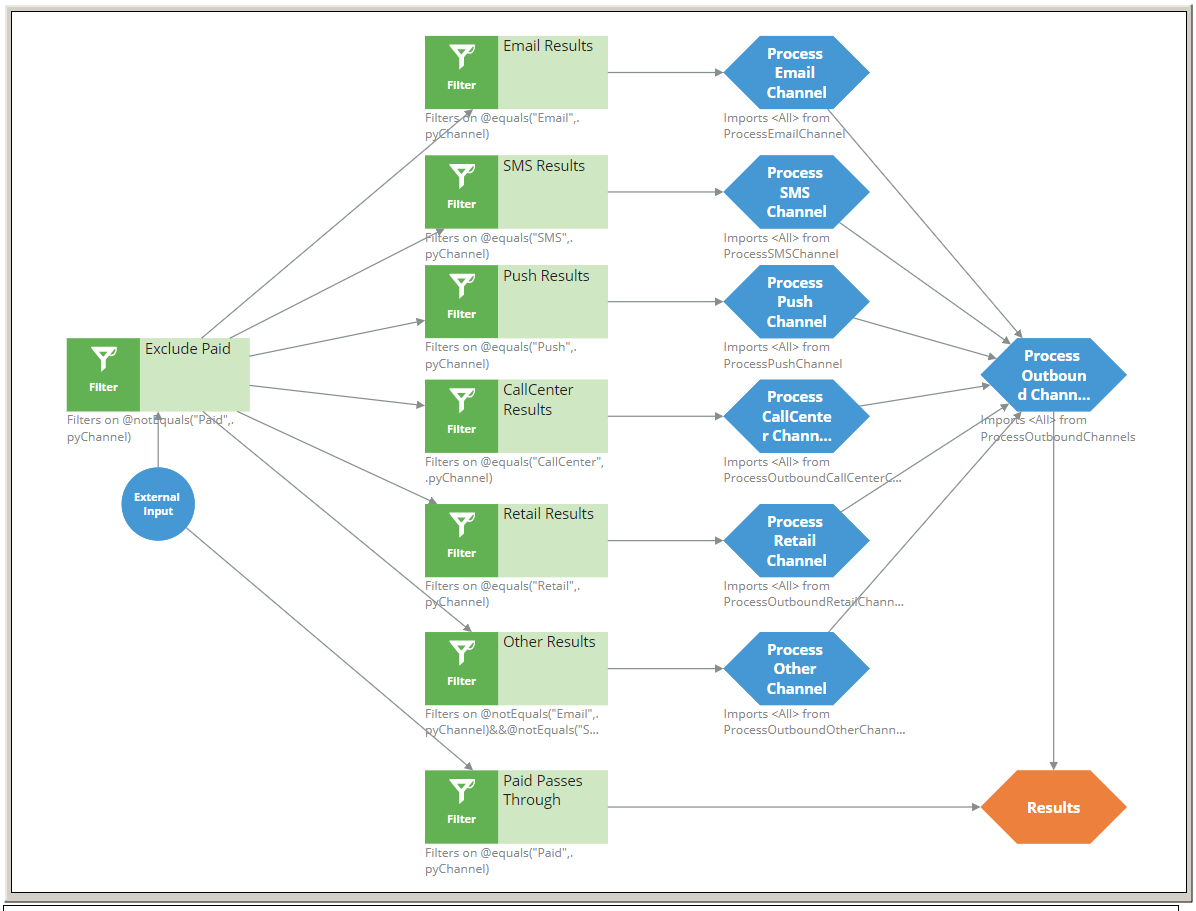
Outbound channel processing

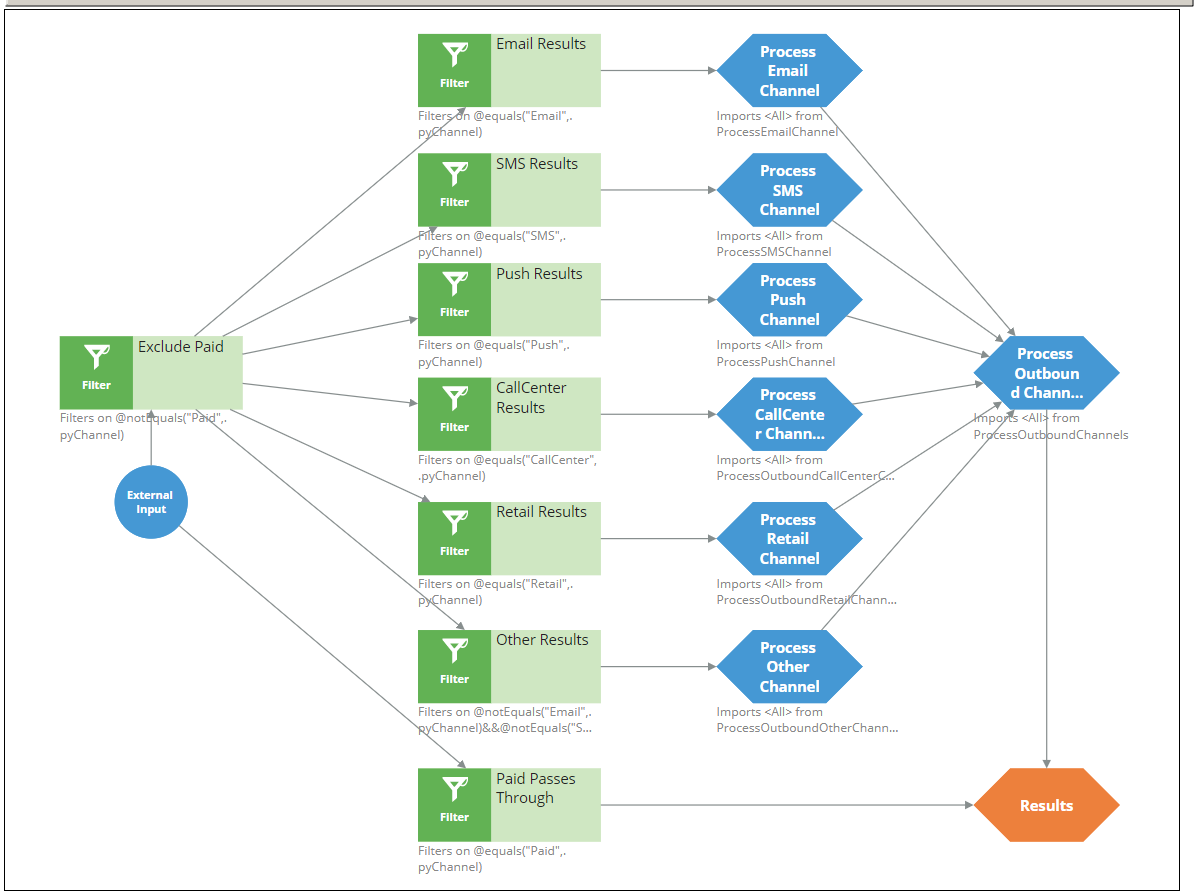
The *Outbound Channel Processing* strategy handles all outbound channel processing.

Pega Customer Decision Hub

The Paid channel is unique in that we want to send all actions to be processed by Paid Media Manager, so those actions pass through. For the other outbound channels, a separate sub-strategy is called for each of Email, SMS, Push, Call Center, Retail and Other (anything not in this list but not Paid). Each of these sub-strategies is intended as an extension point for including any additional channel specific processing.

They all contain no logic and are simply extension points that are then merged into the *ProcessOutboundChannels* strategy, which includes nothing other than a call to the *Top Offer Or Bundle For Outbound* sub-strategy.

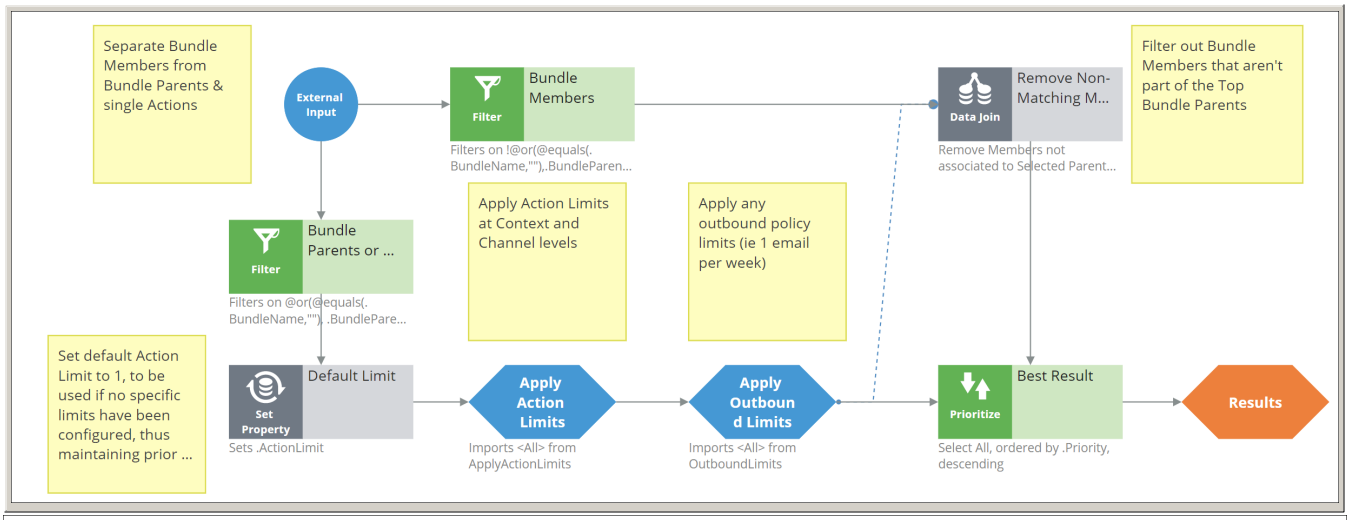


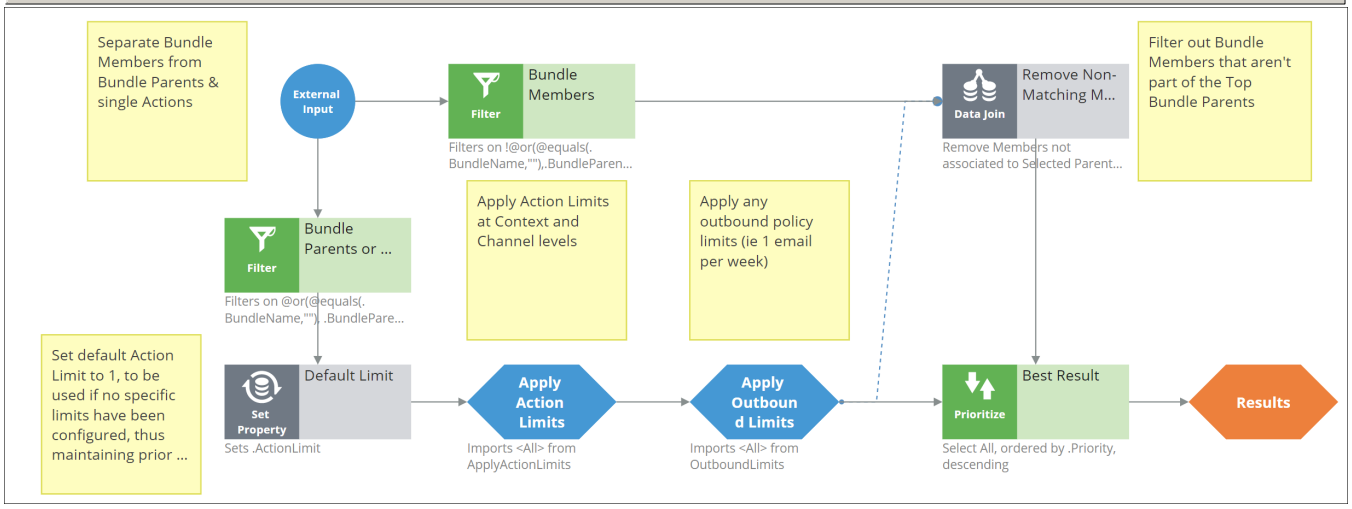


Top Offer Or Bundle For Outbound strategy

The *Top Offer Or Bundle For Outbound* strategy separates out bundled and non-bundled actions, applies action limits and contact policies to filter out actions and treatments as needed, and then prioritizes the bundle parents or treatments based on priority.

If a bundle is selected, the bundle members are also selected and pass through.

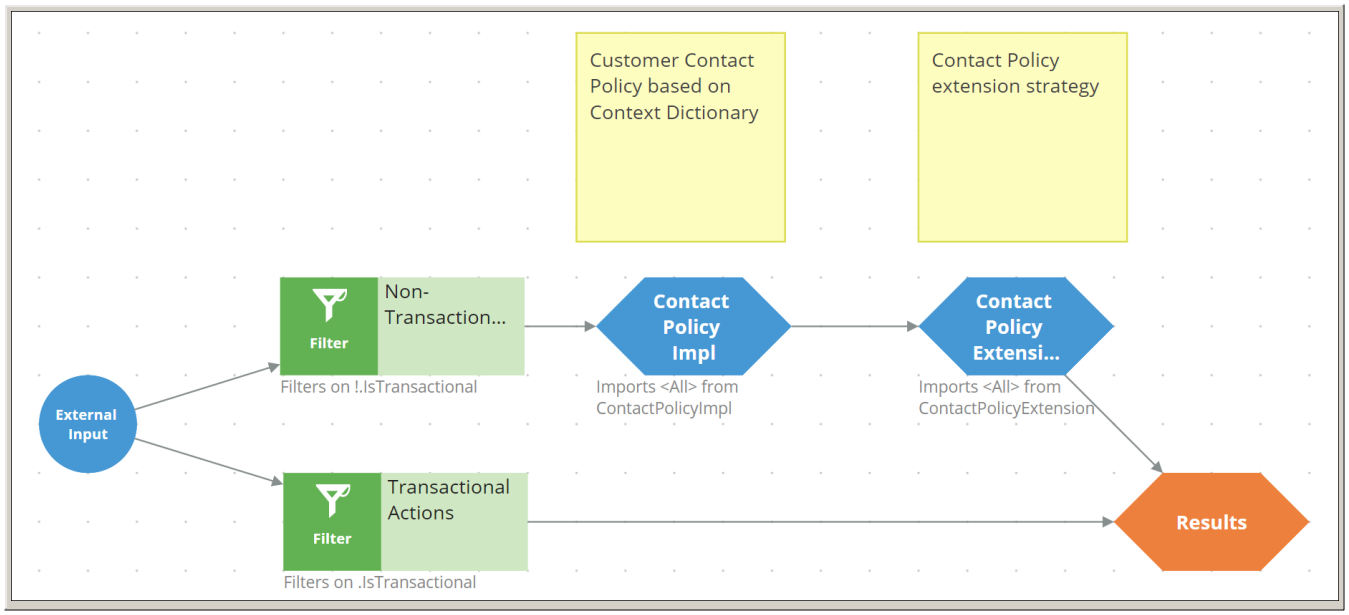


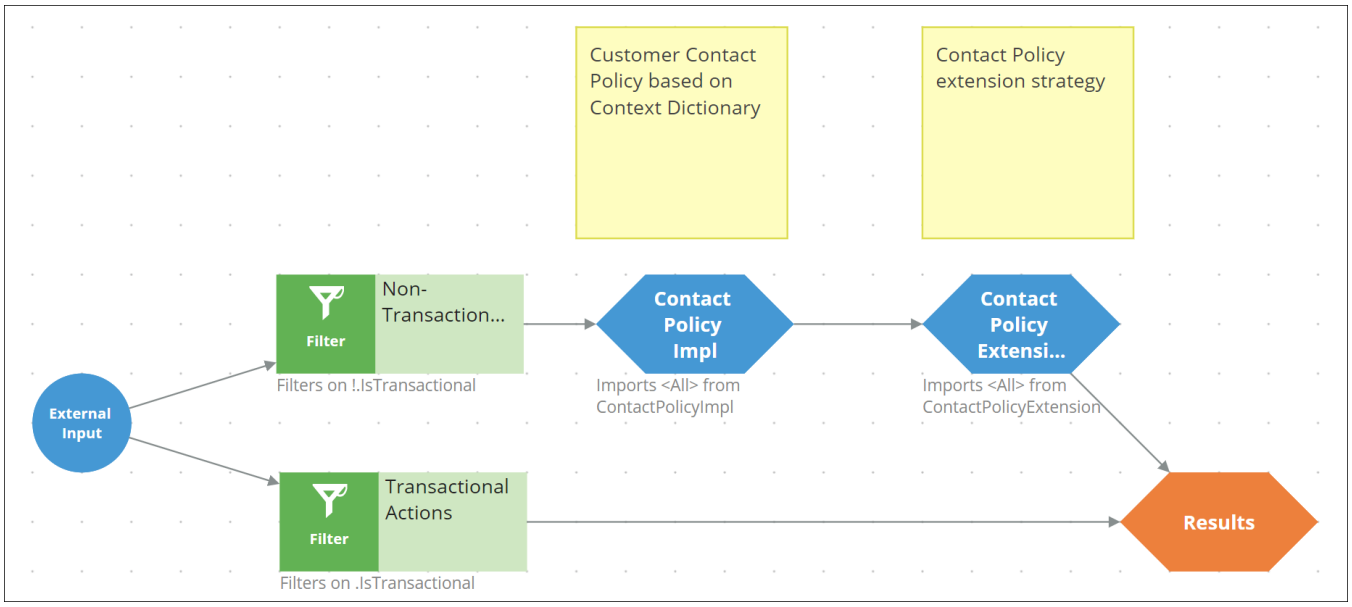


For a detailed description of the logic, refer to the annotations on the strategy.

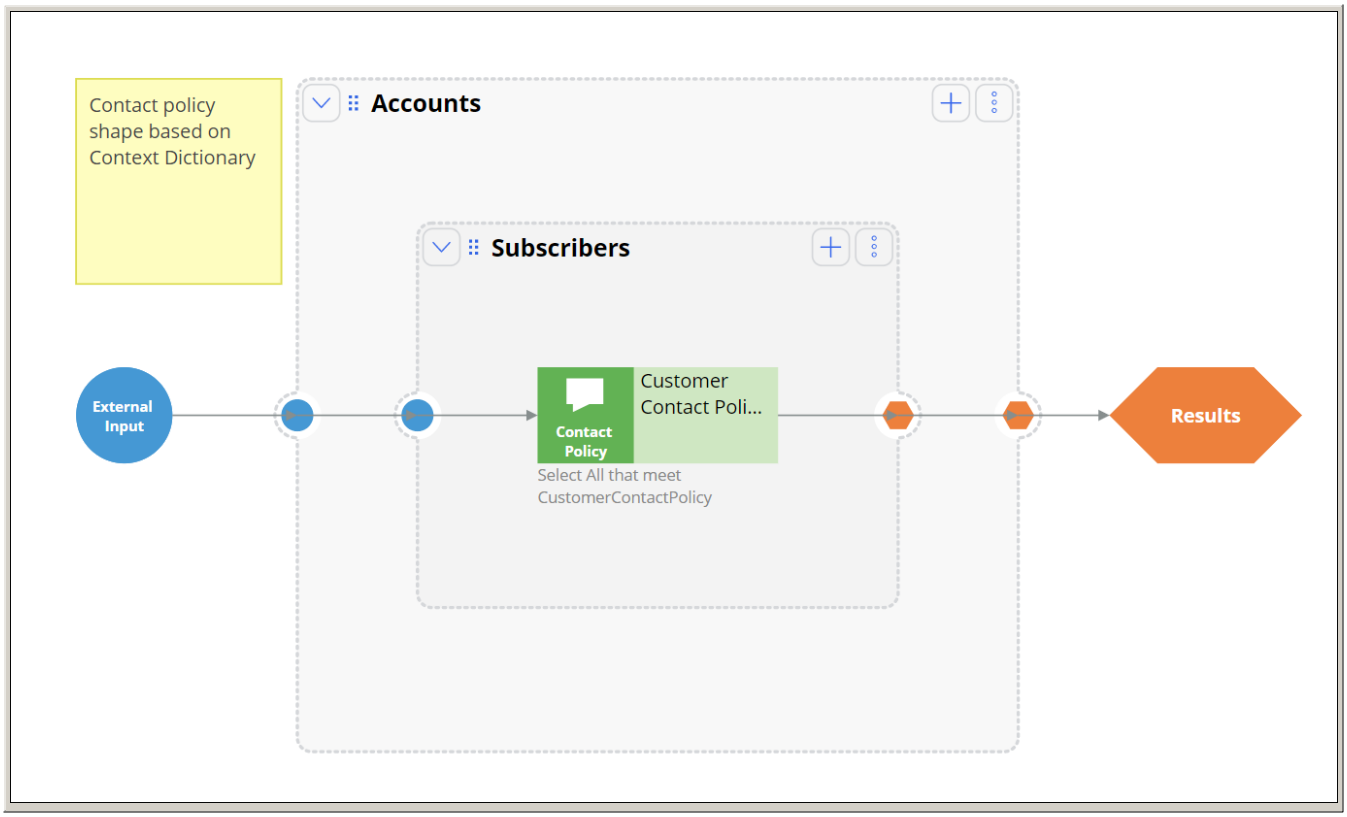
Outbound Limits strategy

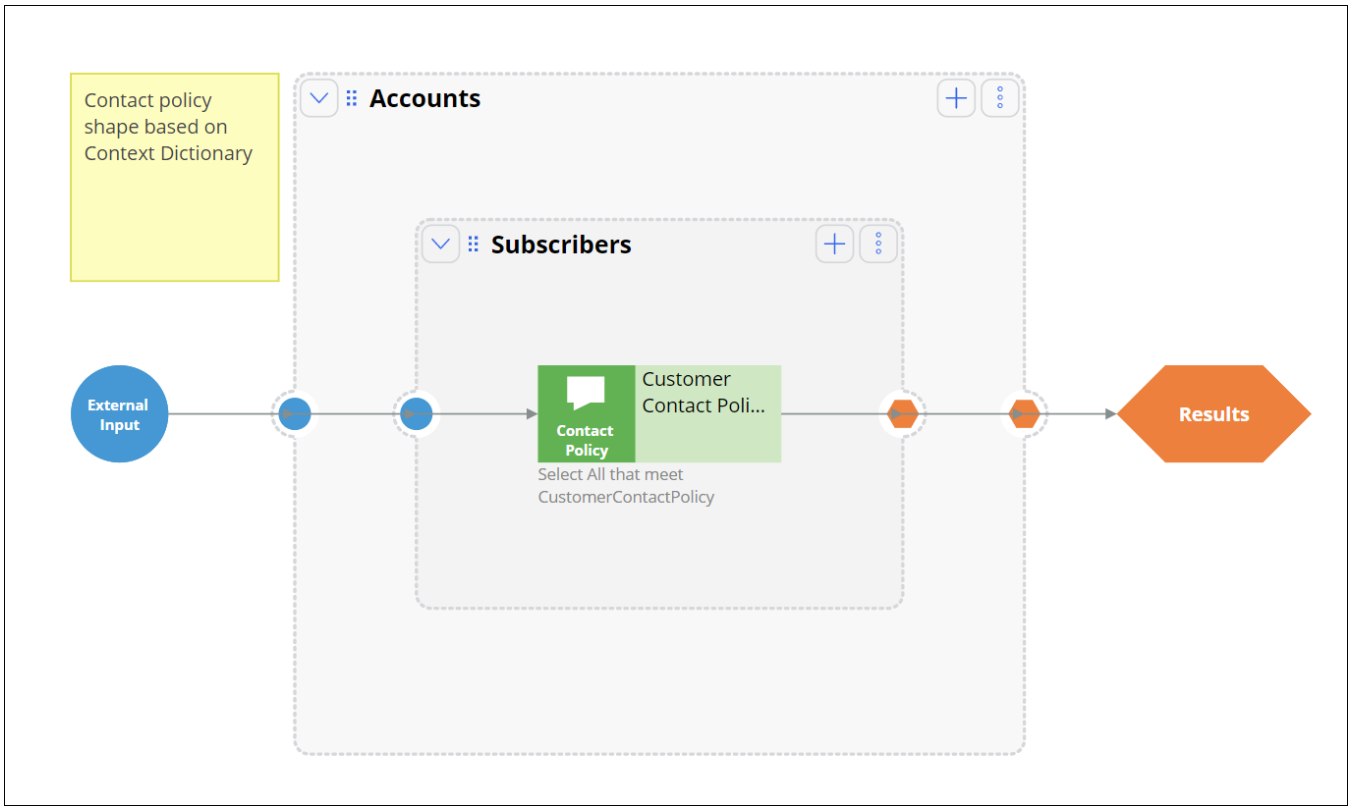
The Outbound Limits strategy implements any contact policies defined within the **Constraints** tab of Next-Best-Action Designer. The strategy enforces these outbound limits for non-transactional actions, whereas transactional actions, such as compliance-related communications, bypass these outbound limits to ensure they are delivered to the customer.





The *Contact Policy Impl* strategy ensures that the only contact policy for the current contact (*OriginalContactID*) is applied. The strategy is generated by Next-Best-Action Designer and is not modifiable.





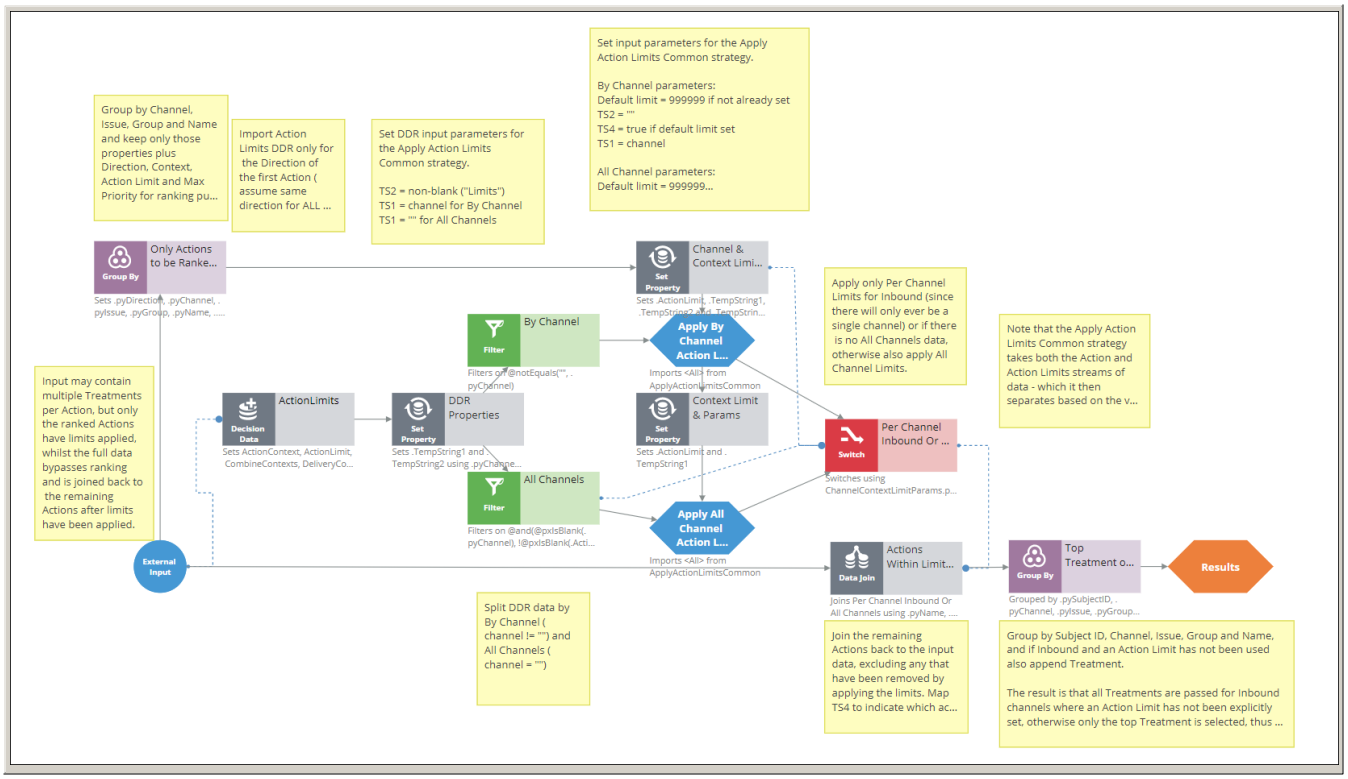
An extension point strategy *ContactPolicyExtension* is provided for any implementation-specific outbound limits.

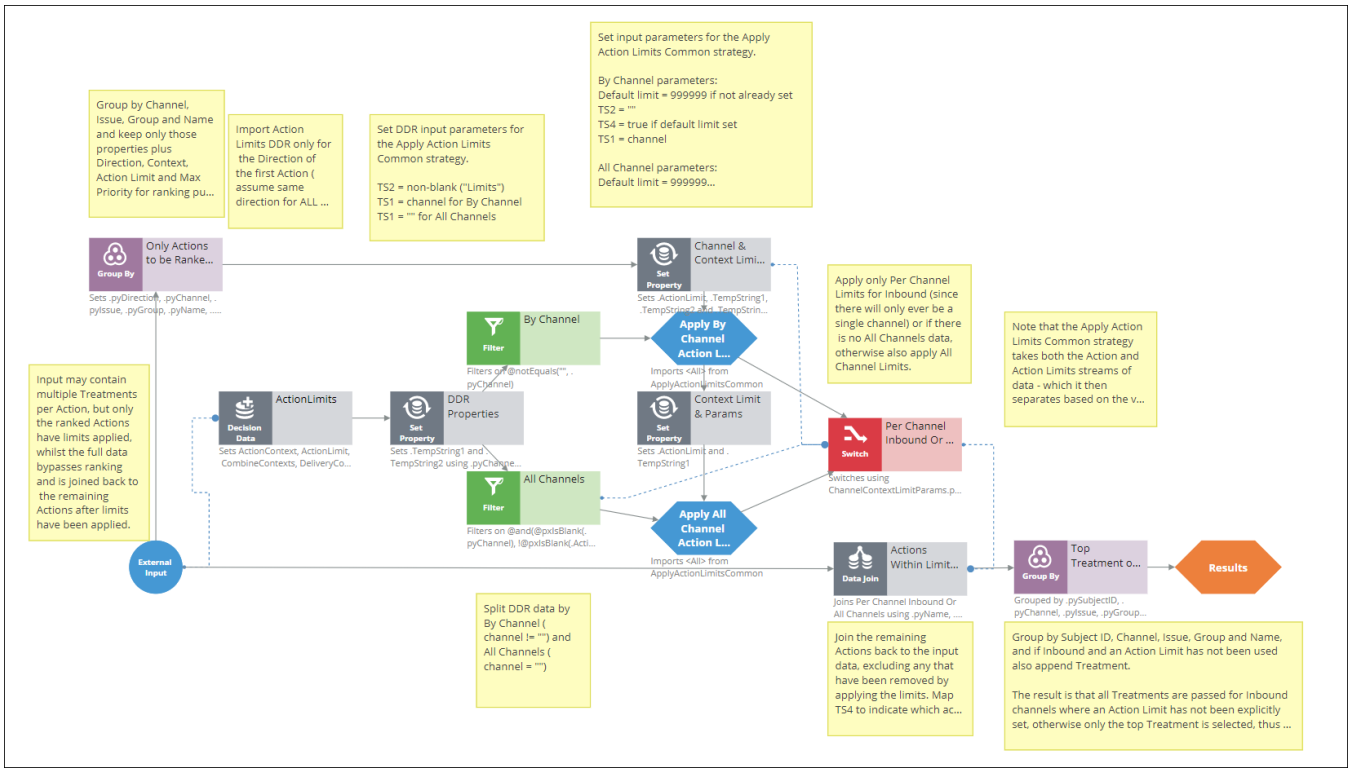
Apply Action Limits strategy

This strategy is called by all of the channel processing strategies (other than Paid) and applies limits on the number of actions emitted for each action context and for each contact, as defined in the Action Limits DDR.

A default limit may be set by setting the property ActionLimit prior to calling the strategy, if this is not set, then a limit of 999999 is assumed, that is, no effective limit.

The functionality is described in more detail in *Additional NBA Strategy framework components*.

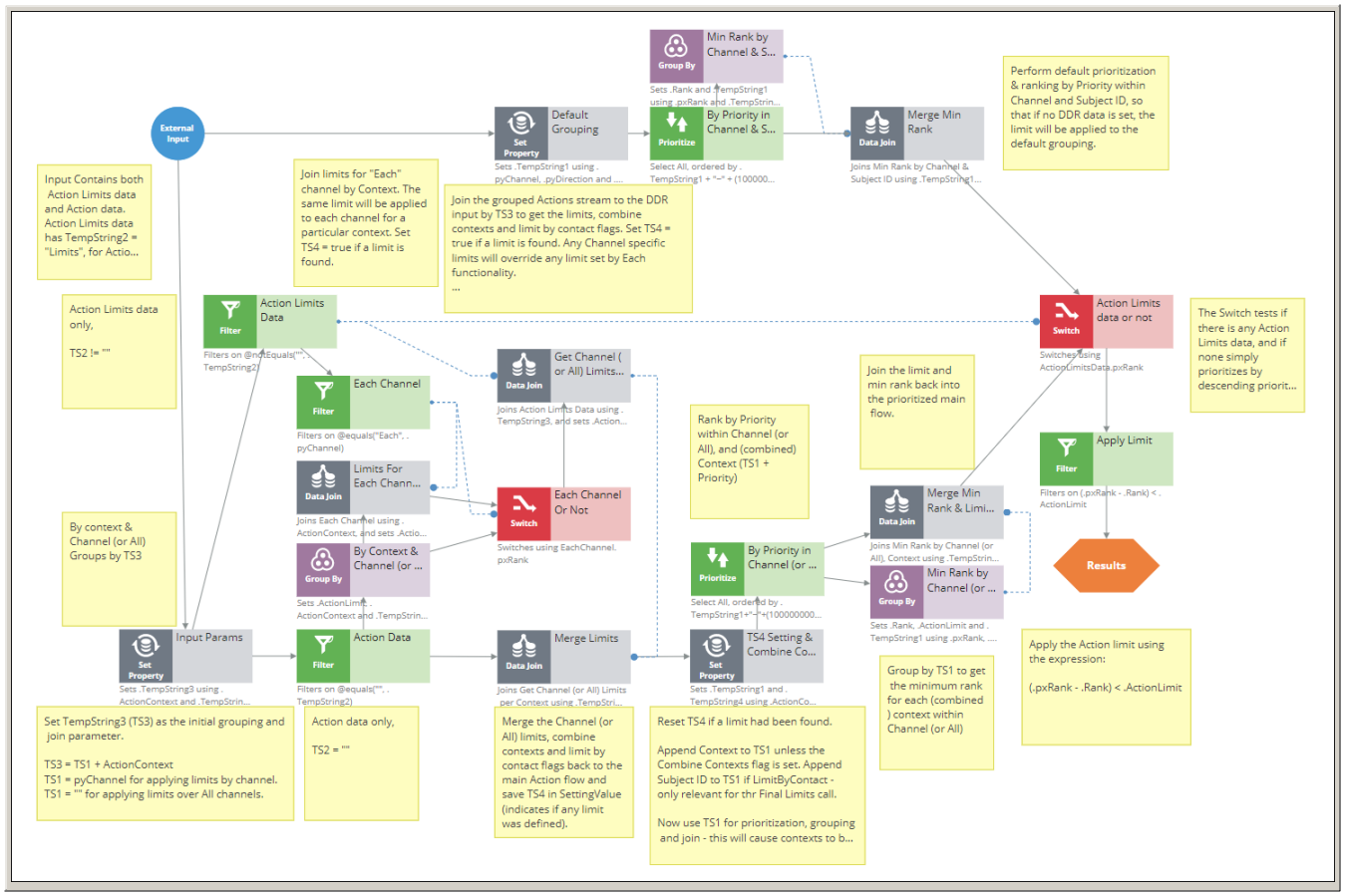


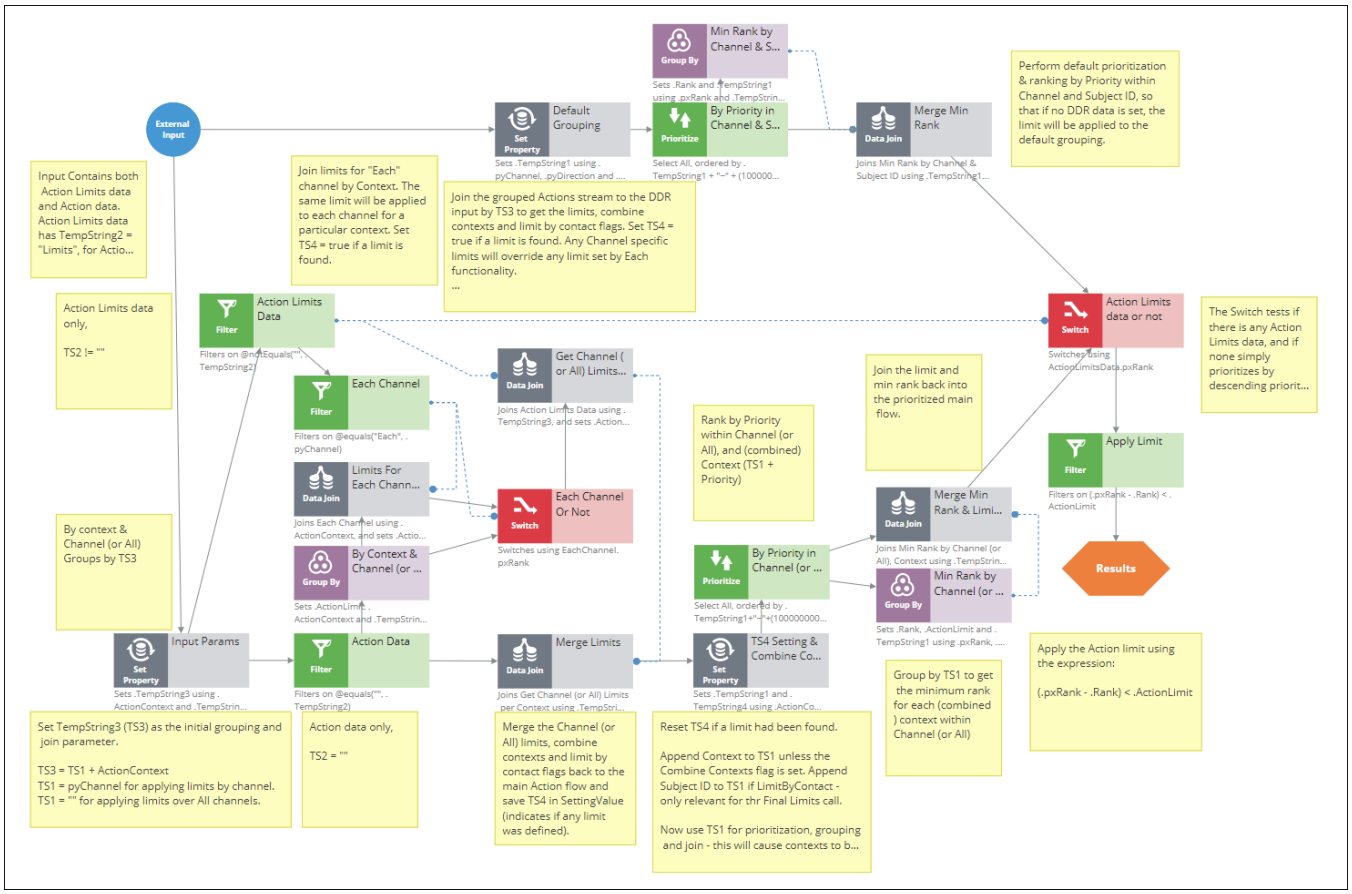


For a detailed description of the logic, refer to the annotations on the strategy.

Apply Action Limits Common strategy

This common sub-strategy is used by both the Apply Action Limits and Apply Final Action Limits strategies and is responsible for filtering actions that exceed specified limits set by direction, channel, context and contact.





For a detailed description of the logic, refer to the annotations on the strategy.

NBA Post-Process Extension Point strategy

This is the last functional component within the strategy framework and may be used for any post-processing which must be performed.

Pega Customer Decision Hub

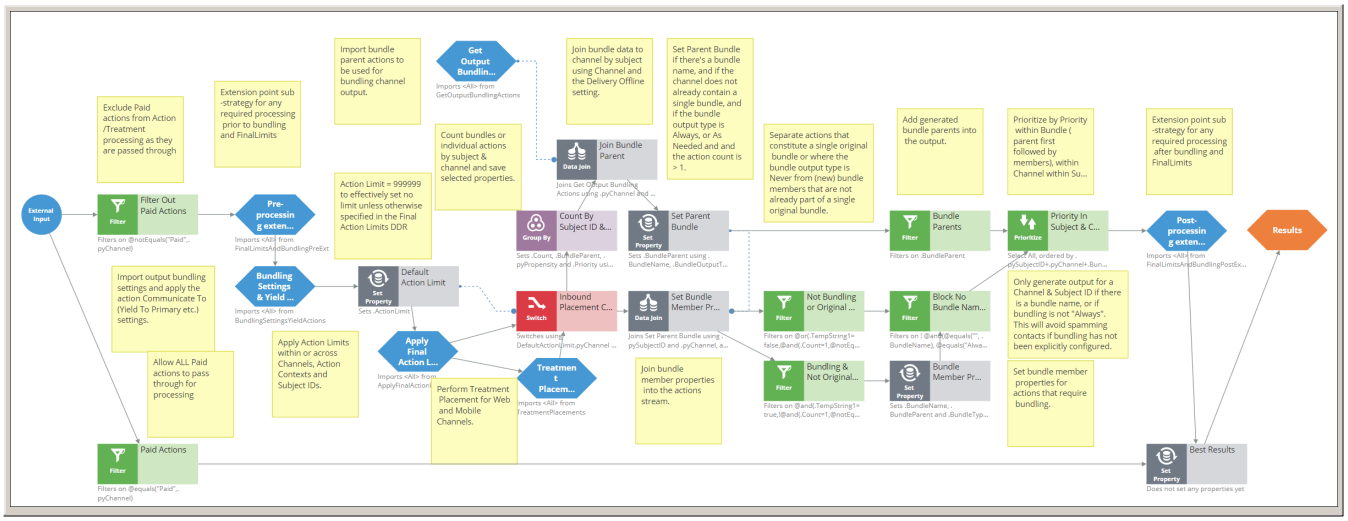
NBA Strategy framework components: Final Action Limits and Bundling strategy

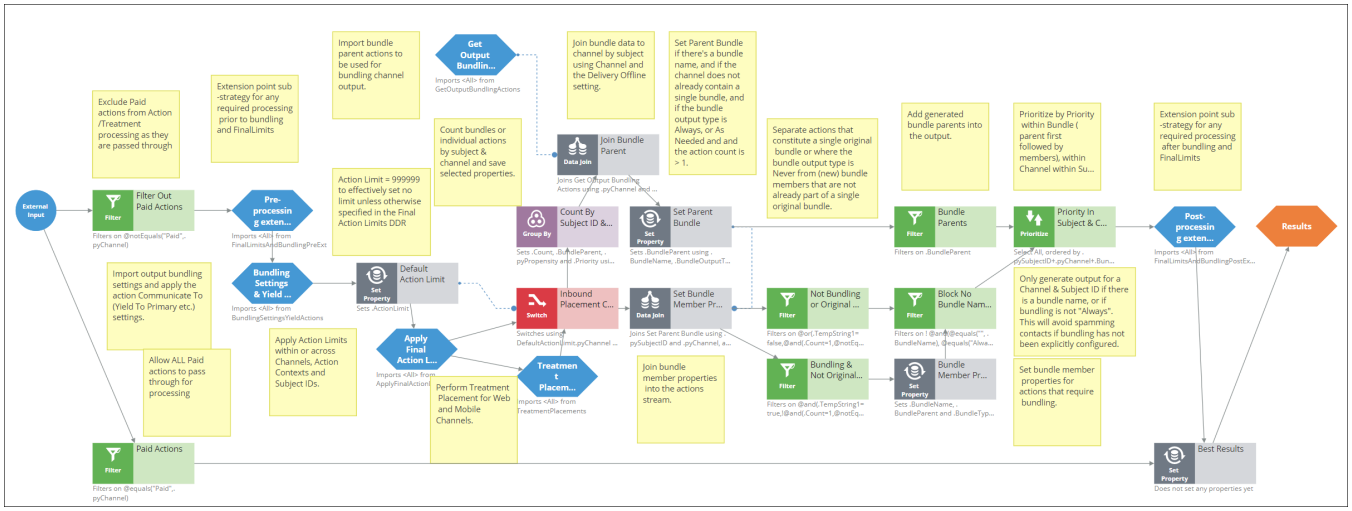
This strategy applies Output Bundling options, action Communicate To (yield to parent etc.) options, Final Action Limits and Web & Mobile Treatment Placements, after all other processing within the framework has completed and the separate context streams have been merged, but before exiting the Trigger strategy.

Pega Customer Decision Hub

Paid actions are excluded from the Action & Treatment processing as they are passed through directly to the output.

Two extension points are provided: *Final Limits and Bundling Pre Extension*, and *Final Limits and Bundling Post Extension* for adding custom functionality before and after the main processing respectively.





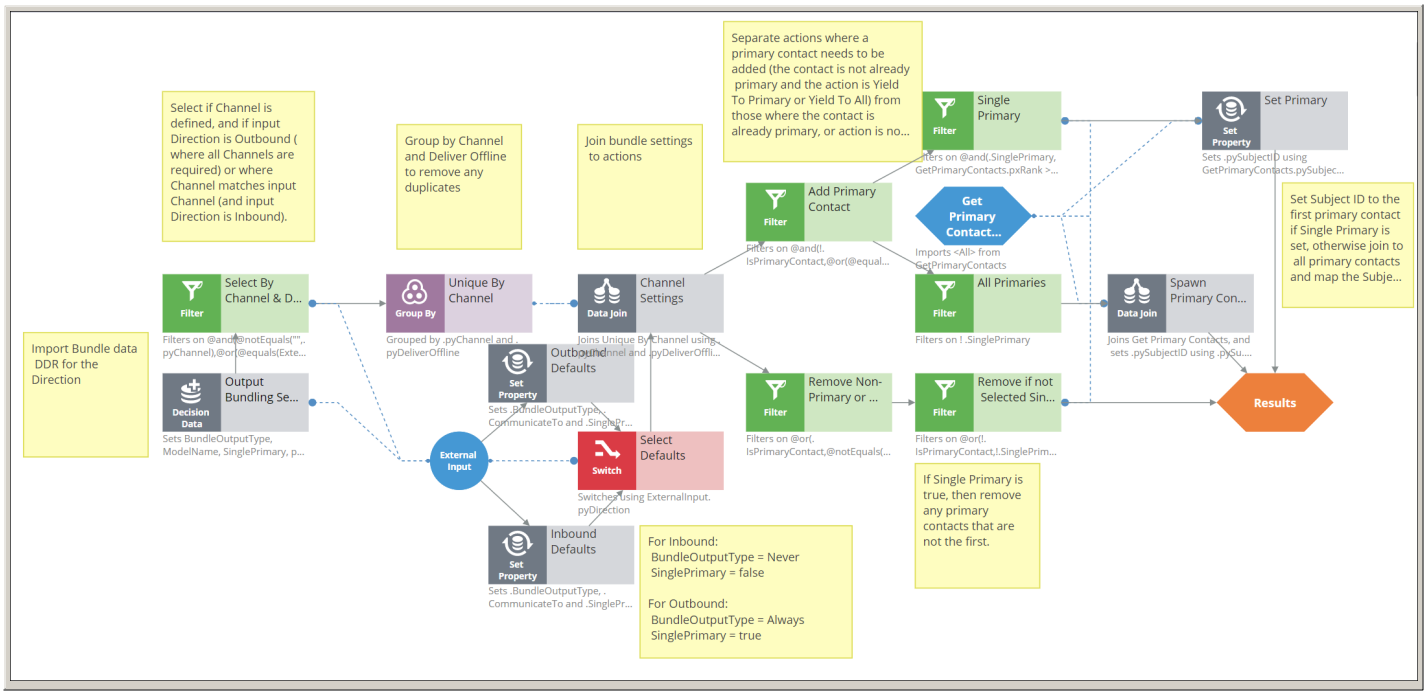
For a detailed description of the logic, refer to the annotations on the strategy.

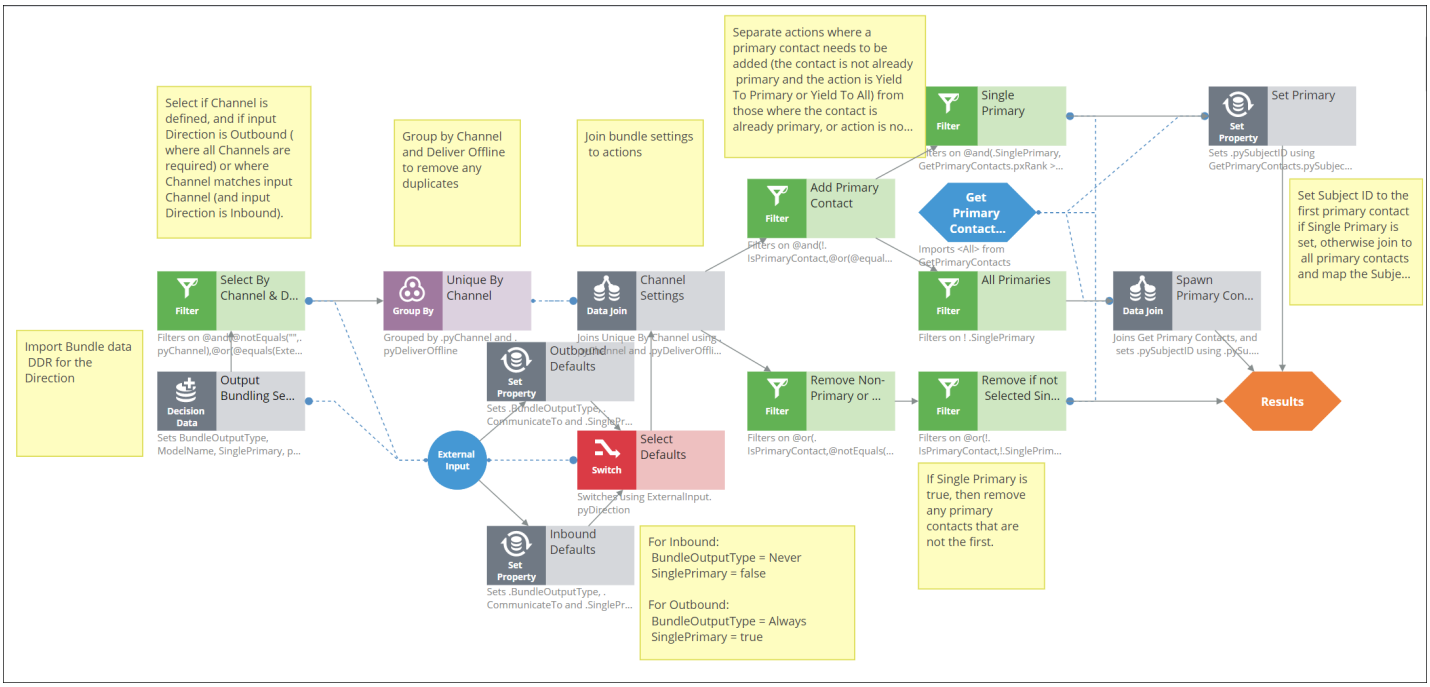
Bundling Settings & Yield Actions strategy

This strategy imports bundling settings from the Output Bundling Settings DDR and matches them to Actions by Direction, Channel and the pyDeliverOffline setting.

It also interprets the action CommunicateTo property (YieldToOriginal, YieldToAll, or YieldToPrimary) and reassigns or propagates actions to other Subject IDs as required.

Further details on the Output Bundling Settings DDR and the action CommunicateTo property can be found in *Setting Output Bundling and Primary Contact Options*.

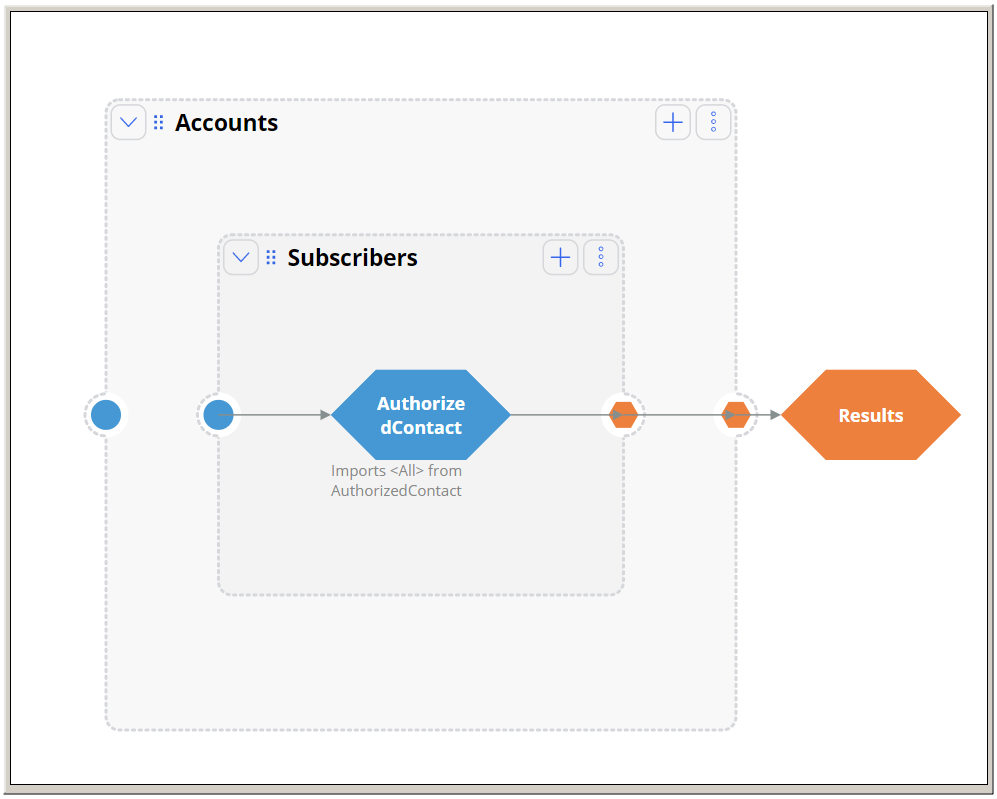


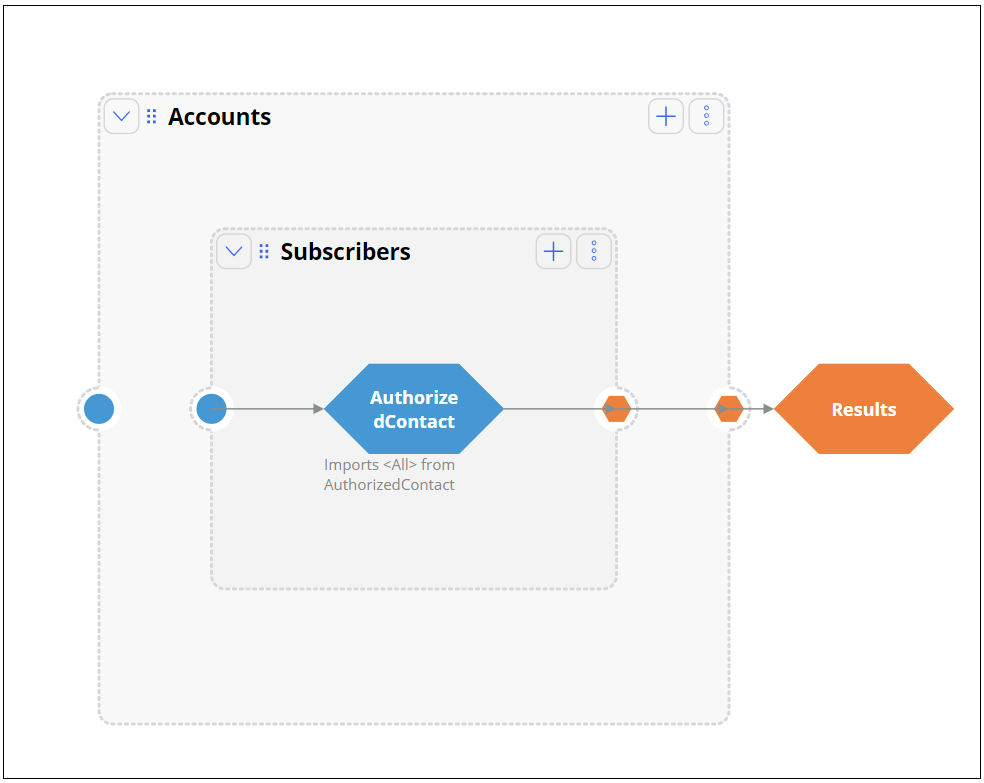


For a detailed description of the logic, refer to the annotations on the strategy.

Get Primary Contacts strategy

This strategy is generated based on the Context Dictionary configuration to call the Authorized Contact strategy from the primary context. In cases where there may be more than one authorized contact, and where one contact should take precedence over the others, the strategy should be overtaken and a Prioritize shape included prior to the exit point to ensure that the most important contact is output first.**Note:** If the Context Dictionary is updated or if the strategy is regenerated for any other reason (for example, product upgrade), the modifications may need to be re-applied.

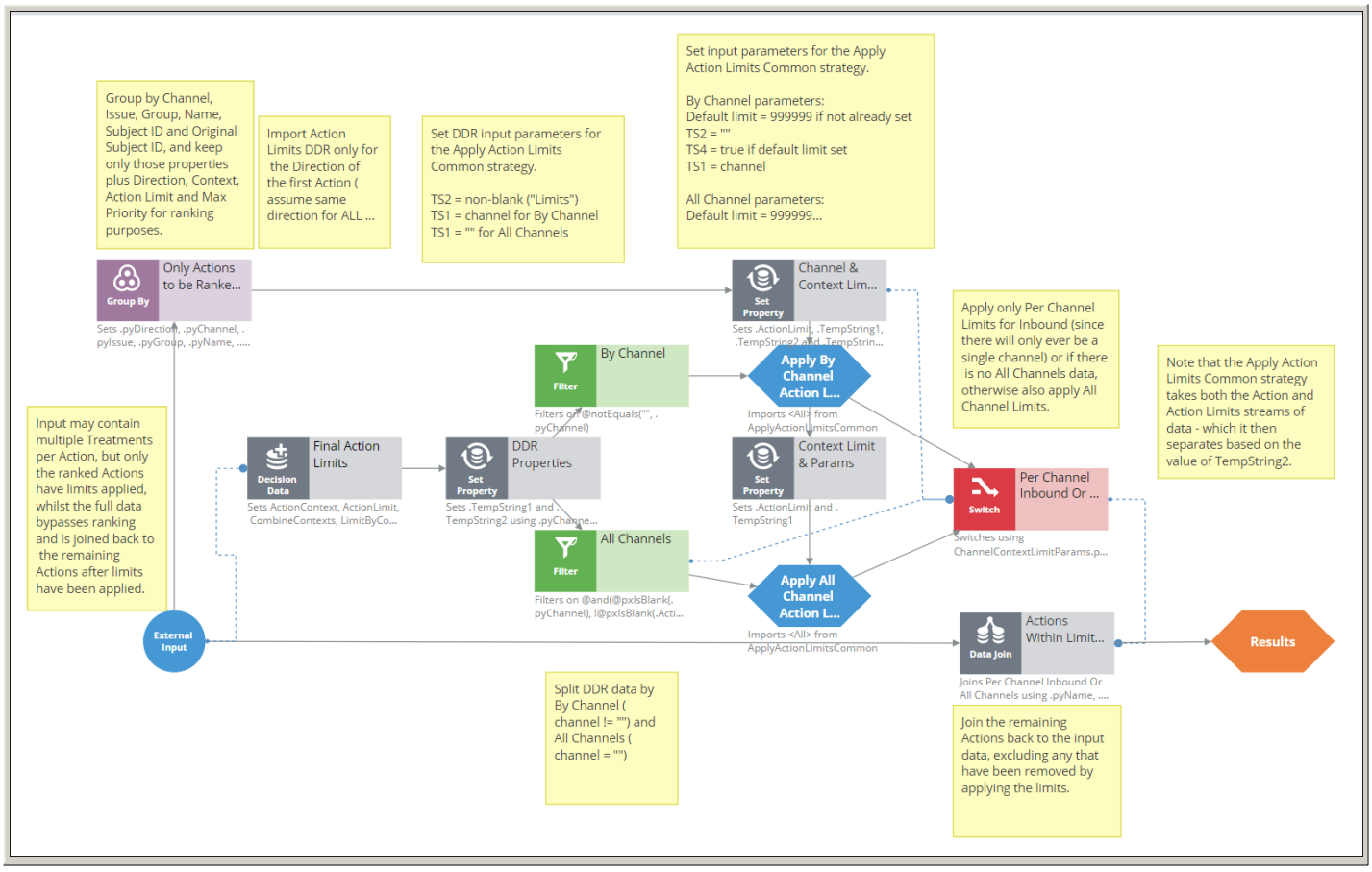


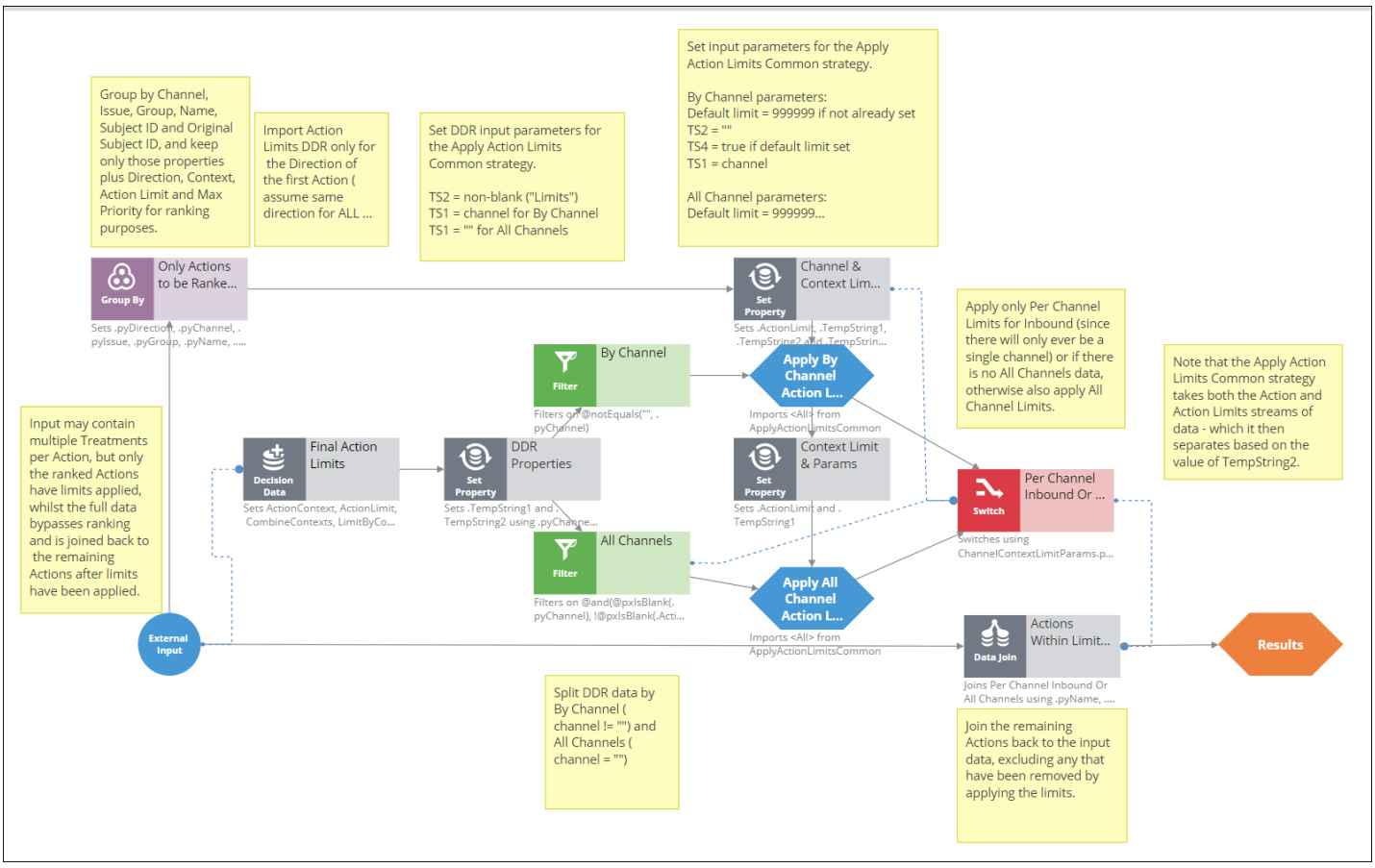


Apply Final Action Limits strategy

This strategy is ostensibly the same as the Apply Action Limits Strategy already described, with just the following differences:

- It uses the Final Action Limits DDR instead of the Action Limits DDR.
- The initial Group By adds *SubjectID* and *OriginalSubjectID*, and keeps these plus *IsPrimaryContact*, because the *LimitByContact* option allows limits to be applied at the subject level or over all subjects.
- The final Data Join adds *SubjectID* and *OriginalSubjectID* in order to allow for the *LimitByContact* option.
- There is no final Group By shape immediately prior to exiting, since any required treatment processing (other than Web or Mobile) will have already been performed.



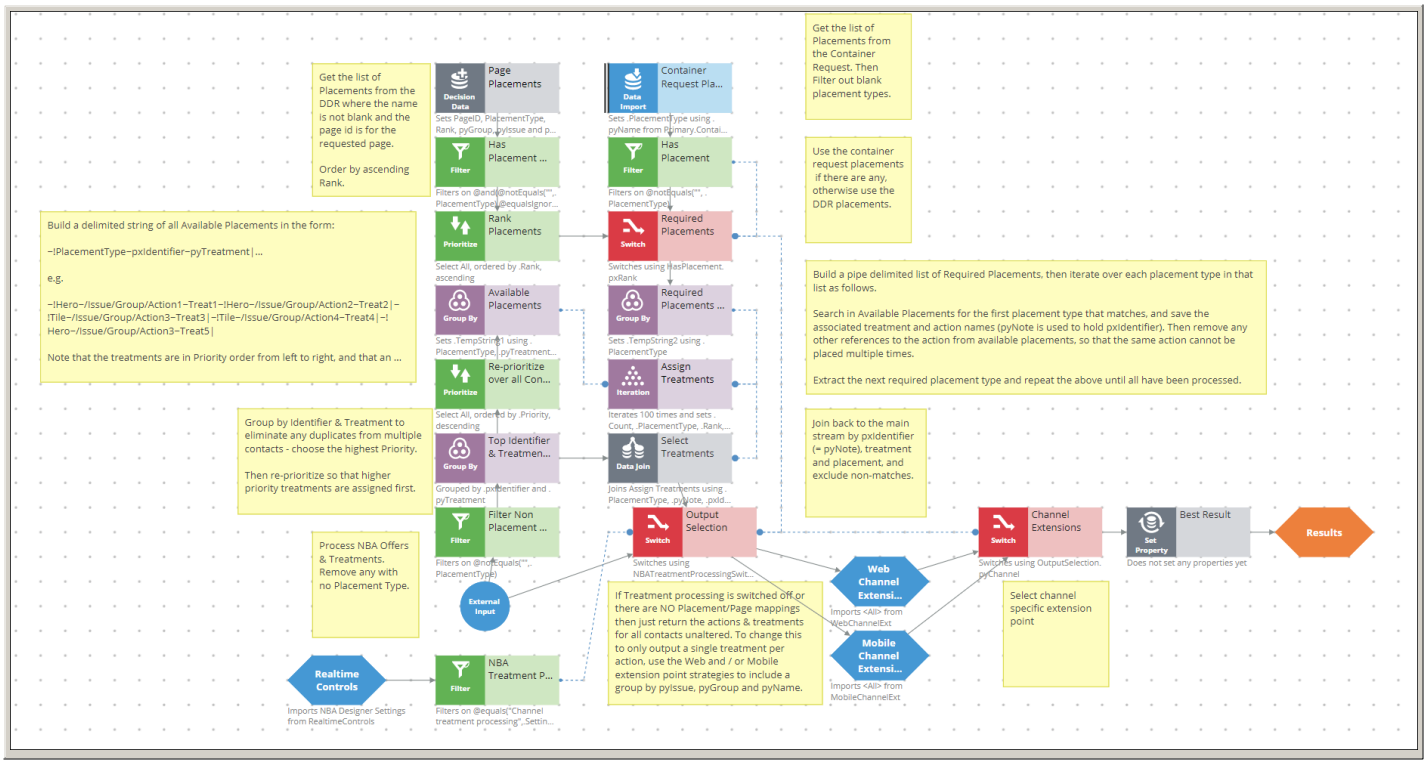


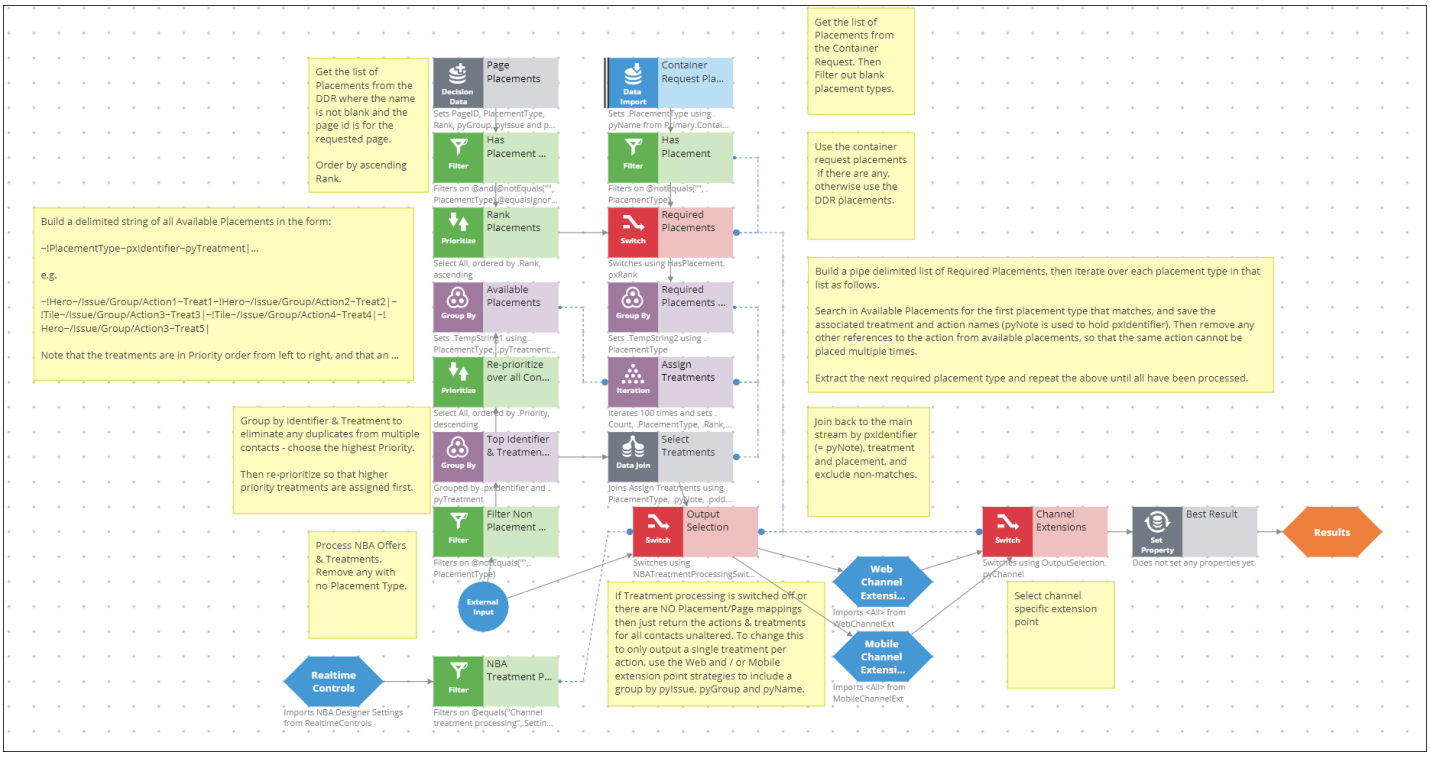
Treatment Placements strategy

The Treatment Placements strategy assigns Treatments to Placement tags for the Web and Mobile Channels based on the placement type list provided in the container call, or defaults provided in the Page Placements DDR.

The goal of the strategy is to process required placements in order - either the order in which they are specified in the call, or by their ascending rank in the Page Placements DDR - and to allocate a treatment to each successive placement in descending priority order based on its placement type.

The logic is quite complex due to the requirement that once an action's treatment has been allocated to a placement, no other treatment from that action may be allocated to any other placements. An Iteration shape is used to accomplish this recursive behavior.



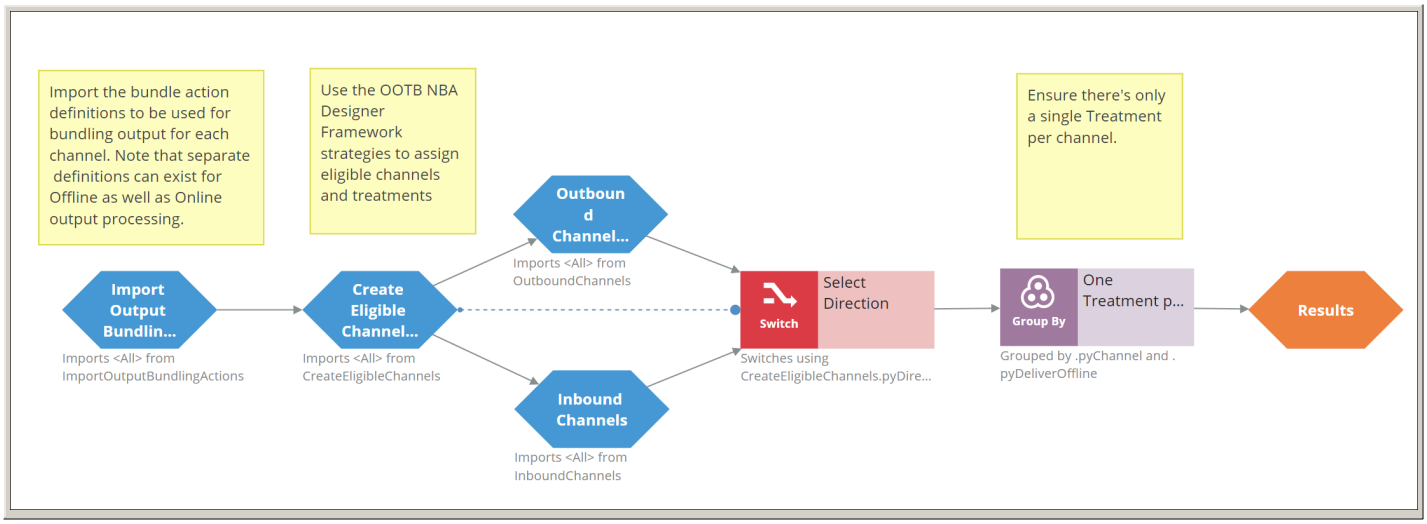


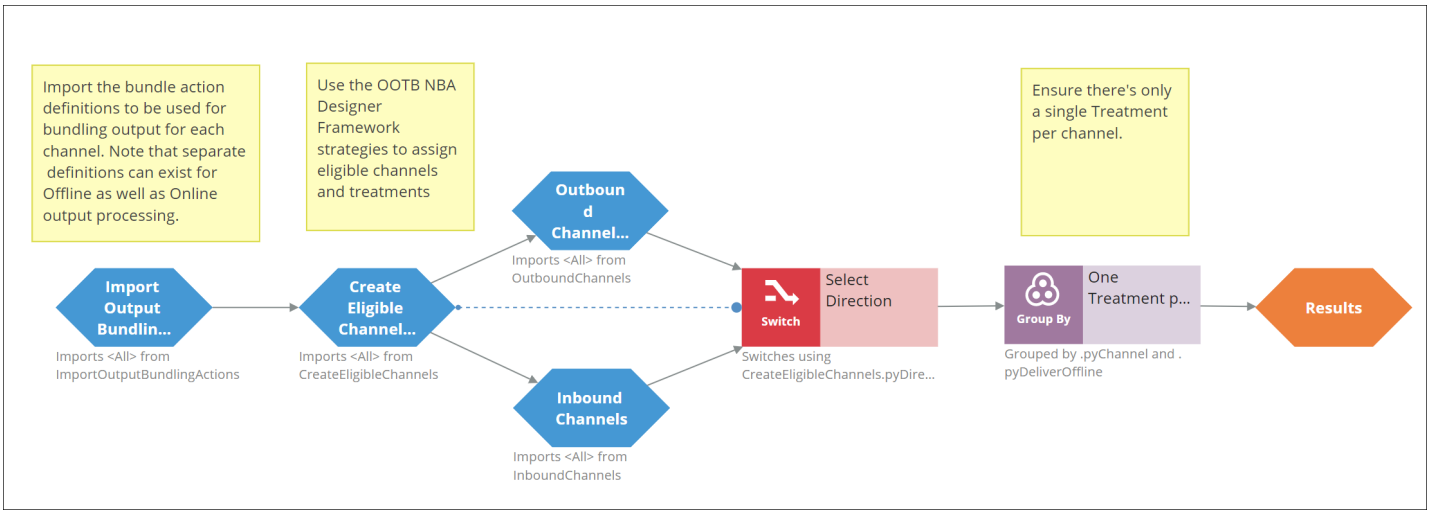
For a detailed description of the logic, refer to the annotations on the strategy.

Get Output Bundling Actions strategy

This strategy imports the bundle action definitions to be used for bundling output for each channel. Note that separate definitions can exist for Offline as well as Online output processing.

It uses the standard Next-Best-Action Designer framework strategies to assign active and eligible channels and treatments, and then uses a Group By to ensure there's only a single Treatment per channel.





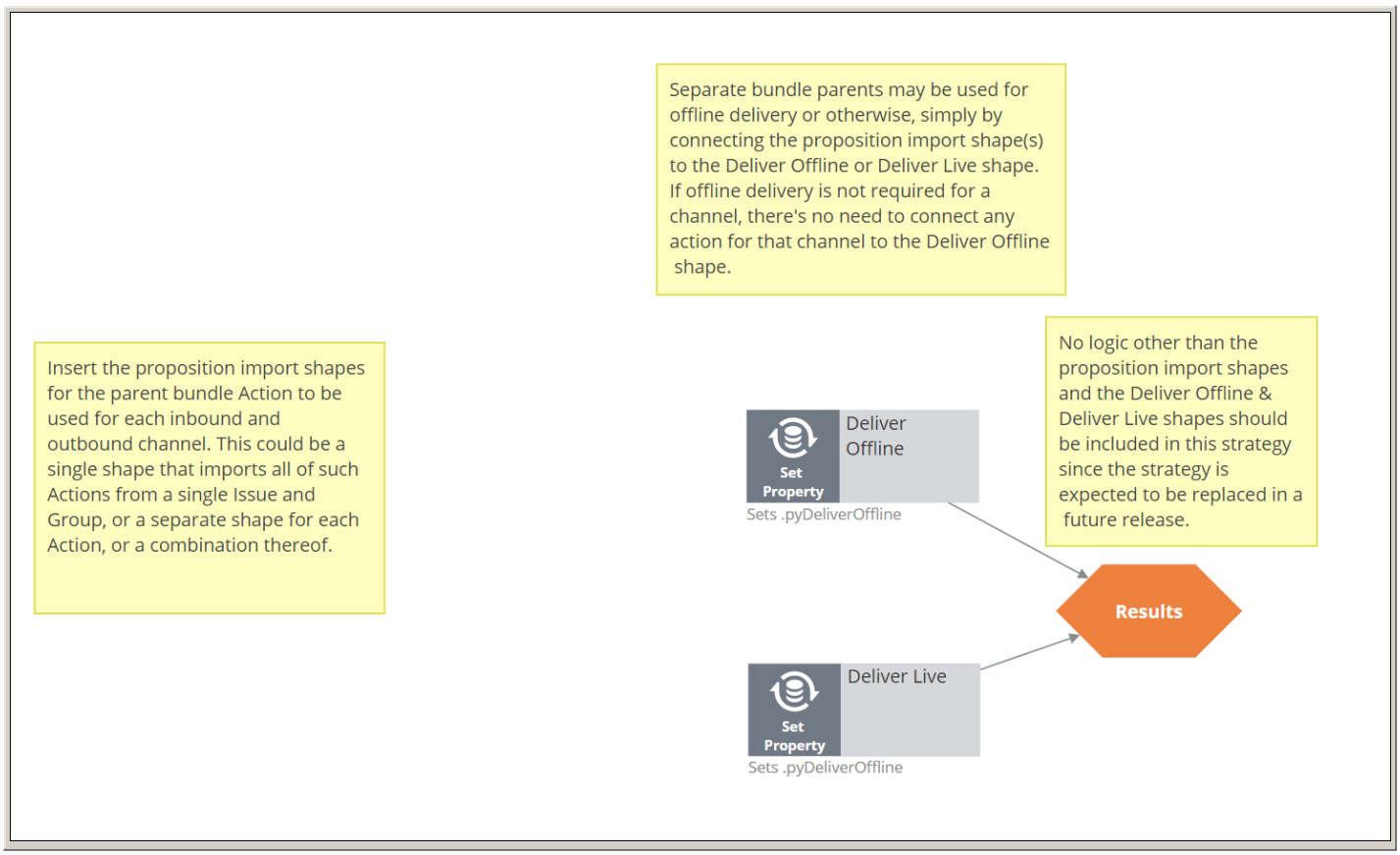
ImportOutputBundlingActions strategy

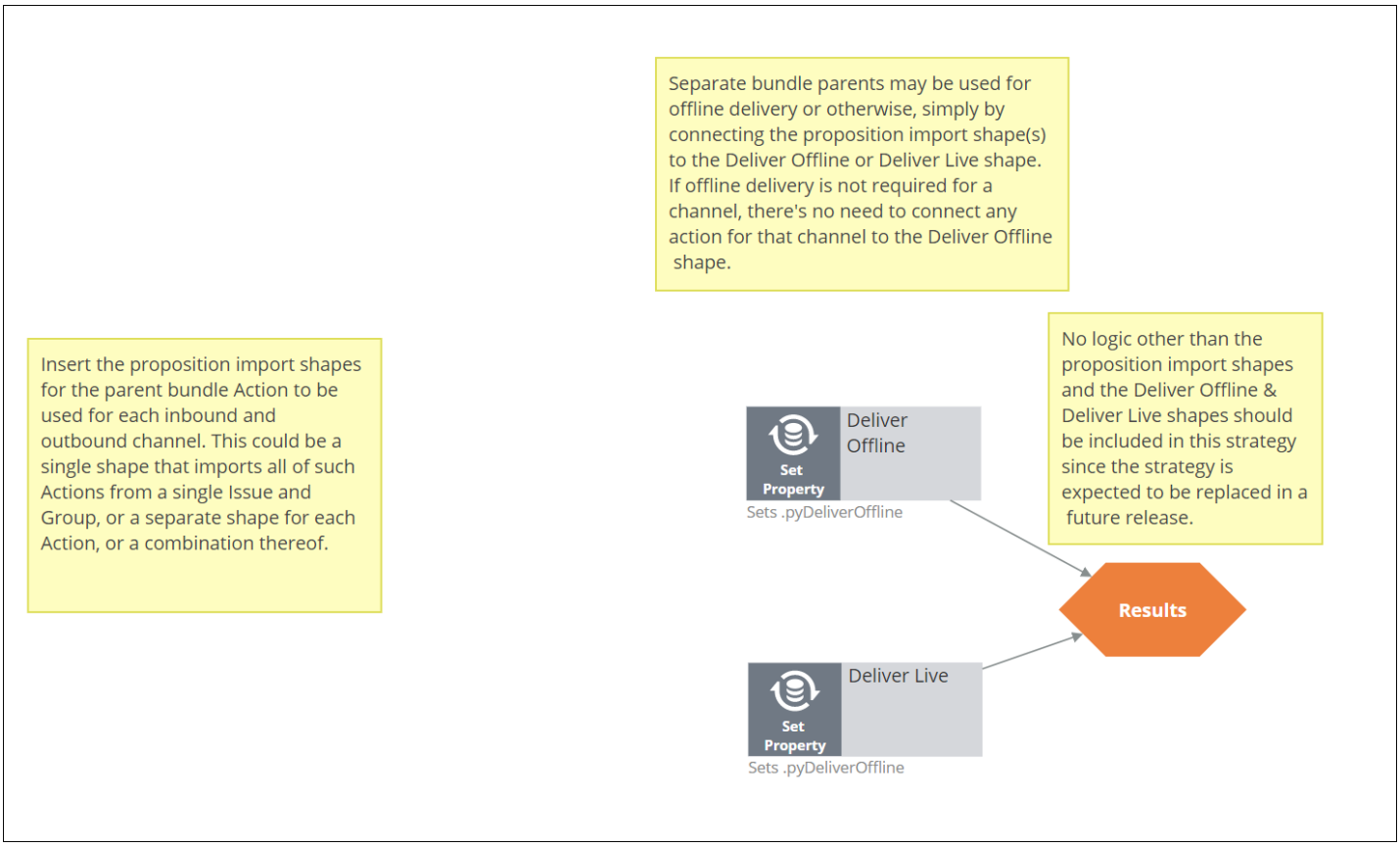
This strategy is provided as a template to be updated as per the implementation requirements.

Insert the proposition import shapes for the parent bundle action to be used for each inbound and outbound channel. This could be a single shape that imports all of such actions from a single issue and group, or a separate shape for each action, or a combination thereof.

Separate bundle parents may be used for offline delivery or otherwise, simply by connecting the proposition import shapes to the Deliver Offline or Deliver Live shape. If offline delivery is not required for a channel, there is no need to connect any action for that channel to the Deliver Offline shape.

No logic other than the proposition import shapes and the Deliver Offline & Deliver Live shapes should be included in this strategy since the strategy is expected to be replaced in a future release.





Additional NBA Strategy framework components

The following features in the Next-Best-Action Designer Framework are driven by Decision Data Rules (DDRs) which need to be configured directly by the user.

Pega Customer Decision Hub

Propensity Thresholds
: Define which actions should not be considered for arbitration.

Action Limits
: Define which actions should be eliminated after arbitration..

Arbitration Influencers
: Define how action outcomes will influence future arbitration.

Journey Upweighting
: Define weight adjustment properties for Journey Stages.

Each of these features is described in more detail in their separate sections.

- Setting Propensity Thresholds
- Setting limits on the number of actions
- Setting Output Bundling and Primary Contact options
- Using Arbitration Influencers
- Adjusting journey weight

Setting Propensity Thresholds

Propensity Thresholds are a means of setting a minimum value for the adaptive model propensity below which actions or treatments will not be considered for arbitration.

Pega Customer Decision Hub

Such thresholds can be set in a hierarchical fashion with more granular values taking precedence, the hierarchy is shown below.

- Direction
- Channel
- Issue
- Group
- (Action) Name
- Treatment

So, for example, if a threshold of 0.6 is set for the Outbound direction, and 0.7 is set for Outbound and the Email channel, 0.7 would be used for Email, and 0.6 for any other outbound channel that had not been explicitly set.

Propensity Threshold concepts

The thresholds are maintained in the Model Propensity Thresholds DDR containing the properties described below.

Property	Description
pyLabel	A user-friendly name for a threshold.
pyDirection	Propensity Thresholds are defined separately for Outbound or Inbound directions. If this is blank, then the same thresholds are applied to both Outbound and Inbound.
pyChannel	The channel to which the Propensity Threshold is to be applied. If this is blank, then the threshold is applied to all channels, however, if any channel is specified, then only channel specific thresholds will be applied, and all thresholds with a blank channel will be ignored.**Note:** If channel-specific thresholds are used, then they must be set

Property	Description
	separately for each channel. Note that this only applies within each direction, so channel could be blank for Inbound, and set individually for Outbound if desired.
pyIssue	The issue to which the threshold is to be applied. If this is blank, then the threshold is applied across all issues.
pyGroup	This is only referenced if *pyIssue* is not blank and is the group to which the threshold is to be applied. If this is blank, then the threshold is applied across all groups within the issue. If not blank, this will supersede any threshold set at the issue level.
pyName	This is only referenced if *pyGroup* is not blank and is the Action to which the threshold is to be applied. If this is blank, then the threshold is applied across all actions within the group. If not blank, this will supersede any threshold set at the issue or group levels.
pyTreatment	This is only referenced if *pyName* is not blank and is the treatment to which the threshold is to be applied. If this is blank, then the threshold is applied across all treatments for the action. If not blank, this will supersede any threshold set at the issue, group or action levels.
pyPropensity	The Propensity Threshold which sets a lower limit for the Final Propensity value of an action or treatment. Only actions and treatments with a Final Propensity greater than this threshold will be considered for arbitration within the NBA Strategy framework.

- Propensity Threshold Use Cases

Propensity Threshold Use Cases

The following Propensity Threshold use cases are examples of how to set a minimum threshold for various scenarios.

Pega Customer Decision Hub

As a user, I want to set a global minimum propensity threshold for all actions in any channel

To set a global threshold, for example, 0.55, simply set the propensity and leave all other columns blank.

pyDirection	pyChannel	pyIssue	pyGroup	pyName	pyTreatment	pyPropensity
						0.55

As a user, I want to set a minimum propensity for my inbound channels

To set the minimum propensity for any actions presented on any inbound channel to 0.6, simply set the direction and propensity.

pyDirection	pyChannel	pyIssue	pyGroup	pyName	pyTreatment	pyPropensity
Inbound						0.6

As a user, I want to set a minimum propensity for my outbound channels

To set the minimum propensity for any actions presented on any outbound channel to 0.55, simply set the direction and propensity.

pyDirection	pyChannel	pyIssue	pyGroup	pyName	pyTreatment	pyPropensity
Outbound						0.55

As a user, I want to set a minimum propensity for the Acquisition Issue for my inbound channels

To set the minimum propensity for any actions in the Acquisition Issue presented on any inbound channel to 0.6, simply set the direction, issue and propensity.

pyDirection	pyChannel	pyIssue	pyGroup	pyName	pyTreatment	pyPropensity
Inbound		Acquisition				0.6

As a user, I want to set a minimum propensity for the Acquisition Issue for my Web channel to be 0.57, and for the inbound Call Center to be 0.62

Set the direction, channel, issue and propensity for each of the Web and CallCenter channels.

pyDirection	pyChannel	pyIssue	pyGroup	pyName	pyTreatment	pyPropensity
Inbound	Web	Acquisition				0.57
Inbound	CallCenter	Acquisition				0.62

As a user, I want to set a minimum propensity for the Acquisition Issue and Devices group for my Email channel to be 0.56, and for the SMS channel to be 0.65

Set the direction, channel, issue, group and propensity for each of the Email and SMS channels.

pyDirection	pyChannel	pyIssue	pyGroup	pyName	pyTreatment	pyPropensity
Outbound	Email	Acquisition	Devices			0.56
Outbound	SMS	Acquisition	Devices			0.65

As a user, I want to set a minimum propensity for any actions in the Acquisition Issue and Devices group for my Email channel to be 0.56, except for the discounted new model top-end phone where I want the minimum propensity to be 0.7

Set the direction, channel, issue, group and propensity for the Email channel, and then override that propensity with any action specific values.

pyDirection	pyChannel	pyIssue	pyGroup	pyName	pyTreatment	pyPropensity
Outbound	Email	Acquisition	Devices			0.56
Outbound	Email	Acquisition	Devices	Phone Offer		0.7

Setting limits on the number of actions

Action Limits and Final Action Limits are two features that can be used to control the number of actions delivered to contacts. These features have a similar impact, but are implemented in different stages of the Next-Best-Action Strategy Framework as described below along with some other related concepts.

Pega Customer Decision Hub

Action Limits Concepts concepts

The two Action Limits DDRs along with some other related concepts are described below.

Concept	Description
Context entity	An entity on which next-best-action decisions can be made, for example, I want the next best action for a Customer, or the next best action for an Account.
Multilevel Context Dictionary	A configuration that has a hierarchy of context entities where actions may be relevant for specific entities, for example: • Retail Banking ◦ Customer – can have multiple Accounts ▪ Account – one per card or account • Telecommunications ◦ Account – can have multiple Subscribers ▪ Subscriber – one for each phone number, can have multiple Devices ▪ Device, for example, phone, tablet, watch • Healthcare ◦ Policy – can have multiple Members ▪ Member – can have multiple Claims ▪ Claim
Primary Context entity	The entity to which all actions will be communicated, that is, the entity representing the person with the contact details such as phone number, email address, etc. In the above examples, this would be the Customer, Subscriber or Member.
Primary Contact	An instance of the Primary Context entity that is authorized to make decisions on behalf of the account or relationship, also known as an Authorized Contact. For example, this may be a parent for a family phone plan or the primary member on a healthcare policy. Note that there may be more than one Authorized Contact for an account / relationship. The *IsPrimaryContact* property is set to true for any actions that are assigned to an authorized contact.
Authorized Contact	This is the name of a sub-strategy that is configured by the user during the implementation phase to only select Primary Contacts for an account or relationship.
Action Context	This is set on the action rule and is stored in the *ActionContext* property on the action. This is the name of the context entity for which the action is relevant, for example, a phone upgrade action would be for a Subscriber (or possibly Device), whereas a family data plan action would be for an Account, or a new credit card action would be for a Customer, whereas a credit line increase would be for an Account.
Action Limits	This is a Data Decision Rule (DDR) containing settings that are evaluated separately for each action context and for each contact (that is, a primary context, such as a Subscriber or Customer) after arbitration has occurred. This means that although a *CombineContexts* property is available for this feature, it currently has no effect, since each action context is evaluated independently from any other. Furthermore, all outbound channels are evaluated as a combined group (within Action Context and contact), and so a limit can be applied across all outbound channels, whereas for inbound channels, only a single channel will be processed during any interaction.
Final Action Limits	This is a Data Decision Rule (DDR) containing settings that are evaluated after Action Limits have been applied, and after actions for all contexts and all contacts have been arbitrated and merged. This is the final process that occurs in the NBA framework prior to output bundling. As such, limits can be applied to combinations of action contexts (using the *CombineContexts* setting), and to individual contacts, or across all contacts (using the *LimitByContact* setting). **Note:** A contact is an instance of the primary context entity with its own unique *pySubjectID*.

Action Limits DDR

Property	Description
pyLabel	A user-friendly name for an action limits setting.
pyDirection	Action limits are defined separately for Outbound and Inbound directions.
pyChannel	The channel (in combination with the action context) to which the Action Limit is to be applied. Possible settings are described below. • Each - the same Action Limit is to be applied to each channel for the Action Context. This removes the need to define a separate limit for each channel. • <Channel Name> (for example, Web, Email) - applies the Action Limit to the specific channel. This will override any value already set by the Each functionality. • <Blank> - indicates that the Action Limit is to be applied over the combination of all channels. That is, it sets a limit for the Action Context regardless of channel. This limit is applied after individual channel limits have been applied, and is only available for Outbound processing, since for Inbound only a single channel is

Property	Description
	processed at a time.
pyIssue	Not currently used.
pyGroup	Not currently used.
pyName	Not currently used.
CombineContexts	Not currently used.
ActionContext	The Action Context to which the Action Limit is to be applied.
ActionLimit	The maximum number of actions allowed for the combination of Action Context and Channel.

Final Action Limits DDR

This is the same as the Action Limits DDR but with the addition of the following two properties.

Property	Description
CombineContexts	When set to true for multiple contexts within a channel specification, this allows a single limit to be applied to the combination of selected Action Contexts.
ActionLimit	Determines whether a limit applies to each contact or to the interaction as a whole. Possible settings are described below. • true - apply the limit n to each contact separately so that the top n actions ordered by descending priority are emitted for each contact. • false - apply the limit n to the interaction so that the top n actions ordered by descending priority within contact priority are emitted for each contact. Contact priority would be determined by whether the contact is a primary contact or not, with primary contacts have a higher priority; the order for contacts with equal priority would be undefined.

Note: For the *CombineContexts* property, the Action Limit needs to be set for each Action Context being combined. If actions for only one of the contexts are available, then the limit assigned to that context will be used; if more than one of the combined contexts are available, then the maximum of the limits for those contexts is used.

In the following example, if only Account context actions were available (i.e. had passed engagement policies, arbitration etc.), a maximum of 2 (Account-context) actions would be output, although if actions for both contexts were available, up to 3 actions (Account and / or Subscriber context) would be output.

ActionContext	CombineContexts	pyChannel	pyDirection	ActionLimit
Account	true	Email	Outbound	2
Subscriber	true	Email	Outbound	3

- Action Limits Use Cases

Action Limits Use Cases

The Action Limits use cases below are examples of how these two features can be utilized. These examples use the telecommunications paradigm of Device within Subscriber within Account.

Pega Customer Decision Hub

As a user, I want to limit the number of actions returned to the top 1 across the account for the CallCenter channel

Action Limits
No limits should be defined for the Inbound CallCenter channel.
Final Action Limits
Combine contexts to apply the limit over all actions. Do not limit by contact as the limits are across the whole account.

pyDirection	pyChannel	Action Context	Combine Contexts	Action Limit	LimitBy Contact
Inbound	CallCenter	Account	true	1	false
Inbound	CallCenter	Subscriber	true	1	false
Inbound	CallCenter	Device	true	1	false

As a user, I want to limit the number of actions returned to the top 1 for each action context across the account for the CallCenter channel

Action Limits
No limits should be defined for the Inbound CallCenter channel.
Final Action Limits
Do not combine contexts to apply the limit to each action context. Do not limit by contact as the limits are across the whole account.

pyDirection	pyChannel	Action Context	Combine Contexts	Action Limit	LimitBy Contact
Inbound	CallCenter	Account	false	1	false
Inbound	CallCenter	Subscriber	false	1	false
Inbound	CallCenter	Device	false	1	false

As a user, I want to limit the number of actions returned to the top 1 per Subscriber (primary context) per action context for the CallCenter channel

Action Limits
Set the limit for each action context - do not combine contexts.

pyDirection	pyChannel	Action Context	Combine Contexts	Action Limit
Inbound	CallCenter	Account		1
Inbound	CallCenter	Subscriber		1
Inbound	CallCenter	Device		1

Final Action Limits
No limits should be defined for the Inbound CallCenter channel.

As a user, I want to limit the number of actions returned to the top 1 Account context action across the account, and the top 3 Subscriber or Device context actions per Subscriber (primary context) for the CallCenter channel

Action Limits
No limits should be defined for the Inbound CallCenter channel, since action contexts need to be combined, and this can only be done in Final Action Limits.
Final Action Limits
For the CallCenter channel, set the Account context limit to 1,do not combine,and do not limit by contact since the limit is across the whole account. The Subscriber and Device contexts are combined with a limit of 3, and need to be limited by contact as the limits are per Subscriber.

pyDirection	pyChannel	Action Context	Combine Contexts	Action Limit	LimitBy Contact
Outbound	CallCenter	Account	false	1	false
Outbound	CallCenter	Subscriber	true	3	true
Outbound	CallCenter	Device	true	3	true

As a user, I want to limit the number of actions returned to the top 1 across any outbound channel across the account

Action Limits
No limits should be defined for the Outbound channels.
Final Action Limits
Set the limit for each action context and combine contexts since only one action is required across all contexts. Do not limit by contact as the limits are across the whole account.

pyDirection	pyChannel	Action Context	Combine Contexts	Action Limit	LimitBy Contact
Outbound	Each	Account	true	1	false
Outbound	Each	Subscriber	true	1	false
Outbound	Each	Device	true	1	false

As a user, I want to limit the number of actions returned to the top 3 for the Email channel and the top 1 for the SMS channel across the account

Action Limits
No limits should be defined for the Email or SMS channels.
Final Action Limits
Set limits for each action context and combine them since the limits are across all contexts. Set separate limits for Email and SMS, do not limit by contact as the limits are across the whole account.

pyDirection	pyChannel	Action Context	Combine Contexts	Action Limit	LimitBy Contact
Outbound	Email	Account	true	3	false
Outbound	Email	Subscriber	true	3	false
Outbound	Email	Device	true	3	false
Outbound	SMS	Account	true	1	false
Outbound	SMS	Subscriber	true	1	false
Outbound	SMS	Device	true	1	false

As a user, I want to limit the number of actions returned to the top 1 Account context action and the top 2 Subscriber or Device actions for the Email channel and the top 1 Subscriber or Device action for the SMS channel across the account

Action Limits
No limits should be defined for the Email or SMS channels.
Final Action Limits

For Email, set the Account context limit to 1 and do not combine; the Subscriber and Device contexts are combined with a limit of 2. For SMS set the Account context limit to 0 (to eliminate any Account actions) and do not combine; the Subscriber and Device contexts are combined with a limit of 1.

Do not limit by contact as the limits are across the whole account.

pyDirection	pyChannel	Action Context	Combine Contexts	Action Limit	LimitBy Contact
Outbound	Email	Account	false	1	false
Outbound	Email	Subscriber	true	2	false
Outbound	Email	Device	true	2	false
Outbound	SMS	Account	false	0	false
Outbound	SMS	Subscriber	true	1	false
Outbound	SMS	Device	true	1	false

Setting Output Bundling and Primary Contact options

The Output Bundling Settings DDR contains the automatic bundling options for delivering actions - primarily on outbound channels, although the options are available for inbound channels if required.

Pega Customer Decision Hub

One of the main goals of bundling actions for outbound delivery is to avoid "spamming" individuals in on-line channels such as email, where without bundling, if multiple actions were selected for a contact, each would result in a separate email being sent.

Automatic bundling allows such multiple actions (for a single contact) to be combined into a single communication containing a predefined bundle "parent", and then each of the actions become bundle "members". However, from a Next-Best-Action Designer perspective, each bundle parent and member is output as a separate record, so a bundle including 3 bundle members would be output as 4 records, one for the parent and one for each member; action flow and treatment template processing provide the mechanism to output the bundle as a single unit.**Note:** This automatic bundling process is distinct from the predefined bundles that can be created on the Action form. If the output already

contains a single predefined bundle on a channel, then it is output as is.

Output Bundling Concepts

The Output Bundling Settings DDR and some bundling-related concepts are described below.

Intended Recipient

This is set under the Engagement policy tab on the action rule and is stored as the *CommunicateTo* property on the action. It determines to which contact or contacts an action will be delivered in a multilevel context dictionary configuration. If the configuration is not multilevel, this property will not be available.

There are three possible values:

- **Eligible contact** (**CommunicateTo**= YieldToOriginal) - deliver the action to the contact for whom it was originally assigned, that is, the one for whom the action passed the engagement policy rules. This is the default setting if *CommunicateTo* has not otherwise been set.
- **Redirect to primary contact** (**CommunicateTo**= YieldToPrimary) - deliver only to a primary contact, that is, one who is an authorized contact on the account. If the original contact is not already a primary contact the action will be redirected. An example could be for a family mobile phone plan where a child's phone is eligible for an upgrade, but the action should be sent to the parent and not the child.
- **Eligible contact and copy to primary contact** (**CommunicateTo**= YieldToAll) - deliver to the originally assigned contact and to a primary contact if the original contact is not already a primary contact. An example could be for a family mobile phone plan where a child's data usage is approaching the line limit, so the action should be sent to the parent as well as the child.

Deliver Offline

This refers to whether the outbound treatment will deliver to an offline medium such as a database table or file or whether the treatment delivers to a live channel. The setting is held in the *pyDeliverOffline* property which is true for offline delivery, or otherwise false. It is automatically set (to true or false) within the NBA Designer Framework based on the treatment definition, and separate bundle settings can be defined for each of these settings.

Output Bundling Settings DDR

Property	Description
pyLabel	A user-friendly name for a bundling setting.
pyDirection	Bundling settings are defined separately for Outbound and Inbound directions.
pyChannel	Bundling settings are defined separately for each channel.
pyIssue	Not currently used.
pyGroup	Not currently used.
pyName	Not currently used.
pyDeliverOffline	Bundling settings are defined separately for Outbound channels that may be delivered online or offline. This should be false for Inbound channels.
BundleOutputType	This determines whether or not output bundling is required for a particular channel and Deliver Offline setting and can be set to one of the following values: • Always - perform bundling for every communication on the channel, regardless of the number of actions assigned to each contact. • Never - never perform bundling for communications on the channel. This is the default for both inbound and outbound channels if no settings for the channel have been defined. • As Needed - perform bundling for a communication on the channel to a contact if more than one action is assigned to the contact.
SinglePrimary	This is only relevant in cases where it is possible to have more than one primary (authorized) contact for an account or relationship. For example, for a mobile phone family plan, parents and adult children may be authorized contacts, whereas minors are not, and for retail banking, only a single primary contact may be allowed. This setting may be set to: • true - all actions that are redirected to a primary contact (*CommunicateTo* = *YieldToPrimary* or *YieldToAll*), and were not originally assigned to a Primary Contact, will be assigned only to the preferred primary contact. All actions assigned to a Primary Contact other than the preferred Primary Contact, that have not been redirected – that is, they were originally assigned to the contact – will be discarded. This is the default for outbound channels if no settings for the channel have been defined. • false - all actions that are redirected to a primary contact (*CommunicateTo* = *YieldToPrimary* or *YieldToAll*), and were not originally assigned to a Primary Contact, will be assigned to each primary contact. Any actions originally assigned to any Primary Contact will be unaffected. This is the default setting for inbound channels if no settings for the channel have been defined.

- Output Bundling Use Cases

Output Bundling Use Cases

The following output bundling use cases are examples of how to definethese settings for various scenarios.

Pega Customer Decision Hub

As a user, I don’t want to bundle actions for any inbound channels or for the SMS channel, but I always want to bundle for email. I use a 3rd party to deliver Email and SMS

For inbound channels, I want actions intended for authorized contacts returned for all authorized contacts, but for outbound I only want those actions delivered to the preferred authorized contact.

Output Bundle Settings

No settings are required for inbound, since the defaults are not to bundle and allow all authorized contacts.

For outbound, Deliver Offline is true for Email and SMS since output will be to a file or table for delivery to the third party, Bundle Output Type is Always for Email, and Never for SMS, and Single Primary is true for both channels, since only the preferred authorized contact is required.

pyDirection	pyChannel	pyDeliver Offline	Bundle Output Type	Single Primary
Outbound	Email	true	Always	true

pyDirection	pyChannel	pyDeliver Offline	Bundle Output Type	Single Primary
Outbound	SMS	true	Never	true

Action Settings

Not required.

As a user, I want to automatically bundle actions across an account for the CallCenter channel and communicate them to a primary account holder

Output Bundle Settings

For the CallCenter channel, Deliver Offline is false since it is not relevant for inbound channels, and Bundle Output Type is Always since the actions are required to be bundled. Single Primary should be false so that actions are returned regardless of which authorized contact is the caller.

Note that any actions not assigned to an authorized contact will still be returned–but see also the Action settings.

pyDirection	pyChannel	pyDeliver Offline	Bundle Output Type	Single Primary
Inbound	CallCenter	false	Always	false

Action Settings

Intended Recipient should be set to either Redirect to primary contact or Eligible contact and copy to primary contact, depending on the nature of the action, since the requirement is to communicate them to a primary account holder.

Using Arbitration Influencers

The purpose of arbitration influencers is to use the insights gained from the outcome of presenting an action to influence the arbitration of the same action - or one or more related actions - at some future interaction.

Pega Customer Decision Hub

For example, during an interaction with a call center agent, if a customer expresses an interest in an offer, but needs to drop off the call due to other commitments, the outcome could be recorded as Maybe-Later. On the next interaction with the same customer it may be prudent to boost the chances of that same action - or some related actions – being selected as the NBA, on the assumption that the customer may still be interested and have a higher likelihood to accept the follow-up action based on their initial Maybe-Later response.

The method is similar to the Contact Policy implementation, whereby action outcomes specified by the policy are captured by the *ProcessSuppressions* Data Flow using the Behavioral Limits strategy, and then later used during an NBA interaction.

Arbitration Influencer Concepts

The Arbitration Influencers DDR and some related concepts are described below.

Concept	Description
Upweighting	The process of increasing the weighting associated with an action identified by an Arbitration Influencer policy, to increase its chances of selection during arbitration. This may be done in increments over a specified duration, starting at an initial value, and increasing until a target value is reached. The starting value will be less than the ending value, and can be negative if desired.
Decay	Similar to upweighting but where the increment is negative, that is, the weighting is decreased over a period. The starting value will be greater than the ending value, and can be negative if desired.

Output Bundling Settings DDR

Property	Description
pyLabel	A user-friendly name.
pyName	A unique name for the action influencer policy - typically set to the *ContactPolicy* (see below).
pyIssue	The issue for which the influencer policy is to be applied. This property is required.
pyGroup	The group for which the influencer policy is to be applied, or blank for all groups and actions within the issue.
Action	The Action for which the influencer policy is to be applied, or blank for all actions within a Group.
pyOutcome	The Outcome for which the influencer policy is to be applied, this must be specified.
pyDirection	The Direction for which the influencer policy is to be applied, or blank for all Directions.
pyChannel	The Channel for which the influencer policy is to be applied, or blank for all Channels.
ContactPolicy	A unique name for the action influencer policy, this must be specified.
WeightAdjustInterval	Time period over which the action weight will be adjusted in minutes.
WeightAdjustDuration	Time interval between adjustments in minutes.
WeightAdjustStartValue	The starting upweight value.
WeightAdjustEndValue	The ending upweight value (if this is less than the start then the weight will be decayed).
TargetIssue	The issue of the action to be influenced. If *UpweightActionInContext* is false this must be specified.
TargetGroup	The group of the actions to be influenced, or blank for all groups and actions within the target issue.
TargetAction	The name of the action to be influenced, or blank for all actions within a target group.
UpweightActionInContext	If true, use the action(s) identified by the source configuration as the target action(s), otherwise use the action(s) identified by the target configuration.

The weight adjustment properties allow the action Weight to be adjusted so that if Action weighting is enabled in the **Arbitration** tab of Next-Best-Action Designer. This will affect the overall priority of the target actions. The detailed weight adjustment calculations are shown below.

Property	Description	Formula

Property	Description	Formula
pyLastResponseTime	The time when the outcome that triggered the action influencer occurred.	.TimeInCurrentStage = @DateTimeDifference(.pyLastResponseTime, @getCurrentTimeOfDayStamp(), "m") .WeightAdjustInterval = @if(.WeightAdjustInterval > 0, .WeightAdjustInterval, 1)
Quantity	Total number of weight duration intervals.	.Quantity = @if(.WeightAdjustDuration > 0, @divide(.WeightAdjustDuration, .WeightAdjustInterval, 0), 1)
TempDouble	The weight increment per interval, negative if start weight is greater than end weight.	.TempDouble = @divide((.WeightAdjustEndValue - .WeightAdjustStartValue), .Quantity, 2)
pyWeight	The incremental upweighting value for the target actions, note that this could be negative.	.pyWeight = @if(.TimeInCurrentStage > .WeightAdjustDuration, .WeightAdjustEndValue, .WeightAdjustStartValue + (@divide(.TimeInCurrentStage, .WeightAdjustInterval, 0, "down") * .TempDouble))

The resultant *pyWeight* is then added to the action Weight property for use in the arbitration formula.

Adjusting journey weight

Similar to the weight adjustment for Arbitration Influencers, the weighting of actions associated with a Journey Stage may also be adjusted.

Pega Customer Decision Hub

The Journey Stages DDR is populated by the UI during Journey Stage editing, however, the editing of the weighting properties will not be available via the UI until a later release, although these properties are included in the DDR for manual editing by advanced users if required.

The properties are the same as those for Arbitration Influencers and are shown below.

Property	Description
WeightAdjustInterval	Time period over which the action weight will be adjusted in minutes.
WeightAdjustDuration	Time interval between adjustments in minutes.
WeightAdjustStartValue	The starting upweight value.
WeightAdjustEndValue	The ending upweight value (if this is less than the start then the weight will be decayed).

The weight adjustment properties allow the action Weight to be adjusted so that if Action weighting is enabled in the **Arbitration** tab of Next-Best-Action Designer. This will affect the overall priority of the Journey Stage actions. **Note:** If journey upweighting is not required, ensure that *WeightAdjustStartValue* and *WeightAdjustEndValue* are both set to zero (or left blank).The detailed weight adjustment calculations are shown below.

Output Bundling Settings DDR

The weight adjustment properties allow the action Weight to be adjusted so that if Action weighting is enabled in the **Arbitration** tab of Next-Best-Action Designer. This will affect the overall priority of the target actions. The detailed weight adjustment calculations are shown below.

Property	Description	Formula
pyLastResponseTime	The time when the outcome that triggered the journey weight adjustment occurred.	.TimeInCurrentStage = @DateTimeDifference(.StageStartDateTime, @getCurrentTimeOfDayStamp(), "m") .WeightAdjustInterval = @if(.WeightAdjustInterval>0, .WeightAdjustInterval, 1)
Quantity	Total number of weight duration intervals.	.Quantity = @if(.WeightAdjustDuration > 0, @divide(.WeightAdjustDuration, .WeightAdjustInterval, 0), 1)
Value	The weight increment per interval, negative if start weight is greater than end weight.	Value = @divide((.WeightAdjustEndValue - .WeightAdjustStartValue), .Quantity, 2)
JourneyWeight	The incremental upweighting value for the target actions, note that this could be negative.	.JourneyWeight = @if(.TimeInCurrentStage > .WeightAdjustDuration, .WeightAdjustEndValue, .WeightAdjustStartValue + (@divide(.TimeInCurrentStage, .WeightAdjustInterval, 0, "down") * .Value))

The resultant *JourneyWeight* is then added to the action Weight property for use in the arbitration formula.

NBA Strategy framework extension points

The Next-Best-Action strategy has a number of predefined extension points that you can use to add your own business logic to the framework. For example, you can use extension points to apply additional contact policies, or adjust model propensities. For a list of available extension points, see below.

Pega Customer Decision Hub

Understanding the naming convention for extension points

Component names for extension points use the following notation:

AllIssues
: Includes all issues from the Next-Best-Action Designer taxonomy.

AllGroups
: Includes all groups within one or all issues from the Next-Best-Action Designer taxonomy.

<Issue>
: The name of an issue in the Next-Best-Action Designer taxonomy.

<Group>
: The name of a group in the Next-Best-Action Designer taxonomy.

<Hierarchy_Level>
: Specific issues and groups from the Next-Best-Action Designer taxonomy. Not all of the below combinations are available for all strategies.

- TopLevel - all groups within all issues
- <Issue> - all groups within the issue
- <Issue>_<Group> - a specific issue and group
- <Issue>_AllGroups - all groups within the issue
- AllIssues_AllGroups - all groups within all issues

Note: Since groups are individually defined for each issue, groups defined for different issues are distinct rules, even if they have the same name. Because of that, AllIssues_<Group> is not a valid combination.

<Context>
: The name of a context entity from the Context Dictionary, for example Customer, Account, or Subscriber.

<Channel>
: The name of a channel, for example Web, Email or SMS. Both built-in and custom channels are supported.

<Direction>
: Inbound or outbound.

<Triggering Strategy>
: Depending on the combination of issues and groups to be processed, and the method of initiation, for example, real-time container, outbound schedule, or event trigger,

the Next-Best-Action Designer strategy can be invoked from the following strategies:

- H_NBA_<Issue>
- NBA_<Issue>
- NBA_<Issue>_<Group>
- NBA_TopLevel
- Trigger_H_NBA_<Issue>
- Trigger_NBA_<Issue>
- Trigger_NBA_<Issue>_<Group>
- Trigger_NBA_TopLevel

->

Indicates the calling strategy hierarchy for the parent strategy. Each successive strategy is called from the previous one in the list.**Note:** Some extension points are referenced more than once, in which case they will have multiple parent strategies.

Extension points in a multilevel implementation

In a multilevel context implementation, the Trigger strategy will include embedded sub-strategies that iterate over each context instance within the parent context, e.g. for a telecommunications implementation this may be Device within Subscriber within Account, whilst for a financial implementation it may be Customer within Account.

The NBA Strategy Framework is invoked for separately for each such context, so that Subscriber context actions will be processed separately from Account context actions, etc.

This needs to be borne in mind when using the extension strategies that are subordinate to the NBA Strategy Framework, since only actions from a single context will be processed during execution. If any logic is required that spans action context - that is, requires the actions from different contexts to be in the same SR, then use one of the two *FinalLimitsAndBundlingPreExt* and *FinalLimitsAndBundlingPostExt* strategies that are invoked after all action contexts have been merged.

Required strategy extensions

Required extension points must always be configured.

Extension point strategy name	Parent strategies	Purpose	Business use case
AuthorizedContact	<Triggering Strategy> <Triggering Strategy>-> FinalActionLimitsAndBundling-> BundlingSettingsYieldActions -> GetPrimaryContacts	In a multilevel context implementation, insert logic to only select primary context records that are authorized contacts. When executed from the triggering strategy, the property *IsPrimaryContact* will be set to true for each of these.	Use this to designate contacts as being the preferred or authorized contacts within a parent entity such as Household or Account, e.g. head of household, or primary account holder. *IsPrimaryContact* may be used in Engagement Policies to select actions only for these contacts, and the Communicate To property on the action form may be used to redirect actions to these contacts.

Optional strategy extensions

For extension points listed in this section, the strategy is empty, and you can configure it as needed to suit your requirements, or leave it blank.

Extension point strategy name	Parent strategies	Purpose	Business use case
NBA_<Issue>_<Group>_Ext	Various-> NBA_<Issue>_<Group>_AllActions	Apply any additional logic to all actions in the named <Issue> and <Group> after the Eligibility, Applicability and Suitability conditions have been evaluated. This logic is executed at the highest context level in the Context Dictionary, regardless of action context, and before the context specific engagement policies.	When migrating to Next-Best-Action Designer, any existing issue or group-specific action eligibility strategies or proposition filters may be included here to augment the default proposition filters. This allows existing artifacts to be re-used and gradually phased out new engagement policies are built in Next-Best-Action Designer. Typically an engagement policy created within Next-Best-Action Designer will not require any additional logic.
NBA_<Issue>_<Group>_<Context>_Ext	NBA_<Issue>_<Group> -> NBA_<Issue>_<Group>_<Context> <Triggering Strategy> -> NBA_TopLevel_<Primary Context>-> H_NBA_<Issue>_<Context>-> NBA_<Issue>_<Group>_<Context> H_NBA_<Issue> -> H_NBA_<Issue>_<Context> -> NBA_<Issue>_<Group>_<Context>	Apply any additional logic to all actions for the named <Context> in the named <Issue> and <Group> after the Eligibility, Applicability and Suitability conditions have been evaluated. This logic is executed only in the context of the named <Context>, but not the primary context.	When migrating to Next-Best-Action Designer, any existing issue or group-specific action eligibility strategies or proposition filters may be included here to augment the default proposition filters. This allows existing artifacts to be re-used and gradually phased out as new engagement policies are built in Next-Best-Action Designer. Typically an engagement policy created within Next-Best-Action Designer will not require any additional logic.
NBA_AllIssues_AllGroups_Ext	Various-> NBA_AllIssues_AllGroups_<Context>	Apply any additional logic to all actions in all issue and group combinations after the Eligibility, Applicability and Suitability conditions have been evaluated. The same logic is executed for actions of any context level based on the context of the parent strategy, so if action context is significant, this needs to be accounted for within the logic.	When migrating to Next-Best-Action Designer, any existing generic action eligibility strategies (that apply to all issues and groups) may be included here to augment the default proposition filters. This allows existing artifacts to be re-used and gradually phased out new engagement policies are built in Next-Best-Action Designer. Typically an engagement policy created within Next-Best-Action Designer will not require any additional logic.
		Logic can be inserted at the start of the Next-Best-Action strategy framework	This effectively achieves the same purpose as the *NBA_AllIssues_AllGroups_Ext* extension, except that it is executed only once after actions from all issues and groups have been merged, rather than separately for

Extension point strategy name	Parent strategies	Purpose	Business use-case
NBA Pre-Process Extension Point	Various-> NBA Strategy Framework	before any other logic is executed. This is after all engagement policies and any of the above extensions have been executed, and actions from all issues, groups, and contexts have been merged. This is a catch-all extension point to allow any global logic to be applied to all actions after the initial engagement policies have been evaluated.	each issue or group. Use this to augment action data - e.g. from one or more DDRs, or derive any calculated properties that may be required later in the Next-Best-Action Designer logic. However, note that if the data is not required as part of the scoring, treatment processing, arbitration or channel processing sections, then this may be better done later in the Next-Best-Action Framework after the number of actions has been reduced.
Create Eligible Channel Extension	<Triggering Strategy>-> NBA Strategy Framework -> Action Level Propensity -> Create Eligible Channels	Used for additional channel processing or validation. Note that this will be in addition to the default channel validation - it does not replace it, i.e. it is intended only to process channels that would otherwise not be selected by the default logic.	The Create Eligible Channels strategy that calls this extension validates and assigns available default channels to actions. If additional channels need to be defined, it is recommended that these are added to the Other category to allow the default logic to also validate and assign these. However, if more granular control is needed for the additional channels than is provided by the default Other channel, then additional logic may included here. Use the logic in the Create Eligible Channels strategy as a template for building the extension logic.
ActionLevelPropensityExt	<Triggering Strategy>-> NBA Strategy Framework-> Action Level Propensity	Logic can be inserted after action scoring has occurred, for example, to include addition models or adjust the model propensity. Note that this extension is not executed for transactional actions.	This extension is largely redundant for model related changes since the concept of modifiable Predictions was introduced and model related logic can be managed within these predictions. However, this could be used to introduce A/B testing or Champion Challenger functionality at the action level if required.
OutboundPreferencesExtension	<Triggering Strategy>-> NBA Strategy Framework-> TreatmentsChannels-> OutboundChannels-> OutboundChannelPreferences	Addition logic can be inserted after the treatment data has been retrieved, but before treatment scoring is performed. **Note:** An extension point is not available for Inbound channels.	This extension applies to all outbound channels except for Paid and is executed after treatments for all outbound channels have been merged. It can be used to apply channel preferences or arbitration across channels, e.g. using an adaptive model or customer preferences to prioritize channels. **Note:** This is not relevant to inbound channels, as only a single channel is ever present for an inbound interaction.
AIModelsExt	<Triggering Strategy>-> NBA Strategy Framework-> TreatmentsChannels-> TreatmentLevelPropensity-> PredictTreatmentPropensity	Logic can be inserted after treatment scoring has occurred, but prior to outbound model maturity processing, for example, to include addition models or adjust the model propensity. **Note:** This extension is not executed for transactional actions.	This extension is largely redundant for model-related changes since the concept of modifiable predictions was introduced and model related logic can be managed within these predictions. However, this could be used to introduce A/B testing or Champion Challenger functionality at the treatment level if required.
TreatmentLevelPropensityExt	<Triggering Strategy>-> NBA Strategy Framework-> TreatmentsChannels-> TreatmentLevelPropensity	Logic can be inserted after treatment scoring and outbound model maturity processing has occurred, for example, to include addition models or adjust the model propensity. **Note:** This extension is not executed for transactional actions.	The only difference between this and the above extension point is that it executed after outbound model maturity has been evaluated, and so some immature actions may have been eliminated.
ControlGroupImpl	<Triggering Strategy>-> NBA Strategy Framework	Logic can be inserted to determine whether the contact is part of a custom control group. If yes, the additional *InControlGroup* extension strategy is executed instead of all of the prior next-best-action processing, including action imports and validation.	Use this strategy to identify customers in universal control groups and to use a custom strategy *InControlGroup* (see below) to generate the output for such customers.**Note:** Note that this could also be used to detect any other alternate behavior required for a customer that requires output other than that generated from the regular next-best-action processing.
InControlGroup	<Triggering Strategy>-> NBA Strategy Framework	This strategy is provided as an alternative action selection mechanism if the *ControlGroupImpl* extension strategy identifies the current contact (customer) as being in a control group. **Note:** The **External input** option is disabled for this strategy, so it does not import any actions from the Next-Best-Action Designer framework and must generate any output required from the framework.	Used in conjunction with ControlGroupImpl above. Use this strategy to generate the output required for customers identified in the ControlGroupImpl strategy, e.g. in a universal control group, or requiring some other alternative processing. For example, for a control group this could be a dummy action that will be persisted to Interaction History in order to identify the customer as being in the control group, but that is not emitted to the channel. It could also include an alternative set of actions and treatments required by a specific group of customers.

Extension point strategy name	Parent strategies	Purpose	Business use case
CalculateValueExtension	<Triggering Strategy>-> NBA Strategy Framework-> Arbitration-> Calculate Business Value	This strategy can be modified using the **Action value** feature from the **Arbitration** tab in Next-Best-Action Designer. It allows the Action Value property to be calculated for all actions, where V = Value in the P x C x V x L calculation..	Typically, the Value property assigned to the action would be used as V in the P x C x V x L arbitration calculation. Use this strategy to adjust or derive the Value property based on customer or other Strategy Result properties.
QnAContextWeightingExt	<Triggering Strategy>-> NBA Strategy Framework-> Arbitration-> QnAContextWeighting	Add any QnA-related logic in this extension strategy to update the *.ContextWeight* property. Changes should be additive to this property, that is, they should take the form of ContextWeight + myContextWeight.	This extension is intended to cater for any Question and Answer data obtained from inbound interaction dialogs where the responses may be used to adjust the context weighting.
Process<Direction><Channel>Channel	<Triggering Strategy>-> NBA Strategy Framework-> ChannelProcessing -> OutboundChannelProcessing	For outbound channels only.**Note:** For channels that are exclusively outbound (for example, Email, SMS) the <Direction> component is omitted from the strategy name. These strategies allow for outbound channel specific logic to be applied prior to the default channel processing performed by the *TopOfferOrBundleForOutbound* strategy.	It is understood that the generic default logic for all outbound channels provided in the *TopOfferOrBundleForOutbound* strategy may not fully meet local requirements, and so these strategies are provided to augment the default functionality. The Action Limits DDR described in Setting Limits on the number of Actions will obviate any need to directly set output limits in these strategies, but these extension points can be used for applying custom logic or setting additional channel specific properties for output.
<Channel>InboundChannelExt	<Triggering Strategy>-> NBA Strategy Framework-> ChannelProcessing -> InboudChannelProcessing-> <Channel>InboundChannel	For inbound Assisted and Other channels only. These strategies allow for additional logic to be applied after all default channel processing has occurred.	The Action Limits DDR described in Setting Limits on the number of Actions will obviate any need to directly set output limits for these channels, but these extension points can be used for applying custom logic or setting additional channel specific properties for output.
ContactPolicyExtension	<Triggering Strategy>-> NBA Strategy Framework-> ChannelProcessing -> OutboundChannelProcessing-> ProcessOutboundChannels-> TopOfferOrBundleForOutbound-> OutboundLimits	Allows for addition contact policy rules to be applied after the default rules have been executed. **Note:** The Contact Policy is not applied to transactional actions.	The default Contact Policy definitions should cater for most requirements, but any local policies than are not possible using default functionality may be included here. **Note:** The default rules cannot be replaced, but are augmented by any custom logic.
FallbackNBAStrategy	<Triggering Strategy>-> NBA Strategy Framework	This strategy is provided as an alternative action selection mechanism if the **Next-Best-Action active** switch in the Next-Best-Action Designer Settings DDR is set to **false** - in other words, if Next-Best-Action is turned off. **Note:** The **External input** option is disabled for this strategy, so it does not import any actions from the Next-Best-Action Designer framework and must generate any output required from the framework.	Use this to provide a default set of actions and treatments in the event that the regular next-best-action strategies need to be disabled temporarily.
NBAPostProcessExtension	<Triggering Strategy>-> NBA Strategy Framework	This is the final stage of the Next-Best-Action Strategy Framework, after all context level action processing has occurred and before actions from different contexts are merged.	While the channel extension strategies described above are useful for implementing any channel specific custom logic, this extension point allows additional logic to be included that spans all channels. However, note that this will only be applied to actions from a single context at a time - if actions from multiple contexts need to be considered, then use one of the Final Limits extension points described below.
FinalLimitsAndBundlingPreExt	<Triggering Strategy>-> FinalActionLimitsAndBundling	This is executed before actions for all contexts have been processed by the Next-Best-Action Designer framework, and before any outbound bundling or final action limits are imposed.	There is no specific use case for this extension point, but it is provided to allow for any additional logic to be inserted after all actions for all contexts have been processed and merged and before final limits and bundling processing is performed.
<Channel>ChannelExt	<Triggering Strategy>-> FinalActionLimitsAndBundling-> Treatment Placements	For inbound Web and Mobile channels only. These strategies allow for additional logic to be applied after all processing of default channels has occurred.	It is understood that the default logic for these channels may not fully meet local requirements, and so these strategies are provided to augment the default functionality. The Action Limits and Final Action Limits DDRs described in Setting Limits on the number of Actions will obviate any need to directly set output limits in these strategies, but these extension points can be used for applying custom logic or setting additional channel specific properties for output.
FinalLimitsAndBundlingPostExt	<Triggering Strategy>-> FinalActionLimitsAndBundling	This is executed before actions for all contexts have been processed by the Next-Best-Action Designer framework, and before any outbound bundling or final action limits are imposed. It is the final	There is no specific use case for this extension point, but it is provided to allow for any additional logic to be inserted after all Next-Best-Action Designer logic

Extension point strategy name	Parent strategies	Purpose	Business use case
		stage of the Next-Best-Action Designer framework strategies.	has been applied for all actions for all contexts

Modifiable strategy extensions

Extension points listed in this section are fully functional out of the box, but you can still modify the logic if it is critically important to your business needs.

Extension point strategy name	Parent strategies	Purpose	Business use case
JourneyWeightAndPredictorsExt	<Triggering Strategy>-> NBA Strategy Framework-> Customer Journeys-> Journey Weight And Predictors	Use this strategy to include any custom logic for setting journey properties or calculating journey weights.	Only use this if the default weight calculations do not suit your needs, or if you want to prepare additional journey related predictors for the models. Bear in mind that any additional model predictors will also need to be included as parameters for the models and so will require the models to be updated, and the generated strategies that call them.
Handle Model Maturity	<Triggering Strategy>-> NBA Strategy Framework-> Action Level Propensity -> Predict Action Propensity <Triggering Strategy>-> NBA Strategy Framework-> TreatmentsChannels -> TreatmentLevelPropensity-> PredictTreatmentPropensity	The values in the **Model Maturity Threshold** and **Population Percentage Set Property** shapes may be changed to suit local requirements. The default values are 200 and 2% respectively. Model Maturity Threshold is the number of positive responses above which the model is deemed to be mature. Population Percentage is the initial percentage of the population that will receive actions for which the models are immature. This will increase towards 100% as the number of positive responses approaches the Model Maturity Threshold.	Only modify this strategy if the default values are not suitable.
<Channel>ChannelStrategy	<Triggering Strategy>-> NBA Strategy Framework-> TreatmentsChannels-> OutboundChannels <Triggering Strategy>-> NBA Strategy Framework-> TreatmentsChannels-> InboundChannels	These strategies may need to be updated if additional treatment metadata needs to be added to the treatment definitions, for example if the *<Channel>Treatments* Decision Data Rule has been extended.	These are default strategies that assign treatments to actions separately for each channel. They are not extensions as such, but are able to be modified to suit local requirements. If additional properties need to be added to treatment definitions, then these strategies may be updated to accomplish this.

Optional DDR extension points

For extension points listed in this section, the Decision Data rule is empty, and you can configure it as needed to suit your requirements, or leave it blank.

For more information about these extension points, see Additional NBA Strategy framework components.

Extension point Decision Data rules	Parent strategies	Purpose	Business use case
ModelPropensityThresholds	<Triggering Strategy>-> NBA Strategy Framework -> Action Level Propensity -> PredictActionPropensity <Triggering Strategy>-> NBA Strategy Framework -> TreatmentsChannels -> TreatmentLevelPropensity-> PredictTreatmentPropensity	Used to set lower limits on the propensity generated by the adaptive models at the Direction, Channel, Issue, Group, Action, or Treatment levels. **Note:** The Treatment level is only referenced if the propensity calculation is set to **Best treatment**.	This DDR only needs to be used if there is a requirement to set minimum propensity values above which action propensities need to be in order to be emitted. Otherwise it may be left empty.
ActionLimits	<Triggering Strategy>-> NBA Strategy Framework -> ChannelProcessing -> OutboundChannelProcessing-> ProcessOutboundChannels-> TopOfferOrBundleForOutbound <Triggering Strategy>-> NBA Strategy Framework -> ChannelProcessing -> InboundChannelProcessing-> AssistedInboundChannel OtherInboundChannelProcessWebChannelProcessMobileChannel	Used to set limits on the number of actions that can be emitted for each action context, limits can be set at the Direction and Channel levels. These limits are processed after arbitration, and before outbound contact policies are applied. For inbound channels, they are only processed for the Web and Mobile channels if **Channel treatment processing** is disabled.	This DDR only needs to be utilized if there's a requirement to set limits on the number of actions that may be emitted per Direction, Channel and Context other than the defaults. Otherwise it may be left empty.
ArbitrationInfluencers	<Triggering Strategy>-> NBA Strategy Framework-> Arbitration-> ApplyInfluencers	Used to define Arbitration Influencer Contact Policies that use the insights gained from the outcome of presenting an action to influence the arbitration of the same action - or one or more related actions - at some future interaction.	This DDR only needs to be utilized if there's a requirement to upweight some actions in future interactions based on a prior response to an action. For example, during an interaction with a call center agent, if a customer expresses an interest in an offer, but needs to drop off the call due to other commitments, the outcome could be recorded as Maybe-Later. On the next interaction with the same customer it may be prudent to boost the chances of that same action - or some related actions – being selected as the next best action, on the assumption that the customer

Extension point Decision Data rules	Parent strategies	Purpose	Business use case
			may still be interested and have a higher likelihood to accept the follow-up action based on their initial Maybe-Later response.
OutputBundlingSettings	<Triggering Strategy>-> FinalActionLimitsAndBundling-> BundlingSettingsYieldActions	Used to set the bundling output options (BundleOutputType, SinglePrimary and pyDeliverOffline) by Direction and Channel, which are mainly used for outbound processing. These settings are applied after actions for all contexts have been processed by the Next-Best-Action Designer framework and prior to evaluating the final action limits.	This DDR only needs to be utilized if there is a requirement to change the default bundling and output settings. Otherwise it may be left empty.
FinalActionLimits	<Triggering Strategy>-> FinalActionLimitsAndBundling-> ApplyFinalActionLimits	Used to set limits on the number of actions that can be emitted for each contact or for the interaction as a whole. Limits can be set at the Direction, Channel and Action Context levels, and the limits for action contexts can be combined.	This DDR only needs to be utilized if there is a requirement to set limits on the number of actions that may be emitted per Direction, Channel, Context and Contact other than the defaults. Otherwise it may be left empty.

Modifiable DDR extension points

Extension points listed in this section are fully functional out of the box, but you can still modify the logic if it is critically important to your business needs.

Extension point Decision Data rules	Parent strategies	Purpose	Business use case
<Channel>Treatments	<Channel>ChannelStrategy	Additional treatment properties may be added to these Decision Data rules. **Note:** You must update the parent strategies to map the new data in the associated Data Join.	See the use case for <Channel>ChannelStrategy in List of extension points in strategies.

Testing the Next-Best-Action strategy framework

You can verify that your Next-Best-Action Designer configuration is giving the desired results by defining a data transform that represents a test customer persona, and running a test for this persona directly from the strategy canvas.

Pega Customer Decision Hub

Strategy framework

The strategy framework consists of the following components:

- Import actions, to import the actions using Proposition Data shapes
- Engagement policies, which filter out actions based on eligibility, applicability and suitability criteria using Proposition Filters
- NBA strategy framework, which applies constraints, calculates the action and treatment propensity using AI, performs arbitration, and finally applies channel-specific rules to select the top actions based on the channel requirements.

These parts of the strategy framework are tested in slightly different ways. These approaches will be discussed in detail in the sections that follow.

- Testing trigger strategies
- Using the trigger strategy to test an engagement policy
- Using the trigger strategy to test the NBAStrategyFramework component
- Testing the lower strategies

Testing trigger strategies

Trigger strategies are referenced by the dataflow which is invoked by the channel services. To test a trigger strategy, use the data transform (persona) that was automatically generated for you based on your Context Dictionary and run the trigger strategy to identify where the actions are dropping out or where the strategy results are unexpected.

Pega Customer Decision Hub
Before you begin: Update the persona data transform to include customer attributes that satisfy any engagement policies that you defined for the strategy. The persona transform includes the following sections:

ContainerPayload
: Inbound channels use the real-time container service to execute the Next-Best-Action strategy framework. This data structure allows you to define and test results for inbound channels, such as the ContainerName, Direction and Channel, as well as any contextual data. To test Inbound scenarios, you need to set the Direction to Inbound, and you must also set Channel to an active channel that you wish to test. For Outbound scenarios you can set the Direction to be Outbound or blank; channel will be ignored.

EventPayload
: Real-time events execute the Next-Best-Action strategy framework to determine if there is an appropriate next best action to take based on the event. This data structure allows you to define and test results for an event such as EventName and EventType.

Customer, Account, Subscriber, etc.
: These sections represent all of the context dictionary related data. Add any property that you want to test against your engagement policy criteria.**Note:** The name of these data structures will vary depending on your Context Dictionary definition.

JourneyStages
: This section should be within the primary context entity section (for example, Customer or Subscriber) and contains data about any active Journey Stages if the Journeys feature is being used.

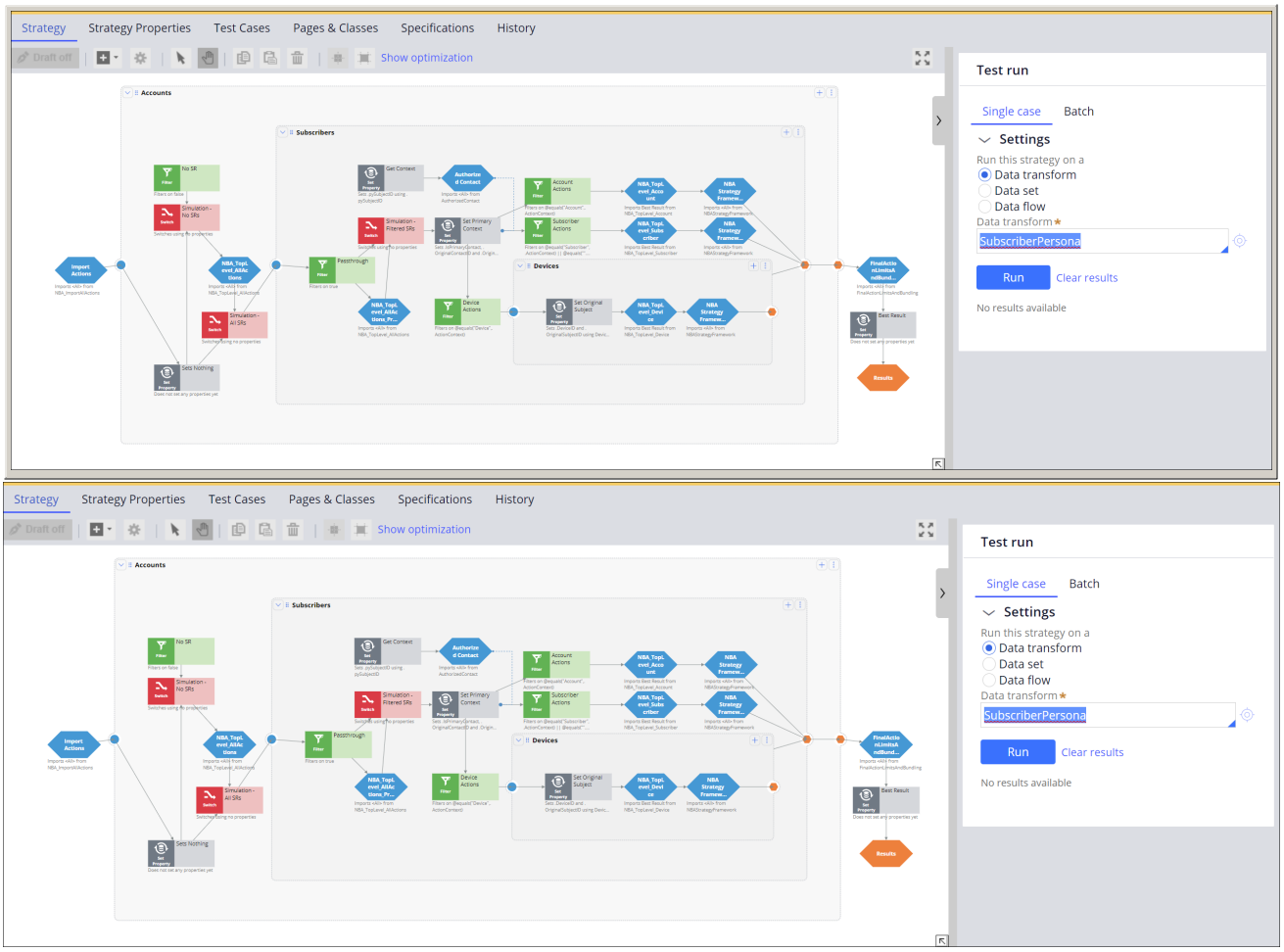
ActionInsights
: This section should be within the primary context entity section (for example, Customer or Subscriber) and contains data related to any Contact Policy Suppressions or Arbitration Influencers.

Other sections may be present depending on your context dictionary configuration, and sections may be added as required.

Save the persona data transform into your implementation layer to create as many personas as you need to test your strategies.

1. In the Pega Customer Decision Hub portal, click **Intelligence Strategies** .
2. Open the *Trigger_NBA_TopLevel* strategy.
3. Open the **Test Run** panel by clicking the slider button on the upper right side of the strategy canvas.
4. In the **Settings** section, select **Data transform**, and then select a persona transform, as in the following figure:

Selecting a customer persona for strategy testing



Note: The example shows a configuration with three context levels. If your Context Dictionary has only one or two context levels, your strategy will look simpler than this.**Tip:** When the trigger strategy is first generated, the strategy shapes may be jumbled on the canvas. The simplest way to re-organize the shapes is to collapse each of the context shapes – starting with the lowest level context – by clicking the collapse button at the top left of the shape. You can then re-arrange the top-level shapes, and then expand each context shape in turn and re-arrange the shapes within it.

5. Click the **Open** icon.
6. On the **Definition** tab of the data transform, specify the characteristics of the persona.
   The persona characteristics are defined as properties of the customer, their journey, and other related data.
   For example: If you have a specific test customer contact that you want to use as a test persona, find the **Set Action** for the target **.CustID** and enter the value into the **Source** box. You can add new properties by right-clicking on a row (not in the Target or Source boxes) and selecting **Add child**.
7. Save the data transform (or click **Save As** to create a new version) and switch back to your trigger strategy.
8. Click **Run**, and then select the **Results** component to view the strategy results.

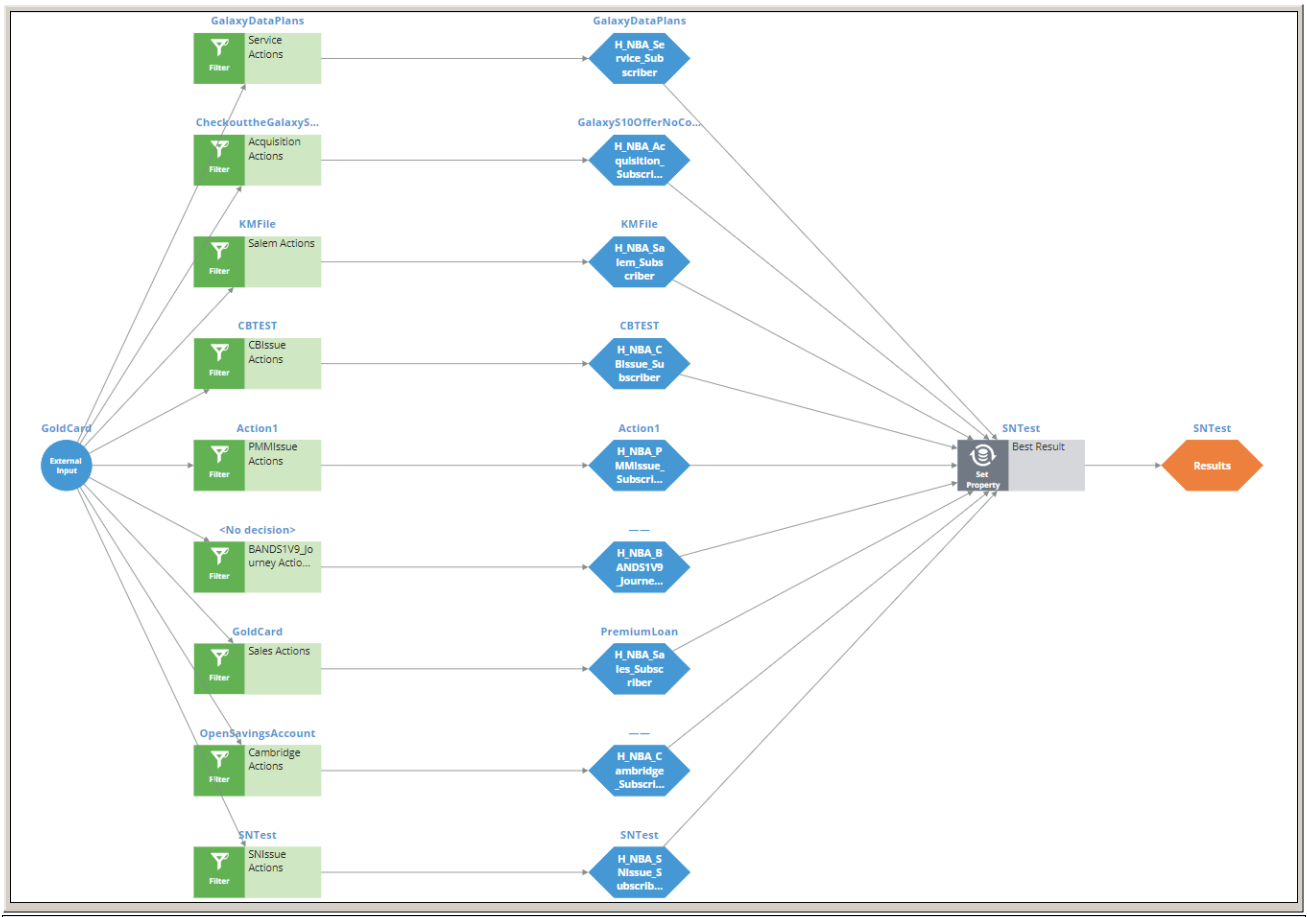
What to do next: To find the strategy component that is not performing as expected, you can click through each shape on the strategy canvas and identify if and where actions are dropping out or where the strategy results are unexpected.

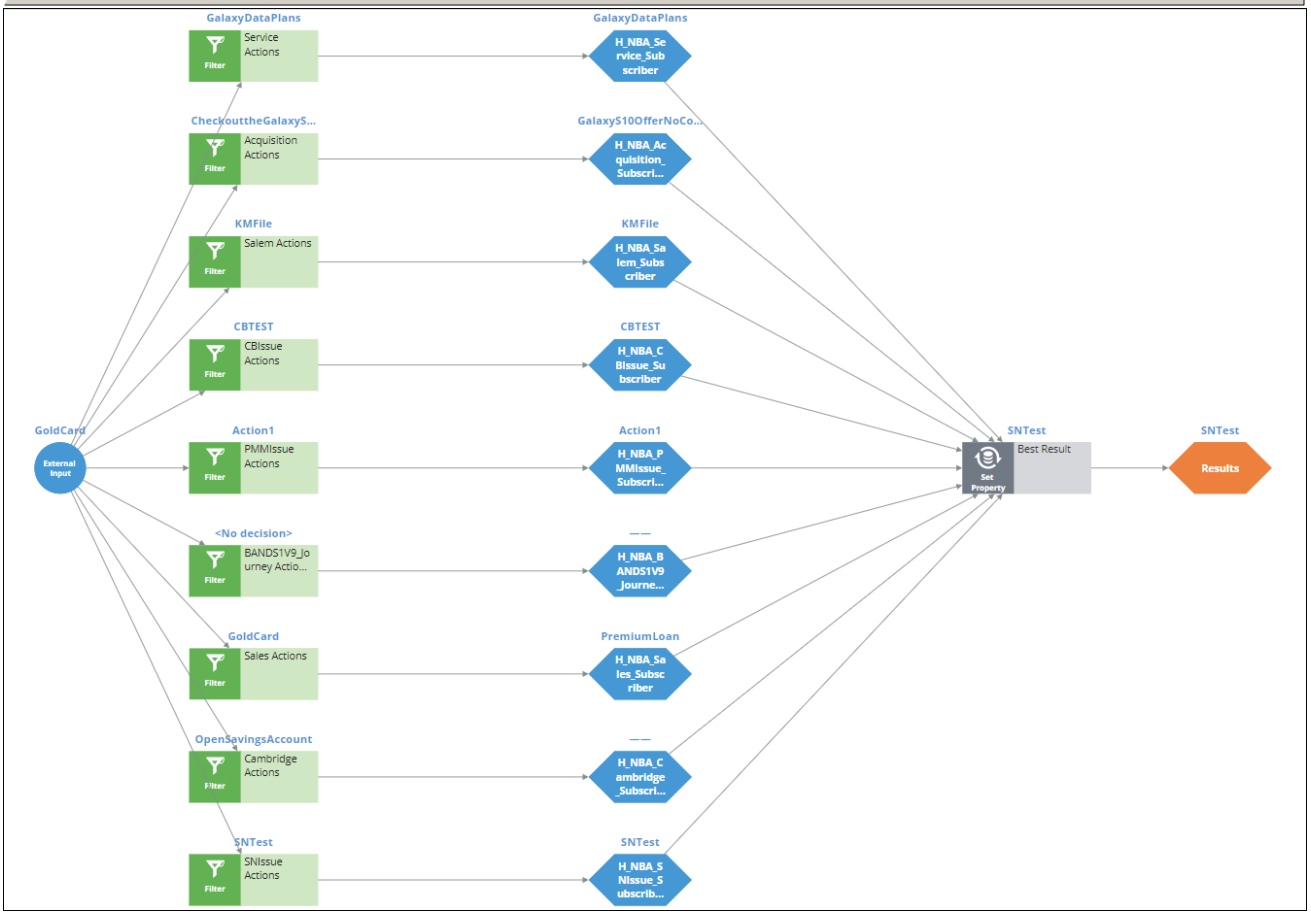
Using the trigger strategy to test an engagement policy

If you see that strategy results are getting dropped from one of more of the engagement policies in the trigger strategy, this is often expected behavior since the policy conditions may be legitimately excluding actions that do not meet the criteria. However, if the results are not as expected, the next step is to test the engagement policy strategy to find the issues and groups where the results are unexpectedly dropping out.

Pega Customer Decision Hub
You can use the combination of the Persona data transform and the trigger strategy to unit test an engagement policy.

1. Right-click on the Engagement Policy strategy that you wish to investigate, and then click **Open strategy**.
2. Open the **Test Run** panel by clicking the slider button on the upper right side of the strategy canvas.
3. In the **Settings** section, select **Data transform**, and then select a persona transform already created for the trigger strategy test.
4. In the **For external inputs use strategy** field, enter the trigger strategy name, for example, Trigger_NBA_TopLevel.
5. Select **Specify a single component within the strategy**, and then enter the name of the component that is immediately prior to the engagement policy you wish to test.
   In the figure below, the Subscriber Actions filter component is chosen to test the *NBA_TopLevel_Subscriber* policy.

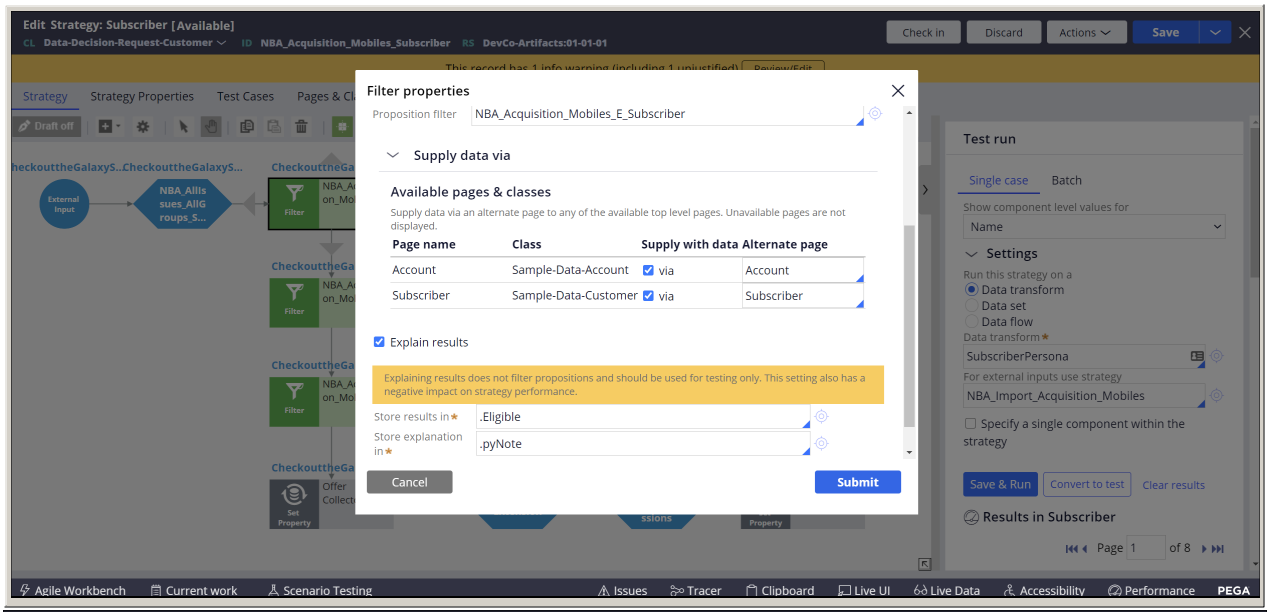


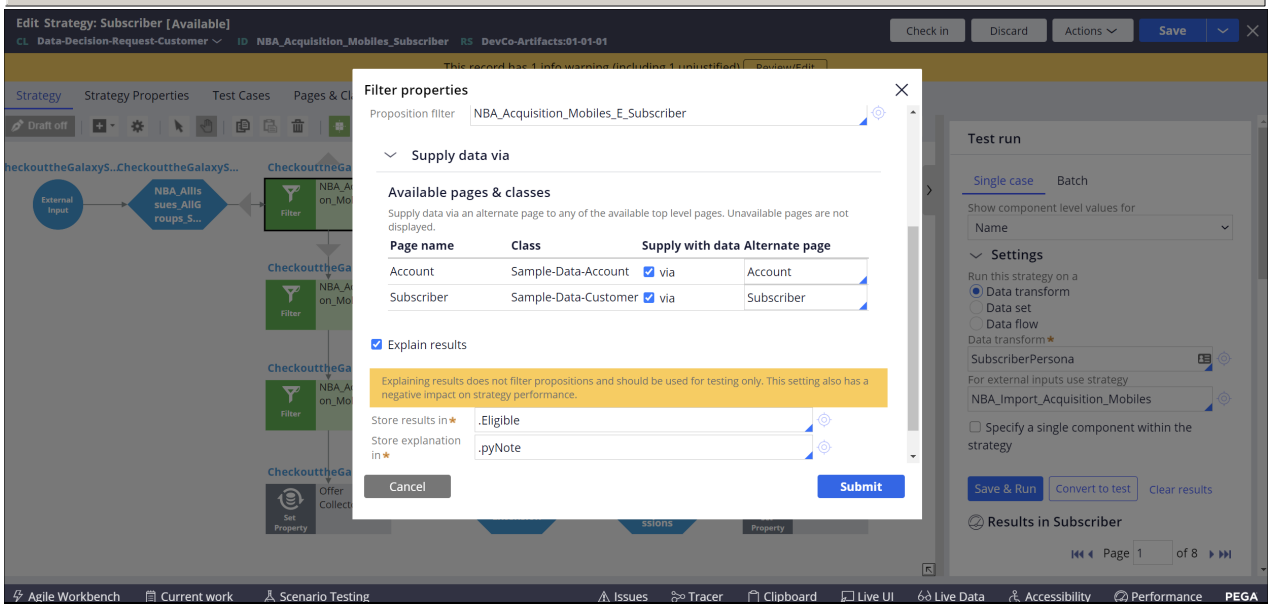


6. Click on each sub-strategy to see where actions are unexpectedly dropping out or where the strategy results are unexpected.
7. Drill down by right-clicking and selecting **Open strategy** and repeat the process until you find the component that is causing the problem.
   Note: Each time you drill down you may use the same data transform and trigger strategy component for the test, but bear in mind that each time you drill down you will be losing the filter from the previous level, and so some unwanted actions from other issues or groups may also be processed.
8. To narrow down the actions being processed, you may directly select the action import strategy for a specific Issue and Group as the external inputs strategy.
   The action import strategy will be named *NBA_Import_<Issue>_<Group>* where *<Issue>* and *<Group>* represent the issue and group that you require.**Note:** Once you do

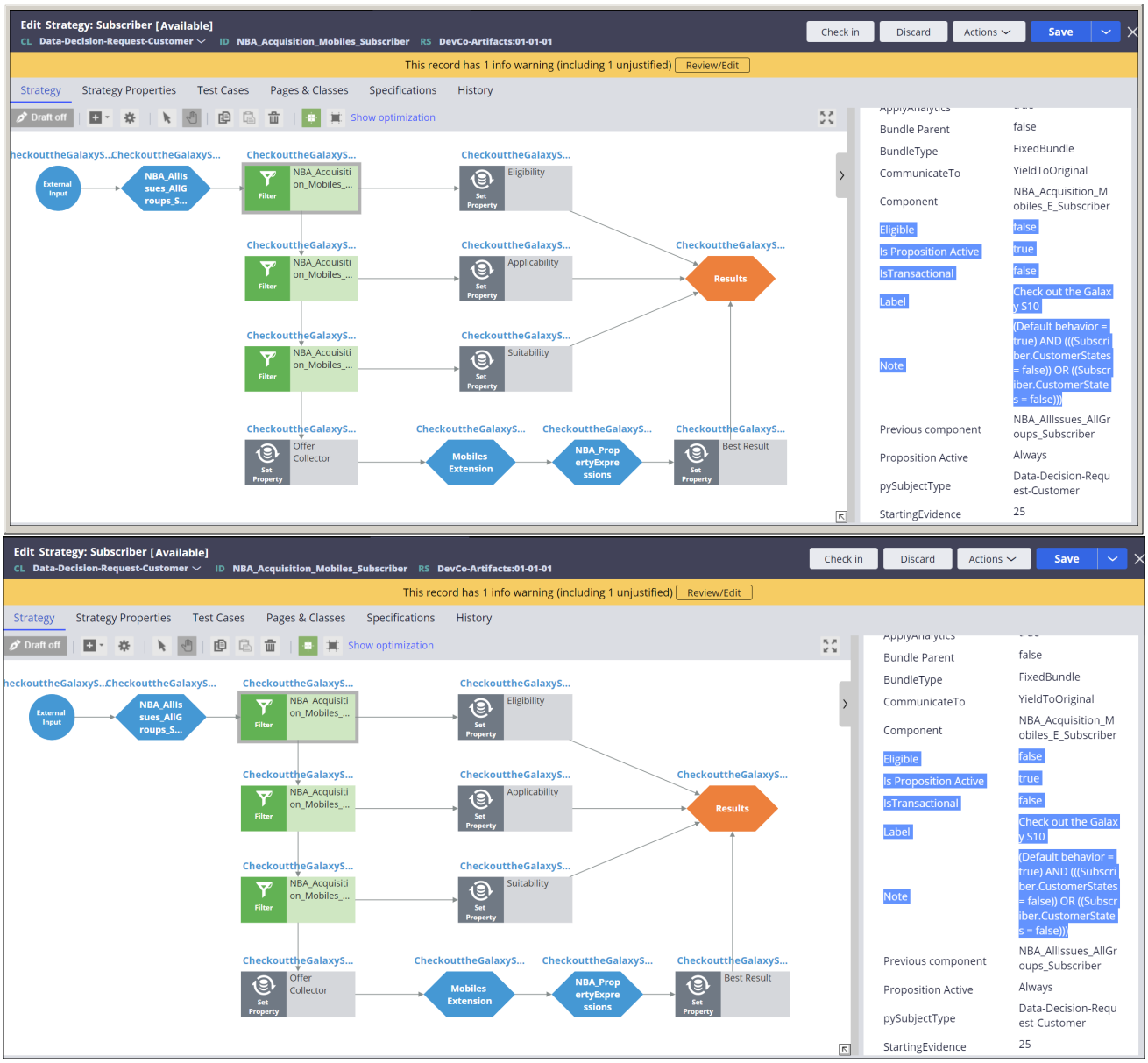
this, you will lose the context filter, and so will be processing actions for all contexts, so bear this in mind when viewing the test results.

9. If the component causing the issue is a Proposition Filter, you can check out the strategy and select the **Explain results** option to examine the results for each action.
   a. Double-click on the Proposition Filter to open it.
   b. Select **Explain results** in the lower left of the **Properties** panel.
   c. Store the results in the **.Eligible** property, and store explanation in the **.pyNote** property, as shown in the figure below.





10. Click **Save and run**. **Result:** The **Test Run** panel shows the results for **Eligible** and **Note**, as shown in the following figure:



Note: The value of the **Eligible** property is set to true if the action met all of the filter conditions, otherwise it is set to false. The **Note** property contains results of all conditions evaluated.

11. After you finish testing, discard your changes to the strategy.

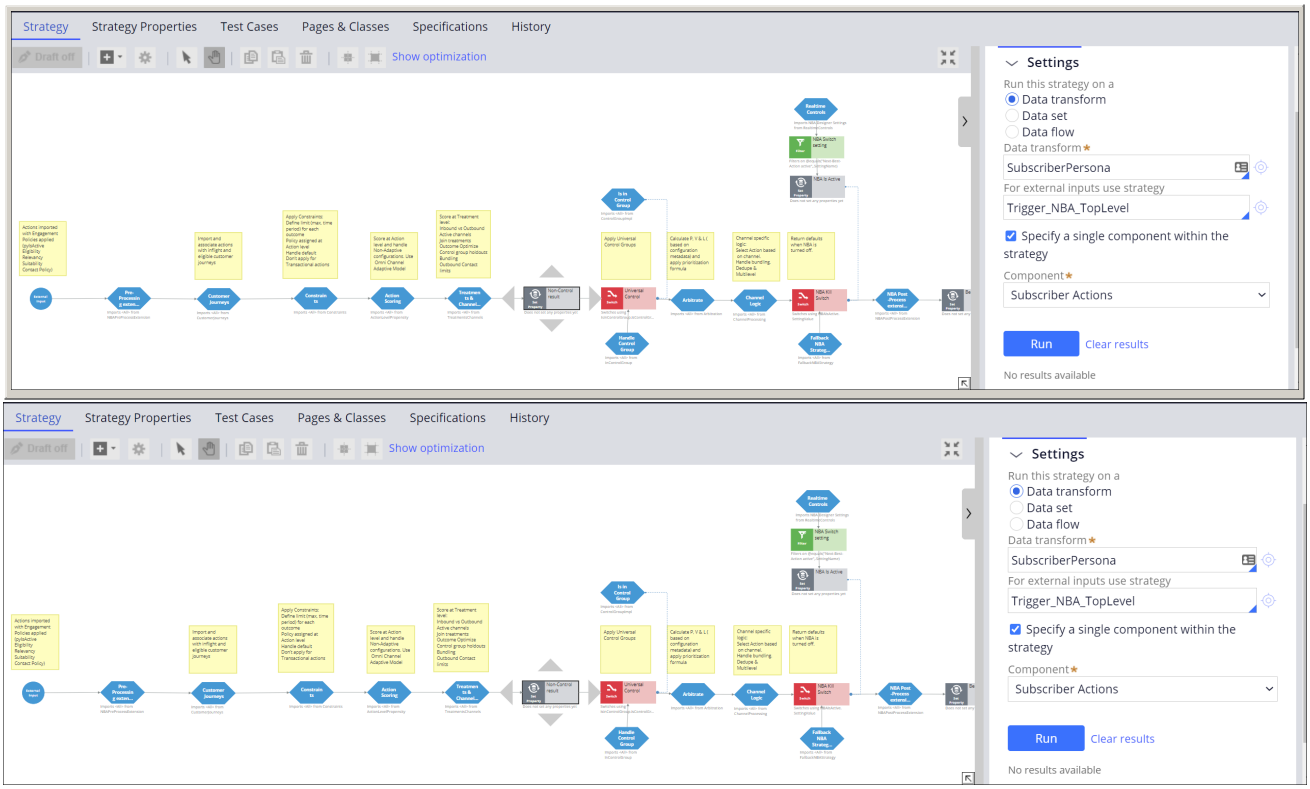
Using the trigger strategy to test the NBAStrategyFramework component

If you see unexpected strategy results from the NBA Strategy Framework component in the trigger strategy, the next step is to test the NBA Strategy Framework strategy to find the high level component where the results are dropping out or are unexpected.

Pega Customer Decision Hub
You can use the combination of the persona Data Transform and the trigger strategy to unit test the strategy framework.

1. In the Pega Customer Decision Hub portal, click **Intelligence Strategies** , and then open the *NBAStrategyFramework* strategy.
2. **Optional:** To test a specific sub-strategy component, right-click on the component, and then click **Open strategy**.
3. Open the **Test Run** panel by clicking the slider button on the upper right side of the strategy canvas.
4. In the **Settings** section, select **Data transform**, and then select the persona transform that you created for the trigger strategy test.
5. In the **For external inputs use strategy** field, enter the trigger strategy name, for example, Trigger_NBA_TopLevel.
6. Select **Specify a single component within the strategy** , and then choose the action filter for the context that you want to test.
   Note: Only a single context is processed at a time by the NBA Strategy framework. When testing, it is important to also process a single context only at a time. In the example below, the **Subscriber Action** filter component is chosen to only process subscriber actions. To test for the Account context, you would need to choose **Account Actions** instead.

   Selecting an action filter component



7. Click **Run**, and then select the **Results** component to view the strategy results.

What to do next: To find the strategy component that is not performing as expected, you can click through each shape on the strategy canvas and identify if and where actions are dropping out or where the strategy results are unexpected. For more information, see Understanding the NBAStrategyFramework component test results.

- Understanding the NBAStrategyFramework component test results

Understanding the NBAStrategyFramework component test results

After you test the NBA Strategy Framework strategy, use the following troubleshooting tips to help identify the component where actions are dropping out or where the strategy results are unexpected.

Pega Customer Decision Hub
You should expect to see the number of selected actions and treatments change as you progress through the components in the strategy from left to right. The following section is a summary of what to expect for each top-level component.

Pre-processing extension point

Unless this strategy has been customized, you should not expect any changes here.

Customer Journeys

This strategy assigns Journey Stage data to actions. If there are journey stages configured with actions assigned to them, the *ActionRequriesJourney* property is set to `true` on the action form, and there is currently no eligible journey for the action, then the action will be dropped, so it is possible to see a reduction in actions at this point.

Constraints

This strategy applies Contact Policies based on prior outcomes, so you could expect to see actions drop out if they are referenced as Suppressions in the *ActionInsights* page list in the Data Transform.

For example, actions that have treatments for multiple enabled channels will cause an increase, but actions that have a single treatment for a channel that has been disabled will cause a decrease, although this latter case will be rare.

If any transactional actions are defined, be aware of how they are treated differently.

The second part of the strategy executes adaptive models for each action and channel combination, and this should not in itself cause any changes.

If the Next-Best-Action Designer **Channel treatment processing** setting is set to `false`, there are other potential changes:

- For outbound interactions on channels other than Paid, where Outbound Model Maturity is enabled, and where a model is immature (that is, has less than 200 positive responses by default), a significant percentage of contacts (98% by default) will not receive the action. Since the percentage test is evaluated based on a hash of the customer ID and the action ID, this is virtually impossible to control in test scenarios, so to overcome this, consider switching Outbound Model Maturity off during testing, or set the direction to be Inbound and choose a suitable channel.
- Ensure you are familiar with any Model Propensity Thresholds DDR settings, since these may also cause actions to be dropped.

Treatments and channels

If the Next-Best-Action Designer **Channel treatment processing** setting is set to `false`, this whole strategy is bypassed and does not cause any changes. Otherwise, the following processing occurs.

The first part is where actions are joined to their treatments. Expect to see no change if each action has a single treatment per channel, or an increase if there may be multiple treatments per channel.

The second part of the strategy executes adaptive models for each action / treatment combination, and this should not in itself cause any changes.

However, as with Action Scoring, there are other potential changes:

- For outbound interactions on channels other than Paid, where Outbound Model Maturity is enabled, and where a model is immature (that is, has less than 200 positive responses by default), a significant percentage of contacts (98% by default) will not receive the action. Since the percentage test is evaluated based on a hash of the customer ID and the action ID, this is virtually impossible to control in test scenarios, so to overcome this, consider switching Outbound Model Maturity off during testing, or

set the direction to be Inbound and choose a suitable channel.
- Ensure you are familiar with any Model Propensity Thresholds DDR settings, since these may also cause actions to be dropped.

Universal Control

You should only expect to see a change if the *Is In Control Group* and *Handle Control Group* strategies have been customized, which is not common.

Arbitrate

This strategy calculates the priority for each action and treatment, and prioritizes them. There should be no changes to the number of selected actions and treatments.

Channel Logic

This strategy applies channel specific processing, including applying actions limits which may reduce the number of actions and treatments. However, this is normally only done for outbound interactions where the default is one action or treatment per contact per context. Ensure that you are familiar with any Action Limits DDR settings, since these may cause actions to be dropped.

NBA Kill Switch

If the Next-Best-Action Designer **Next-Best-Action active** setting is set to true, there will be no change. In the unlikely event that the setting is set to false, all actions and treatments will be replaced by the output from the Fallback NBA Strategy.

NBA Post-process extension

Unless this strategy has been customized, you should not expect any changes here.

Testing the lower strategies

Once you identify the point in the framework where you are seeing unexpected results, you may want to test that sub-strategy and perhaps some related strategies.

Pega Customer Decision Hub
For this, you will need to create a more sophisticated strategy for use as the external inputs strategy to ensure that all of the appropriate context actions are available, and that your strategy results contain the output from all of the required steps prior to the sub-strategy that you wish to test.

The simplest way to ensure that your external input strategy has executed each of the steps prior to the one you want to test is to actually use the NBA Strategy Framework itself as the input strategy, and specify the single component to use within the strategy as the one immediately prior to the one you wish to test.

The problem with this is that the NBA Strategy Framework has an External Input component, and there is no way to specify an external input strategy within an external input strategy. To overcome this, you will need to make a copy of the NBA Strategy Framework strategy, disable external input, and then provide a set of actions to use as input for the tests. For more information, see the following topics:

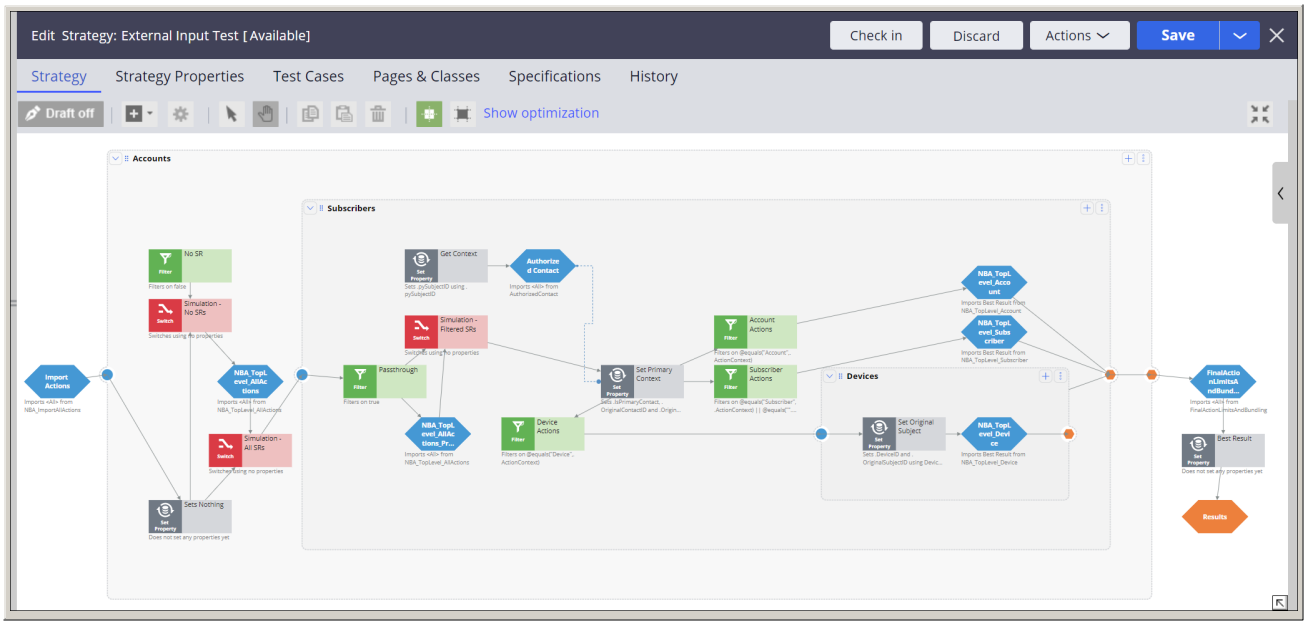
- Creating a test strategy for selecting actions
- Creating an External Input strategy to test the framework
- Testing the Action Scoring component
- Testing a strategy within a lower-level strategy
- Testing the Final Action Limits And Bundling strategy
- Testing the strategies within Final Action Limits And Bundling

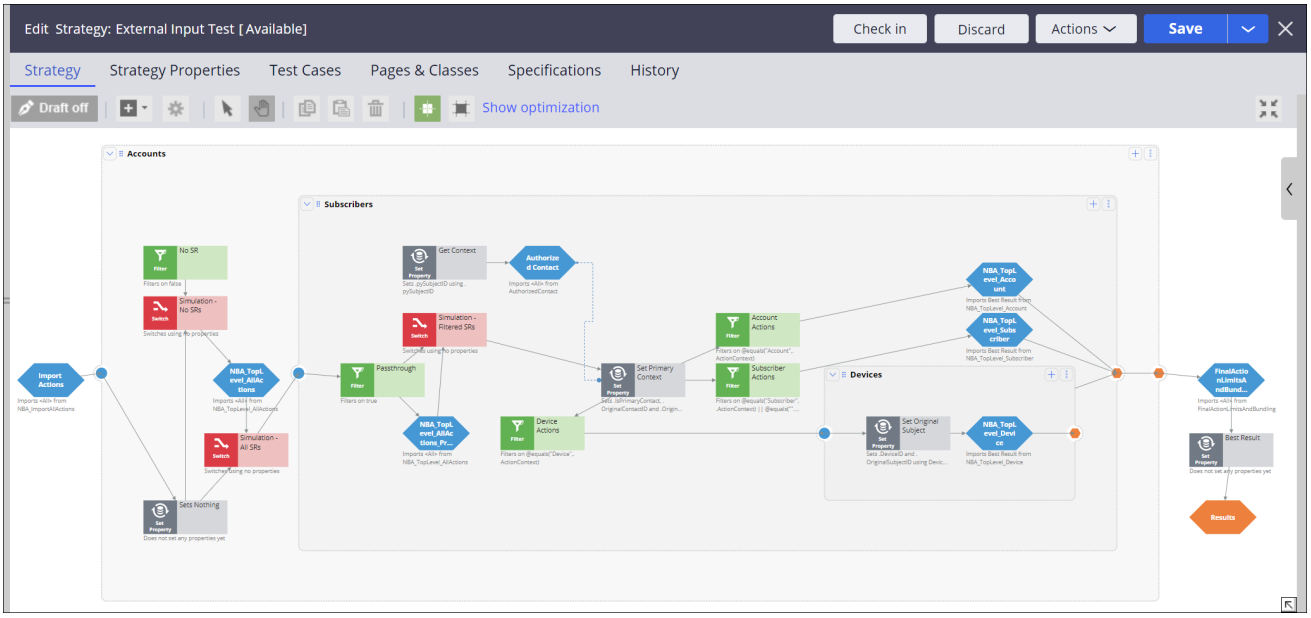
Creating a test strategy for selecting actions

Testing the lower strategies in the NBA Strategy Framework requires that you create a strategy that includes the set of actions against which to test.

Pega Customer Decision Hub
Since the trigger strategy is the source of action data for the framework, it is best to create a modified version of the trigger to provide the actions needed for the NBA Strategy Framework tests.

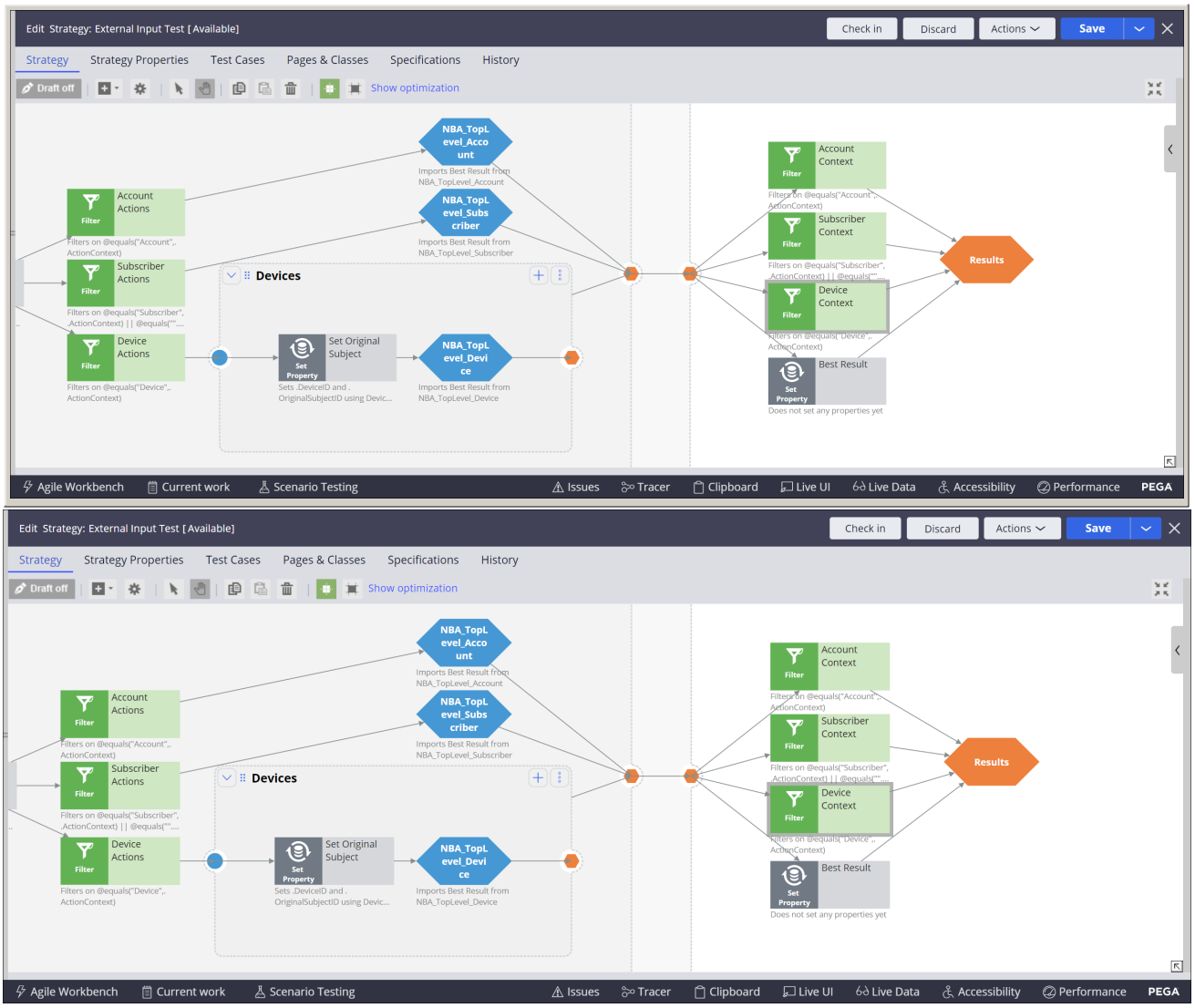
1. In the Pega Customer Decision Hub portal, click **Intelligence Strategies** .
2. Open the *Trigger_NBA_TopLevel* strategy.
3. Click **Save As** and change the label to be the name of the test strategy, for example, External Input Test.
4. Ensure that **[No branch]** is selected in the **Development branch** field.
5. Click **Create and open**.
6. Since you will be using this strategy to test the NBA Strategy Framework, remove any references to the framework from the strategy.
   Delete each NBA Strategy Framework sub-strategy shape and connect the previous component (the engagement policy sub-strategy shape) to the context shape exit point. The result should resemble the following figure:





Note: If your Context Dictionary has only one or two context levels, your strategy will look simpler than the above figure.

7. Since the framework is executed separately for each context, for a multi-level context dictionary, create a separate exit point for each context.
   a. To the right of the outermost context shape, create a separate Filter for each context using the Filter conditions from their respective filters within the context shapes.
      Note: Name the filters appropriately, for example, Account Context, Subscriber Context, and so on.
   b. Connect the exit point of the outermost context shape to each filter, and then connect each filter to the Results shape.
8. Remove the **Final Action Limits & Bundling** strategy, and then connect the exit point of the outermost context shape directly to Best Result.
   The result should resemble the following figure:



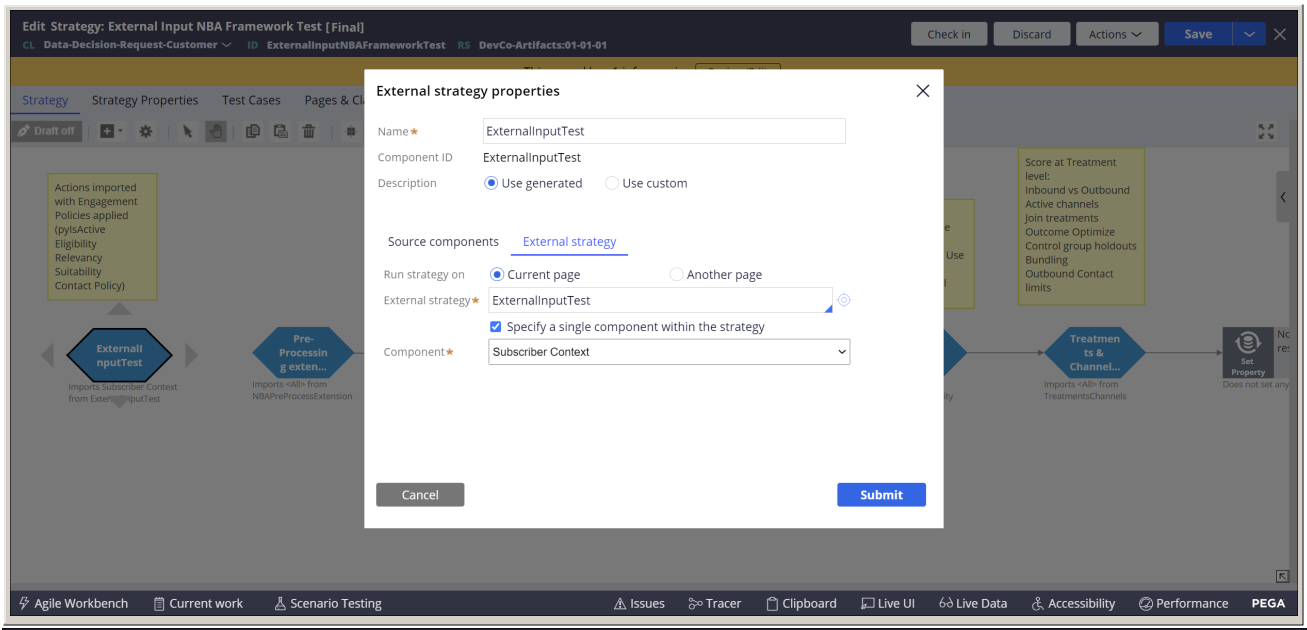
Result: This strategy can now be used provide actions for each individual context, and for all contexts if required.

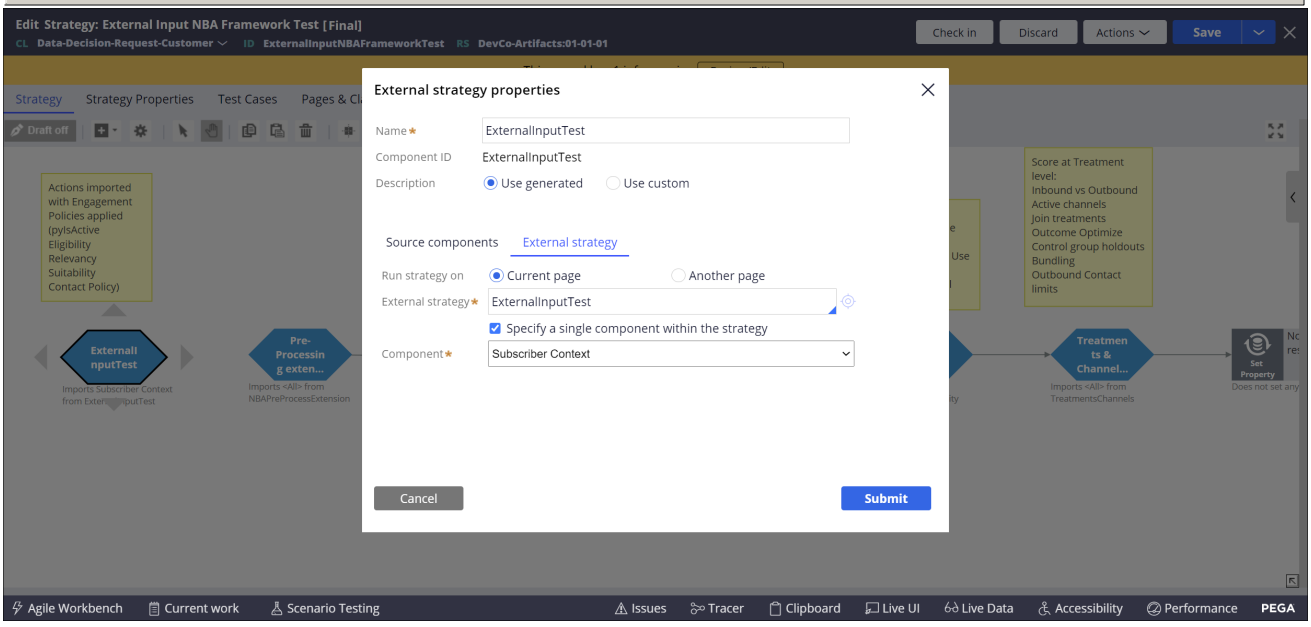
Creating an External Input strategy to test the framework

Once the actions source is available, you can use this to replace the External Input in the NBA Strategy Framework.

Pega Customer Decision Hub

1. In the Pega Customer Decision Hub portal, click **Intelligence Strategies** .
2. Open the *NBA Strategy Framework* strategy.
3. Click **Save As** and change the label to be the name of a new test strategy, for example, External Input Test.
4. Ensure that **[No branch]** is selected in the **Development branch** field.
5. Click **Create and open**.
6. Right-click on an empty area of the canvas, and then click **Disable external input**.
7. In the **Input Disable Warning** message, click **Submit**.
8. Right-click to the left of the leftmost component in the strategy.
9. Select **Sub-strategy External** .
10. Double click on the new component.
11. In the **External strategy** field, enter ExternalInputTest.
12. Select **Specify a single component within the strategy**.
    Note: You may need to do this twice, since the first check is used to verify the strategy name.
13. From the **Component** list, select the action context that you want to use for testing.
    Note: Because the framework is only ever executed for a single context at a time, it is highly recommended to use a single context when testing the framework. Select the context where you are seeing issues that you need to test. If required, you can change your selection later.





14. Connect the new **External Input Test** shape to the **Pre-processing extension point** shape.
15. Save and check in your changes.

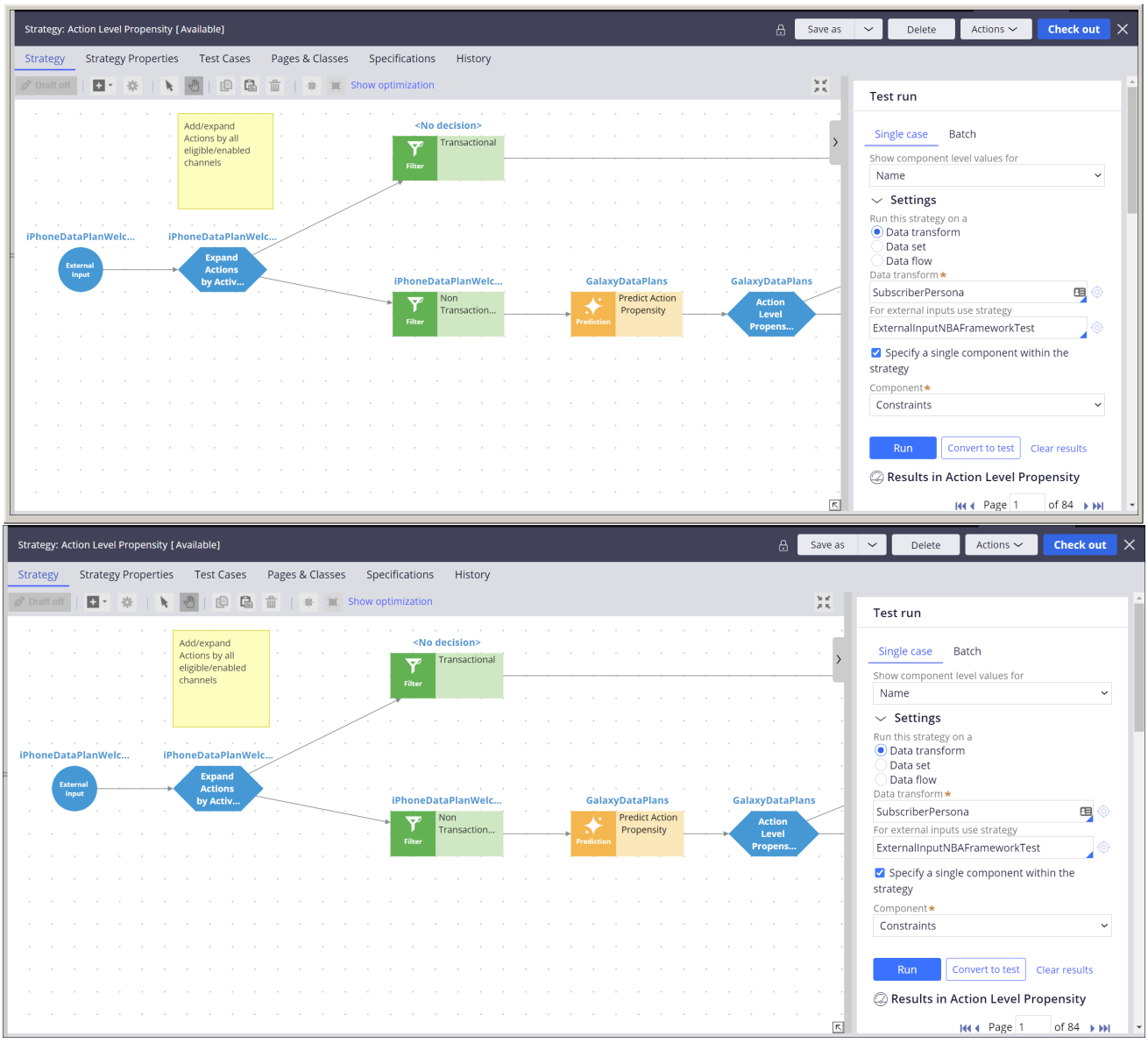
Result: You have created a test strategy that can be used to test sub-strategies within the NBA Strategy Framework using a specific action context. If you need to test using a different context, you can either use the **Save as** option to create an alternative version, or you can check out the strategy, change the component referenced in the call to the External Input Test sub-strategy, and then save the changes for use during your tests.**Tip:** It is recommended that you use the second method while you may still be making changes to the strategy. When the strategy is stable and you have no further changes to make, you can safely make one or more copies without having to keep updating the copies.

Testing the Action Scoring component

This section provides an example of how to test a sub-strategy with the NBA Strategy Framework. A similar technique can be used for strategies other than Action Scoring.

Pega Customer Decision Hub

1. In the Pega Customer Decision Hub portal, click **Intelligence Strategies** .
2. Open the *NBA Strategy Framework* strategy.
3. From the NBA Strategy Framework strategy, right-click the Action Scoring component, and then click **Open strategy** to open the Action Level Propensity strategy.
   Tip: The Action Level Propensity strategy will test expanding actions with active and eligible channels, in addition to running the Predict Action Propensity prediction to determine the action level propensity.
4. Open the **Test Run** panel by clicking the slider button on the upper right side of the strategy canvas.
5. In the **Settings** section, select **Data transform**, and then select a persona transform.
   Note: To make any changes to the persona for the test, follow the steps used when setting up the person for the trigger strategy tests.
6. In the **For external inputs use strategy** field, enter ExternalInputNBAFrameworkTest.
7. Select **Specify a single component within the strategy** .
8. Select the component that immediately precedes the component you are testing. **For example:** For the Action Level Propensity strategy, since you are testing the Action Scoring component, select the Constraints component, as in the following figure:



9. Click **Run**.
10. Click on individual components within the strategy to check where actions are dropping out or where the strategy results are unexpected.

Testing a strategy within a lower-level strategy

To test a strategy that is called from within a lower level NBA Strategy Framework strategy, some minor extensions to the External Input NBA Framework Test strategy are required.

Pega Customer Decision Hub
For example, if you want to test the components within the Action Level Propensity strategy, keep the following considerations in mind:

- Because the Expand Actions by Active Channels sub-strategy is the first component in the parent strategy, you can test it by using the same persona Data Transform and external input strategy as for the parent strategy.
- If you want to test the Predict Action Propensity strategy, you need to ensure that the Expand Actions by Active Channels component has executed first.

- Testing the Predict Action Propensity strategy

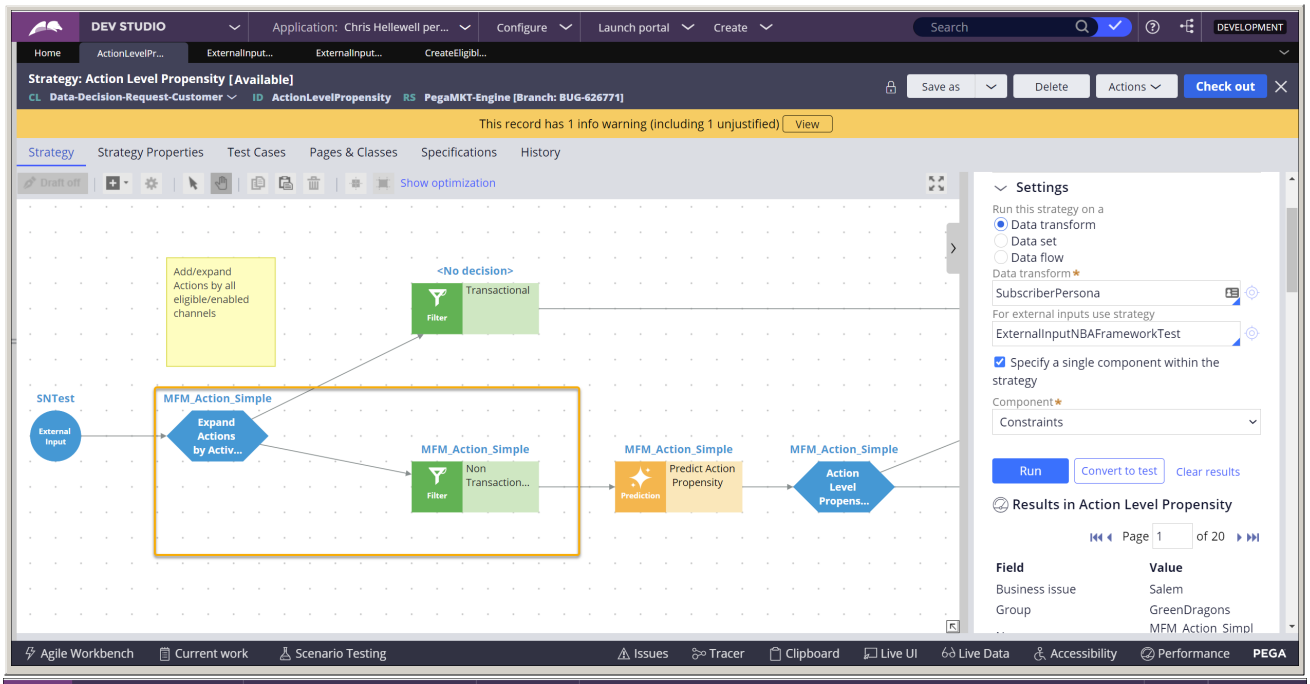
Testing the Predict Action Propensity strategy

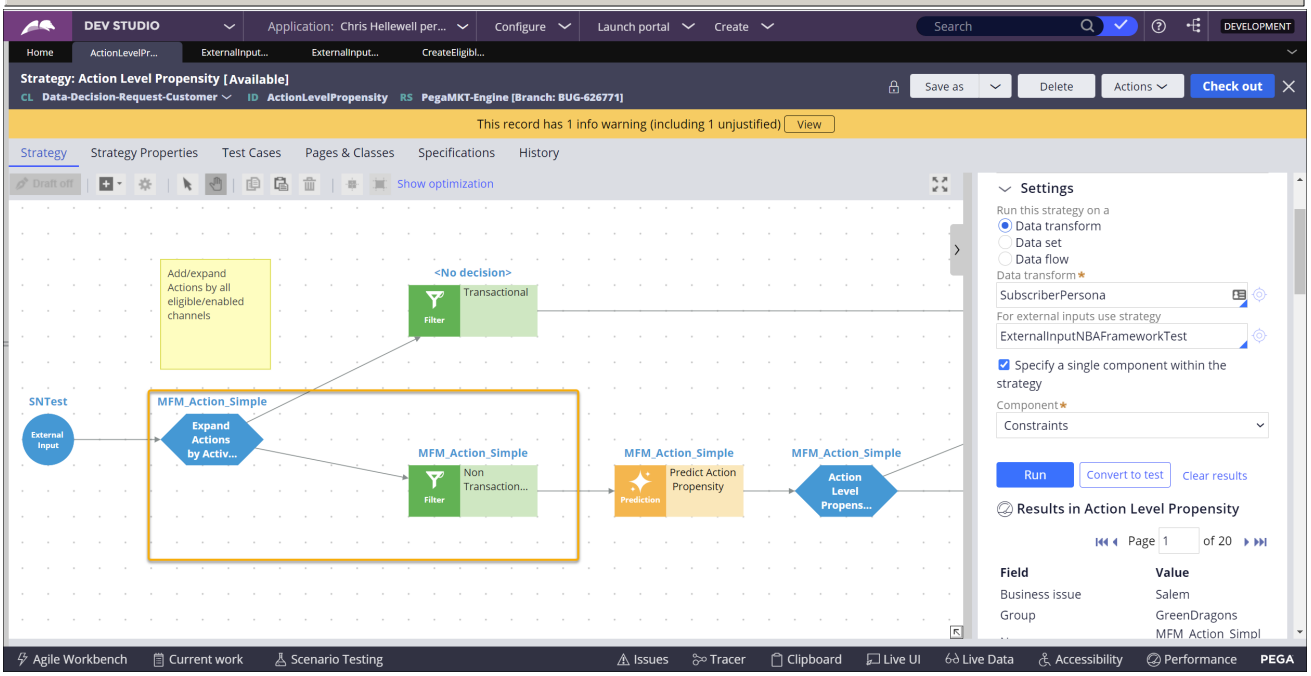
To test this sub-strategy, you need to ensure that the Create Eligible Channels strategy (the Expand Actions by Active Channels component) has executed first, and that the Non Transactional Filter has also been executed.

Pega Customer Decision Hub

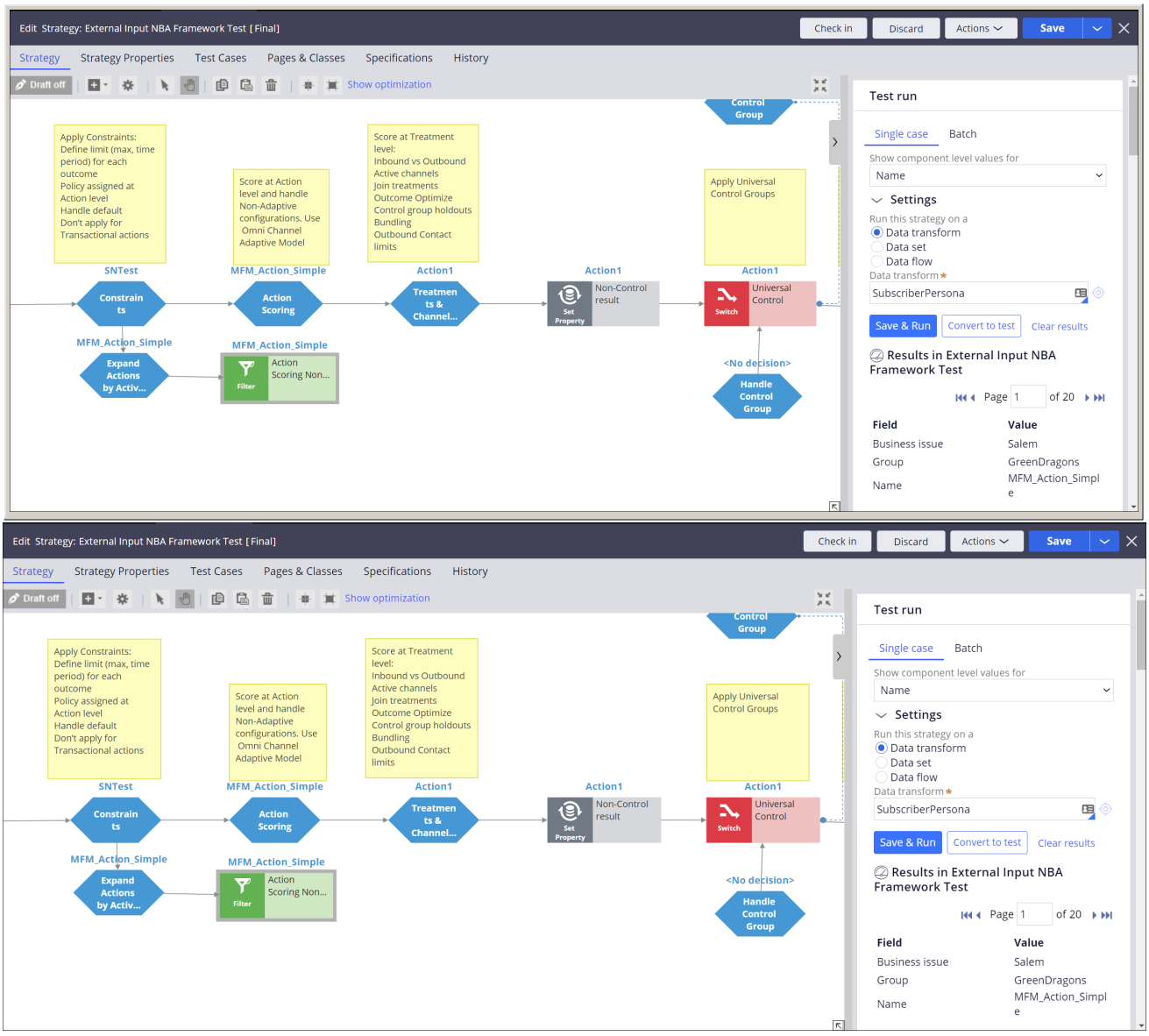
1. In the Pega Customer Decision Hub portal, click **Intelligence Strategies** .

2. Open the *External Input NBA Framework Test* strategy.
3. If the strategy is not already checked out, click **Check out**.
4. From the External Input NBA Framework Test strategy, right-click the Action Scoring component, and then click **Open strategy** to open the Action Level Propensity strategy.
5. Press the **Ctrl** button and click on the Expand Actions by Active Channels and Non Transactional Filter components to select them both.





6. Right-click on one of the selected components, and then click **Copy** to copy them both.
7. Return to the External Input NBA Framework Test strategy, and then paste the copied components in a spot directly below the Constraints sub-strategy component.
8. Join **Constraints** to **Expand Actions by Active Channels**.
9. Join **Expand Actions by Active Channels** to the **Non Transactional Filter**.
10. Double-click the **Non Transactional Filter** and change the name to `Action Scoring Non Transactional`.



This will help identify the component as being the Non Transactional filter from the Action Scoring component.**Note:** Do not join the output from this filter to any other shape on the canvas, so that its output will not affect any other parts of the strategy.

11. Click **Save** to save the strategy, or **Check in** if you are not making any further changes.
12. Verify that the test results for the Action Scoring Non Transactional filter in the *External Input NBA Framework Test* strategy are the same as those for the Non Transactional filter in the *Action Level Propensity* strategy.
13. Test the Predict Action Propensity component by doing the following steps:
    a. From the Action Level Propensity strategy, right-click the Predict Action Propensity component, and then click **Open strategy**. (right click, Open strategy).
    Note: Predict Action Propensity is actually a Prediction rather than a Strategy, but can be tested in the same was as a strategy.
    b. Open the **Test Run** panel by clicking the slider button on the upper right side of the strategy canvas.
    c. In the **Settings** section, select **Data transform**, and then select a persona transform.
    Note: To make any changes to the persona for the test, follow the steps used when setting up the person for the trigger strategy tests.
    d. In the **For external inputs use strategy** field, enter ExternalInputNBAFrameworkTest.
    e. Select **Specify a single component within the strategy**.
    f. Select the **Action Scoring Non Transactional** filter that you created in step 7.
    g. Click **Run**.
    h. On the strategy canvas click on the **External Inputs** shape and check that the test results are the same as those for the Action Scoring Non Transactional filter in the External Input NBA Framework Test, and for the Non Transactional filter in the Action Level Propensity performed in the tests in step 12.
    i. Click on individual components within the strategy to check where actions are dropping out or where the strategy results are unexpected.

What to do next: Use a similar technique to test other strategies within strategies.

Testing the Final Action Limits And Bundling strategy

This is the last component of the trigger strategy, invoked after the processing for each of the individual action contexts has been merged. It may be tested using a similar technique to that used for testing the NBA Strategy Framework.

Pega Customer Decision Hub

1. In the Pega Customer Decision Hub portal, click **Intelligence Strategies** .
2. Open the *Final Action Limits And Bundling* strategy.
3. Open the **Test Run** panel by clicking the slider button on the upper right side of the strategy canvas.
4. In the **Settings** section, select **Data transform**, and then select a persona transform.
   Note: To make any changes to the persona for the test, follow the steps used when setting up the person for the trigger strategy tests.
5. In the **For external inputs use strategy** field, enter the trigger strategy name, for example, Trigger_NBA_TopLevel.
6. Select **Specify a single component within the strategy**.
7. Select the component that immediately precedes the Final Action Limits And Bundling strategy.
8. Click **Run**.
9. Click on individual components within the strategy to check where actions are dropping out or where the strategy results are unexpected.

Testing the strategies within Final Action Limits And Bundling

Once you identify the point in the Final Action Limits And Bundling strategy where you are seeing unexpected results, test that sub-strategy and related strategies.

Pega Customer Decision Hub
In a similar manner to what was done for the NBA Strategy Framework, you will need to create a test strategy for use as the external inputs strategy to ensure that all of the required steps prior to the sub-strategy you wish to test have been executed.

The simplest way to do this is to use a modified version of the Final Action Limits And Bundling strategy itself as the input strategy, and specify the single component to use within the strategy as the one immediately prior to the one you wish to test. You will also need to replace External Input with a test version of the trigger strategy. For more information, see the following topics:

- Creating External Input from the Trigger strategy
- Creating an External Input strategy to test Final Limits
- Testing the Treatment Placements component

Creating External Input from the Trigger strategy

Testing the strategies within Final Action Limits And Bundling requires that you create version of the trigger strategy that excludes Final Action Limits And Bundling. This can be done by following the steps below.

Pega Customer Decision Hub

1. In the Pega Customer Decision Hub portal, click **Intelligence Strategies** .
2. Open the *Trigger_NBA_TopLevel* strategy.
3. Click **Save As** and change the label to be the name of a new test strategy, for example, External Input Final Limits.
4. Ensure that **[No branch]** is selected in the **Development branch** field.
5. Click **Create and open**.
6. Delete the Final Action Limits And Bundling sub-strategy shape and the Best Results shape, and then connect the outermost context shape directly to the Results shape, as in the following figure:



Result: This strategy can now be used to replace the External Inputs in a test version of the Final Action Limits And Bundling strategy.

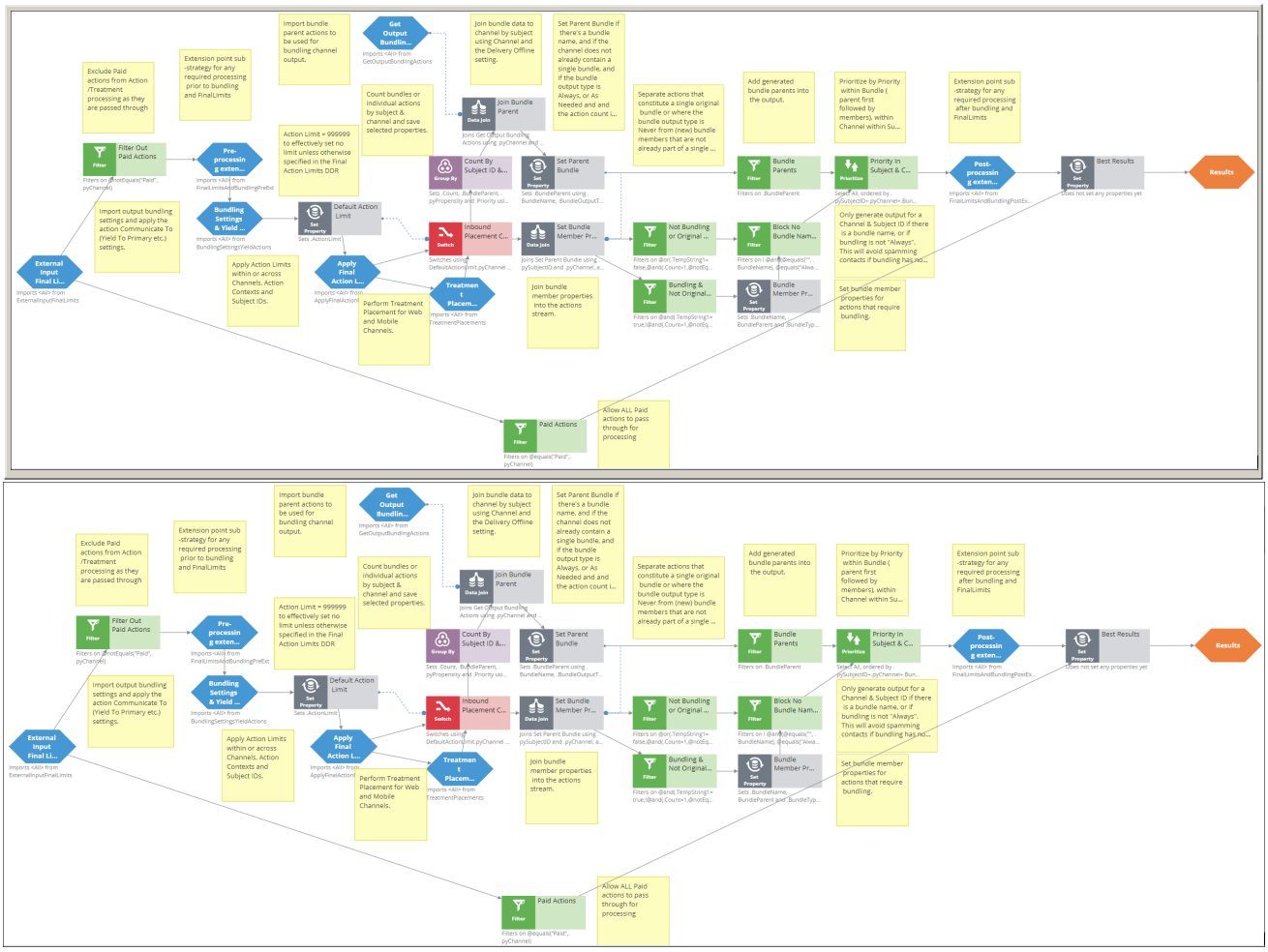
Creating an External Input strategy to test Final Limits

You can now use the External Input Final Limits strategy to replace the External Input in the Final Action Limits And Bundling strategy.

Pega Customer Decision Hub

1. In the Pega Customer Decision Hub portal, click **Intelligence Strategies** .
2. Open the *Final Action Limits And Bundling* strategy.
3. Click **Save As** and change the label to be the name of a new test strategy, for example, External Input Final Limits Test.
4. Ensure that **[No branch]** is selected in the **Development branch** field.
5. Click **Create and open**.

6. Right-click to the left of the External Inputs shape at the left side of the canvas.
7. Select **Sub-strategy External** .
8. Double click on the new component.
9. In the **External strategy** field, enter ExternalInputFinalLimits, and then enter a name for the component, for example, External Input Final Limits .
10. Connect the new **External Input Final Limits** sub-strategy to the same components to which External Input is joined, that is, the **Filter Out Paid Actions** filter shape and the **Paid Actions** filter shape, as in the following figure:



11. Right-click on an empty area of the canvas, and then click **Disable external input**.
12. In the **Input Disable Warning** message, click **Submit**.
13. Save and check in your changes.

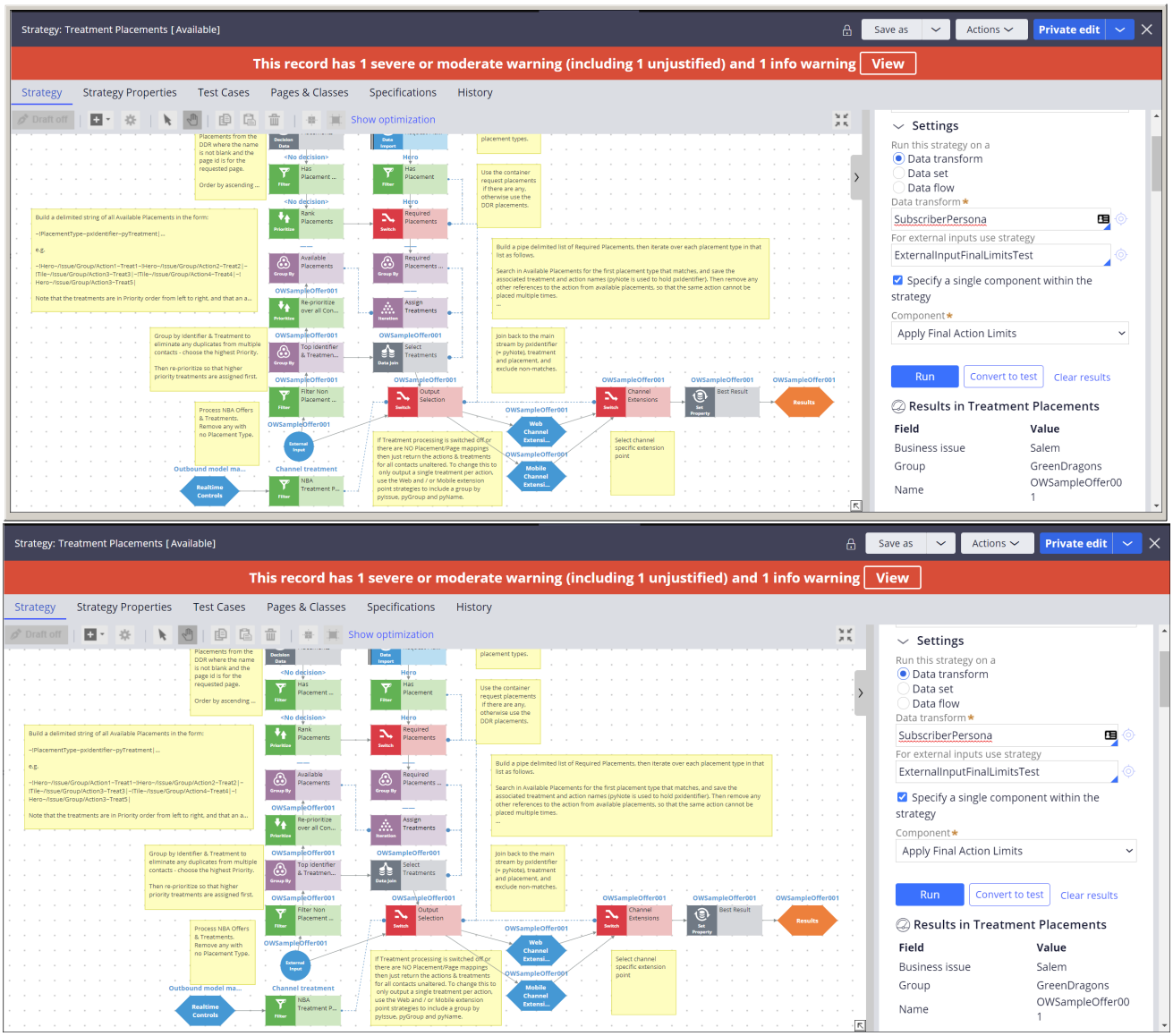
Result: You have created a test strategy that can be used to test sub-strategies within the Final Action Limits And Bundling strategy.

Testing the Treatment Placements component

This article provides an example of how to test a sub-strategy with the Final Action Limits And Bundling strategy, a similar technique can be used for strategies other than Treatment Placements.

Pega Customer Decision Hub

1. In the Pega Customer Decision Hub portal, click **Intelligence Strategies** .
2. Open the *Final Action Limits And Bundling* strategy.
3. From the Final Action Limits And Bundling strategy, right-click the Treatment Placements component, and then click **Open strategy**.
4. In the **Settings** section, select **Data transform**, and then select a persona transform.
   Note: To make any changes to the persona for the test, follow the steps used when setting up the person for the trigger strategy tests.
5. In the **For external inputs use strategy** field, enter the trigger strategy name, for example, ExternalInputFinalLimitsTest.
6. Select **Specify a single component within the strategy** .
7. Select the component that immediately precedes the component you are testing.
   For the Treatment Placements strategy, select Apply Final Action Limits.



8. Click **Run**.

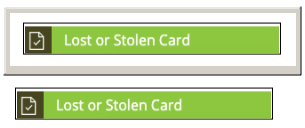
Configuring Next-Best-Action Designer for Pega Customer Service

If you use the Pega Customer Relationship Management suite of applications to share data and service cases across the Pega Customer Service and Pega Customer Decision Hub applications, Pega Customer Service can display the customer's next best action in the customer composite screen, based on the customer's profile and the reason for calling. With the release of Next-Best-Action Designer version 2, to display the next best actions within Pega Customer Service, you must configure the Next best action widget which displays them.

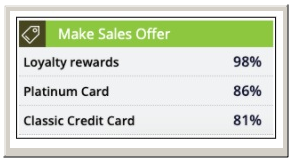
Pega Customer Decision Hub

The following next-best-action types can be displayed within Pega Customer Service:

Service actions
: These actions correspond to service flows within Pega Customer Service. On the display, the actions appear as a green bar, as in the following figure:

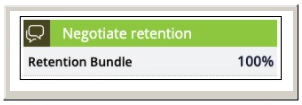


Standard actions
: These actions are grouped together based on the Issue (that is, Sales) and invoke the same flow within Pega Customer Service, but display specific action data based on the selected action.





Negotiation actions
These actions are displayed individually and invoke a negotiation service flow within Pega Customer Service.





- Configuring the Next best action widget
- Configuring actions in Pega Customer Service

Configuring the Next best action widget

To configure the Next best action widget which displays Pega Customer Decision Hub actions in Pega Customer Service, update the *MapActionForIssueAndGroup* decision table.

Pega Customer Decision Hub

The *MapActionForIssueAndGroup* decision table uses the *ActionID* and *Action* properties to control how actions are displayed within Pega Customer Service, based on their issue and group in Pega Customer Decision Hub.

ActionID
The value of this property needs to match an entry in the *MapActionProperties* decision table in the Pega Customer Service application.

Action
The value that the customer service representative sees in the green bar in the Next best action widget.

1. Log in to Pega Customer Decision Hub as an operator with access to Dev Studio.
2. In Dev Studio, edit the *MapActionForIssueAndGroup* decision table.
3. On the **Results** tab, expand the **Additional Allowed Results** section.
4. In the **CRMSamples** section, review the set of default actions, as in the following table:

Result	Target property	Value
Default	.ActionID	.pyIssue + "action"
	.Action	"Make " + .pyIssue + " action"
ServiceAction	.ActionID	.pyName
	.Action	.pyLabel
SalesAction	.ActionID	TopOffers
	.Action	"Makes Sales Offer"
NegotiateRetention	.ActionID	RetentionPeak
	.Action	"Negotiate retention"
NegotiateSales	.ActionID	SalesNegotiator
	.Action	"Negotiate sales"

5. Make sure the *ActionID* matches entries in the *MapActionProperties* decision table in Pega Customer Service.
6. Map your issues and groups to these allowed results by clicking on the **Table** tab of the *MapActionsForIssueAndGroup* decision table.
7. Add a row for each of your issues and map it to the appropriate allowed results, as in the following table:

Business Issue	Business Group	Return
Sales		SalesAction
Service		ServiceAction
Negotiation	Retention	NegotiateRetention
Negotiation	Sales	NegotiateSales

Configuring actions in Pega Customer Service

After configuring the mapping in the Pega Customer Decision Hub, actions should now appear in the Next best action widget in Pega Customer Service. To ensure that they appear as required and launch the correct service flow, modify the *MapActionProperties* decision table in Pega Customer Service.

Pega Customer Decision Hub

1. Log in to Pega Customer Service as an operator with access to Dev Studio.
2. Open the *MapActionProperties* decision table.
3. On the **Table** tab, review the list of entries.

   The list should contain an entry for each possible *ActionID* sent by Pega Customer Decision Hub.

4. Map each ActionID to an allowed result.
   This includes the following properties:
   - The service class and name of the flow that starts when a customer service representative selects the action.
   - How the action is displayed to the customer service representative.

Upgrading the Next-Best-Action Designer from legacy versions

In the Pega Customer Decision Hub, Next-Best-Action provides automated decisions based on predictive analytics, adaptive analytics, and traditional business rules. With the Next-Best-Action Designer, you can define Next-Best-Action decisions and requirements which apply to the whole organization, or to specific business issues or groups.

Pega Customer Decision Hub

Pega Marketing 8.3 introduced the new Next-Best-Action Designer version 2. Pega Customer Decision Hub 8.5 expanded the capabilities of Next-Best-Action Designer version 2 by adding enhancements that support multilevel context decisioning. If you defined Next-Best-Action using one of the previous versions of Next-Best-Action Designer, you can

continue to use the legacy configuration. However, to benefit from the new features, upgrade your Next-Best-Action configuration.

Note: If you never implemented the Next-Best-Action Designer configuration in Pega Marketing or Pega Customer Decision Hub 8.2 or earlier, open and save your Context Dictionary page after the upgrade to ensure that the Next-Best-Action Designer is accessible. For more information, see Resaving the Context Dictionary configuration.**Caution:** Before you upgrade the Next-Best-Action Designer in a production environment, first perform the upgrade in a sandbox environment and review the performance of your strategies to ensure that they continue to give the expected results. After the Next-Best-Action Designer is upgraded, it cannot be downgraded back to a previous version.

- Migrating Next-Best-Action Designer from version 1 to version 2
- Updating Next-Best-Action Designer version 2 to support multilevel configurations

Migrating Next-Best-Action Designer from version 1 to version 2

When planning the migration, consider the various environments that you may have in your pipeline. First do the migration in a single environment, in a dedicated, unlocked upgrade ruleset version, and then migrate all the rule changes to other environments in the pipeline.

Pega Customer Decision Hub
To migrate the Next-Best-Action Designer:

1. Suspend any outbound schedules defined for the existing Next-Best-Action configuration defined in Next-Best-Action Designer version 1.
2. Ensure that all Pega Customer Decision Hub assets are checked in.
3. Ensure that the ruleset and version used by your access group are unlocked, or create an unlocked version of the ruleset. By default, the ruleset used to store the configuration is *Artifacts*.
4. Ensure that the ruleset and version used by your Context Dictionary is unlocked. You can check the ruleset and version on the **Settings Context Dictionary** landing page in App Studio.
5. In Dev Studio, click **Records SysAdmin Dynamic System Settings** .
6. Click **Create**.
7. Fill out the following information:
   - **Short description**: optionally, enter the description of this Dynamic System Setting, for example, Enables the upgrade of Next-Best-Action Designer to version 2.
   - **Owning Ruleset**: enter PegaMKT-Engine.
   - **Setting Purpose**: enter MKTEnableNBADversion2Upgrade.
8. Click **Create and open**.
9. Set the value of the setting to **true**.
10. Click **Save**.
11. Log off from Pega Customer Decision Hub and then log back in.
12. In App Studio or in the Pega Customer Decision Hub portal, click **Next-Best-Action Designer** . The Next-Best-Action Designer now shows the option to upgrade to version 2.
13. Click **Migrate now**. The configuration of Next-Best-Action and its strategy framework is upgraded.
14. If you do not plan to implement action treatments, introduced in Pega Marketing 8.2, deactivate Next-Best-Action treatment processing. This ensures that offers created before the upgrade function correctly. After you associate your offers with treatments, you can enable treatment processing again. For more information, see Treatments and channels.
    1. Click the **gear** icon in the upper right corner of the Next-Best-Action Designer.
    2. Set the **Channel treatment processing** toggle to **OFF**.
    3. Click **Submit**.

What to do next: After migrating Next-Best-Action Designer, review and test the migrated strategies to ensure that they work with the new configuration.

Updating Next-Best-Action Designer version 2 to support multilevel configurations

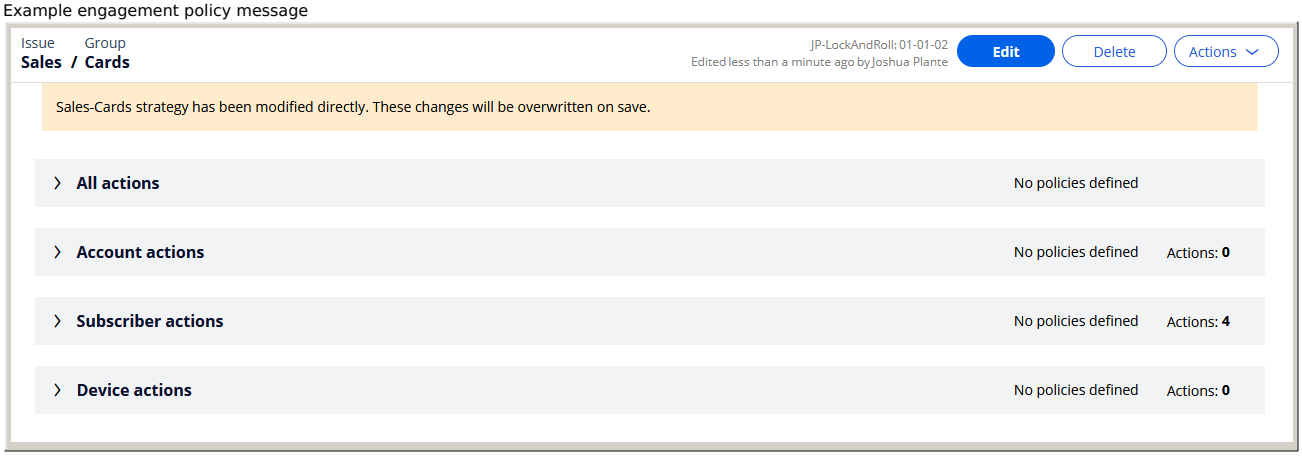
If you already used Next-Best-Action Designer version 2 in versions of Pega Customer Decision Hub earlier than 8.5, after upgrading to Pega Customer Decision Hub 8.5, you must update your Next-Best-Action Designer configuration to support multilevel decisioning.

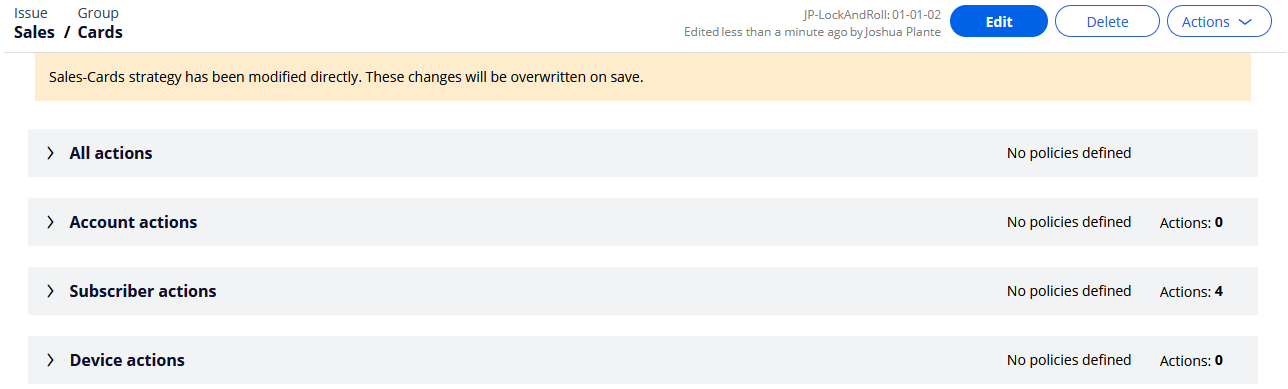
Pega Customer Decision Hub
When planning the update, consider the various environments that you may have in your pipeline. First do the update in a single environment, in a dedicated, unlocked upgrade ruleset version, and then migrate all the rule changes to other environments in the pipeline.

1. Ensure that no next-best-action outbound runs are active at moment and that no runs are planned for the same time frame as the Next-Best-Action Designer upgrade. **Note:** The upgrade can take 30-45 minutes.
2. Ensure that no inbound requests, such as real-time event and real-time container requests, are planned for the same time frame as the Next-Best-Action Designer upgrade.
3. Ensure that the Context Dictionary configuration is saved and valid.
4. Ensure that the ruleset and version used by your access group are unlocked, or create an unlocked version of the ruleset. By default, the ruleset used to store the configuration is Artifacts.
5. **Optional:** Check in all checked-out artifacts related to Next-Best-Action Designer to avoid potential issues during the upgrade.
6. **Optional:** Lock and roll the Artifacts ruleset version. For more information, see https://community.pega.com/knowledgebase/articles/pega-cloud-services-post-upgrade-tasks-7x-8y/cloud/locking-and-rolling-ruleset-versions.
7. Click **Update now**.

What to do next: After the update, when viewing an engagement policy in Next-Best-Action Designer, you may see a message stating that the backing strategy has been updated manually, as in the following figure:

Example engagement policy message





The message does not indicate an error and can be safely ignored. To remove the message, you can edit the engagement policy and save it without making any changes.

Importing Next-Best-Action settings from a template

To accelerate the process of implementing next-best-action, use predefined templates that contain next-best-action settings.

Pega Customer Decision Hub

Pega provides industry-specific templates with the recommended business structure, engagement policy, arbitration, and channel settings for the Communications, Healthcare, Retail Banking, and Insurance industries. You can import these templates to quickly set up the basic settings for your organization. In addition to the predefined templates, you can save your own Next-Best-Action settings as a template, and then import the template into a different Pega Customer Decision Hub instance, for example, to transfer the settings from a stage to a production environment.

1. Go to Pega Marketplace.
2. In the **Search Marketplace** field, enter NBA Designer Template.
3. Browse the list of available templates, and download the template that is applicable to your industry.
4. In the Pega Customer Decision Hub portal, click **Next-Best-Action Designer** .
5. **Optional:** If you previously configured or imported Next-Best-Action settings, you can change the template by clicking **Taxonomy Actions Switch template** .
   Important: The template change overwrites any settings that you previously configured.
6. Upload the next-best-action template.
   Tip: Templates developed for a specific industry may require certain artifacts to exist in your application. For example, a template created for Pega Customer Decision Hub for Financial Services expects that the required customer contexts exist in your application. To ensure that the template can be used in your application, do one of the following steps:
   - Use templates applicable to your industry.
   - Import the relevant artifacts into your application before you select the next-best-action template.
   - During template import, clear the **Engagement policy** check box. Clearing the check box imports only the taxonomy without the engagement policies defined for industry-specific contexts.
7. Review the imported settings and adjust them manually as needed.

Manually defining the structure of business issues and groups

Use the Next-Best-Action Designer to build the structure of business issues and groups in your organization.

Pega Customer Decision Hub

Business issues are different areas of focus for your business. A business issue is the purpose behind the actions you offer to your customers. For example, actions which started with the purpose of retaining existing customers should be grouped under the business issue of Retention. Actions started with the purpose of acquiring new customers belong to the business issue of Acquisition.

Business groups organize customer actions into categories. For example, as part of the business issue of Acquisition, you can create groups for products like Credit Cards, Mortgages, or Personal Loans, with the intention of offering these to potential new customers.

Together, business issues and groups form the organizing structure for your customer interactions. Each action you prepare for your customers is associated with a business issue and a group.

1. Log in as an operator whose user group has the *CanManageNBA* privilege.
   By default, this privilege is granted to managers and administrators through the *PegaNBAM_FW:MarketingManager* role.
2. In the Pega Customer Decision Hub portal, click **Next-Best-Action Designer Taxonomy** .
3. Click **Edit**.
4. To add an issue, click **Configure issues**.
5. Enable issues for Next-Best-Action by performing the following tasks:
   - To enable an existing issue to use Next-Best-Action, click on it in the **Configure Business Issues** window.
   - To add a new issue, click **+ Create** in the **Configure Business Issues** window.
6. Click **Apply**. **Result:** Pega Customer Decision Hub adds the group to the business structure.
7. **Optional:** If you want the business structure to display the issue description in the Description column, click **More Edit issue** , and then fill out the **Short description** field.
8. Click **More Configure groups** .
9. Enable groups for Next-Best-Action by performing the following tasks:
   - To enable an existing group to use Next-Best-Action, click on it in the **Configure Business Groups** window.
   - To add a new group, click **+ Create** in the **Configure Business Groups** window.
10. Click **Apply** to save your changes. **Result:** Pega Customer Decision Hub adds the group to the business structure.
11. **Optional:** If you want the business structure to display the group description in the Description column, click **More Edit group** by the name of the group, and then fill out the **Short description** field.
12. Click **Submit**.
13. Click **Save**.

- Defining an alias for a group of actions
- Best practices for configuring the business structure

Defining an alias for a group of actions

By default, Pega Customer Decision Hub uses the term actions to refer to the engagement messages. If needed, you can change this to a more appropriate term, such as Promotions or Nudges.

Pega Customer Decision Hub
Tip: You can use different terms for actions based on the business issue and group to which the product belongs. For example, if your business structure includes a Loans group within the Retention issue, you can rename actions related to that issue and group to Loan Loyalty Program. For other issues and groups, you may use other, group-specific terms, or keep the default name.

1. In the Taxonomy tab of Next-Best-Action Designer, click **More Edit group** by the name of the group.

2. In the **Action naming (singular)** field, enter the name of actions from this group, for example, Promotions.
3. Click **Submit**.
4. Click **Save**.

Best practices for configuring the business structure

The following table shows the best practice business structure for a Financial Services application.

Pega Customer Decision Hub
Note: You can use one of the templates published in the Pega Marketplace. Templates define a standard configuration to accelerate your implementation. If you want to build the hierarchy manually, follow the best practices below.

Priority	Issue name	Issue description	Group name	Group description	Sample propositions
1	Collections	Recommendations for customers who are currently in arrears, or are at high risk or entering arrears.	Pre-collection	Recommendations for customers at risk of entering arrears.	• "It's time to reduce your high balance" • "Setup autopay today!" • "Call us if you are experiencing financial difficulty, we can help"
			Post-collection	Recommendations for customers in arrears.	• "Please pay your amount due" • "Call us"
			Payment plans	Personalized repayment plans based on outstanding balance and credit risk. Includes logic to determine if interest or principal forgiveness is appropriate.	• "Here is a plan to pay weekly" • "Call us to find out about reducing your balance"
2	Service	Servicing activities appropriate for or pending with this customer as this time.	Account	Account-related servicing recommendations for the customer.	• "Would you like to increase your credit line?" • "Setup auto-pay" • "Would you like to add a beneficiary?"
			Customer	Customer-related servicing recommendations.	• "We need your updated address" • "Missing contact information" • "Would you like to complete the issue you started on call center?"
3	Retention	Retention actions for customers with a high churn risk (that is, likely to close account or reduce business with the bank).	Proactive	The customer is at high risk of reducing product holdings, or reducing engagement with the bank.	• "Here is an offer for twice the points you usually receive on your card"
			Reactive	The customer has shown a reduced activity with the bank, or closed an account.	• "We want you back, here is a special offer"
4	Sales	Sales offers and messages for customers and prospects. This includes upsell, cross-sell, and standard messages.	Home loans	Specific offers for collateralized loans in real estate.	• "FirstMortgage30yr"
			Auto loans	Specific offers for collateralized loans for automobiles and other transport.	• "Personalized loan for you - 48 months at 8% APR"
			Personal loans	Non-collateralized loans for individuals (not corporations).	• "Unsecured personal loan"
			Personal cards	Credit card offers for individuals (not corporations).	• "WorldMastercard"
			Personal deposits	Savings, checking, and other deposit account offers.	• "PremiumChecking"
			Insurance products	Insurance products (applicable outside North America).	• "Home Insurance" • "Travel cancellation insurance"
			Brokerage	Investment product offers.	• "Investment account" • "Rollover IRA"
			Home loans	The primary product is a home loan with ancillary products.	• "FirstMortgage30yr"
			Auto loans	The primary product is an auto loan with ancillary products.	• "Used48Mo"

Priority	Issue name	Issue description	Group name	Group description	Sample propositions
5	Bundles	Product packages for customers who are candidates for right-sizing (that is, bundling, increasing share of wallet).	Credit cards	The primary product is a credit card with ancillary products.	• "WorldMastercard"
			Deposit accounts	The primary product is a deposit account with ancillary products.	• PremiumChecking
			Insurance	The primary product is insurance with ancillary products.	• "Home Insurance"
			Brokerage	The primary product is an investment offer with ancillary products.	• "Rollover IRA"
6	Nurture	Messages that enhance customer engagement by highlight the benefits of current product holdings.	General	General messages for customer engagement.	• "Welcome to U+ Bank"
			Home loans	Messages to increase customer engagement with home loan products.	• "FirstMortgage30yr"
			Auto loans	Messages to increase customer engagement with auto loan products.	• "We have partnered with GEICO for insurance"
			Credit cards	Messages to increase customer engagement with credit card products.	• "Your card does not incur FX fees"
			Deposit accounts	Messages to increase customer engagement with deposit account products.	• "Did you know we have over 10 000 ATMs nationally?"
			Brokerage	Messages to increase customer engagement with investment products.	• "Adding a beneficiary is a good way to..."

Managing actions

An action holds various details about a particular offering, such as Start Date, End Date, and Expected Revenue. In addition to the proposition data, the Action rule enables the user to visually specify the flow of steps for a particular offering (Flow).

Pega Customer Decision Hub
Important: To disable an action so that the next-best-action strategy framework does not select it, set the Availability action attribute to Never or set the **End date** to a date in the past. Do not withdraw the action rule using the Pega Platform construct of rule availability as this approach will not prevent the next-best-action strategy framework from selecting it.

Each action is backed by a Decisioning proposition. The system automatically manages this relationship by creating, deploying, and deleting the proposition instance as needed. Since an action is closely tied to a proposition, it must always be created in the context of an Issue and a Group.

The actions landing page lists available action rules in the system. This landing page is accessible in the Pega Customer Decision Hub portal via: **Content Actions** .

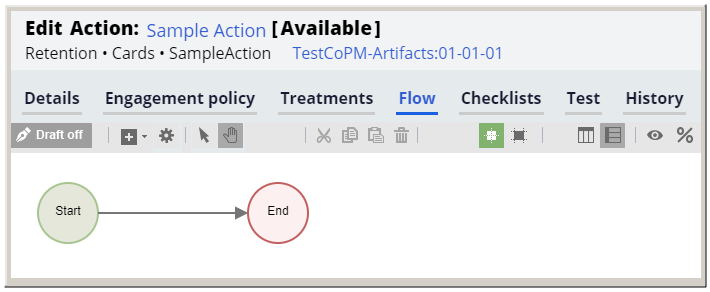
This landing page lists the twenty most recently updated actions in the system. Users can use the **View more results** at the end of the list to load the next twenty results. Users can also use the filters at the top of the page to search for actions by Issue / Group and Name.

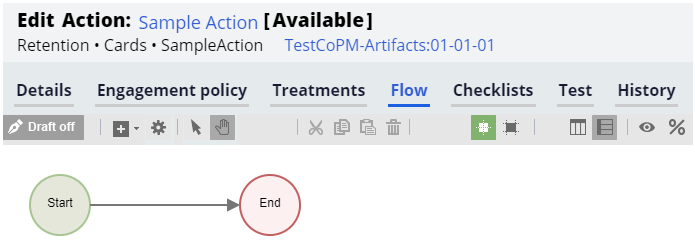
For each listed action, the landing page displays the following information:

- Rule name - Clicking the name opens the Action rule.
- Friendly name
- Action availability - Start and/or end date of the action, if configured
- Impression rate of this action - Ratio of total impressions (associated with this action) and the action volume
- Action volume - Total number of times this action has been initiated
- Conversion rate of this action - Ratio of total conversions (associated with this action) and the action volume
- Icons indicating the channels utilized by the action
- Image associated with the action

The **Create** button enables marketers to create a new action.

A newly created action starts out with the Start and End shapes, as shown below.





The action rule contains the following tabs:

Tab Name	Purpose
Diagram	Create the Flow. Refer to the Defining action flows section.
Details	Specify the details about the offering. Refer to the Creating actions section.
Test Action	Test the Flow. Refer to the Testing the action flow section.
History	Specify the Description and Usage information for this action.

- Creating actions
- Monitoring actions
- Reviewing action performance
- Model learning for new actions

Creating actions

Various details about the action can be specified on the **Details** tab of the action rule. These details can be used in Strategies for making decisions. They can also be included in treatments.

Pega Customer Decision Hub
Note: Each of the data fields listed in the following sections is part of the proposition associated with the action. To utilize a particular field in a different part of the application, reference the associated property using the OfferData page under the action class.

Action details are classified into the following groups:

- Basic Action Attributes
- Bundle Attributes
- Custom Attributes
- Creating a New Attribute
- Defining action flows
- Importing and exporting actions

Basic Action Attributes

Basic action attributes are available for every action and are categorized into the following sections:

Pega Customer Decision Hub

- Details
- Availability
- Financials
- Expected Outcomes

- Details
- Next-Best-Action configuration
- Availability
- Financials
- Expected Outcomes

Details

The following table describes the data fields contained in the **Details** section:

Pega Customer Decision Hub

Input Field	OfferData Property	Description / Purpose
Key code	KeyCode	Marketing identifier for this action. Users can output this value in treatments and/or use it internally to track action performance.
Variant	Variant	Variation in a set of actions (e.g. size, color, etc.). This property is used in action bundling for Advisor. Variants for the same action will be grouped and selectable during bundling.
Category	Category	Classification of the action. This property is used in the Advisor Negotiation use case.
Description	ShortDescription	Brief description of the action. This property is used in the Advisor Assessment and Negotiation use cases.
Benefits	Benefits	Benefits associated with the action
Image URL	ImageURL	Use this field to provide a link to an image to display for this action when it is rendered as part of a Real-Time Container.
Click through URL	ClickThroughURL	Use this field to provide a link to a web page or a file that provides more details on this action. When this action is included in a Real-Time Container, clicking this action in the Container will re-direct to the URL specified in this field. **Note:** For proper re-direction, this URL should begin with the protocol prefix - such as http:// or https://
Thumbnail image	pyImage	Use this field to reference an image (Binary File rule) which represents this action. This image displays in the details side bar, when the action is referenced in Outbound Campaigns (in the Action card and the Add action configuration window), and in the Actions landing page.
Starting Evidence	StartingEvidence	Number of action responses required before a model is reliable. This field is used in conjunction with Starting Propensity in the adaptive model propensity smoothing formula.
Starting Propensity	StartingPropensity	Default propensity for the action before the model is reliable. This field is used in conjunction with Starting Evidence in the adaptive model propensity smoothing formula.

Next-Best-Action configuration

The following table describes the data fields contained in the **Next-Best-Action configuration** section:

Pega Customer Decision Hub

Input Field	OfferData Property	Description / Purpose
Transactional		This option is **off** by default. Only set the toggle to **on** if you do not want to apply the usual contact policies to this action, for example, because the action is a regulatory communication which the client must receive regardless of contact policies. For more information about contact policies, see Setting constraints, contact policy limits, and controls.
Apply machine learning to calculate priority		This option is **on** by default. Only set the toggle to **off** if you do not want to automatically calculate the action priority using adaptive model scores, for example, because the action is a regulatory communication and usual priority calculations do not apply. If this option is deactivated, Pega Customer Decision Hub uses the **Starting propensity** and **Business weight** values to determine action priority.
Starting evidence	StartingEvidence	This value indicates how many action responses must be collected from customers before a model is reliable. This field is used in conjunction with Starting Propensity in the adaptive model propensity smoothing formula.
Starting propensity	StartingPropensity	Before the adaptive models have enough data to be reliable, this value indicates the default propensity for the action. This field is used in conjunction with **Starting Evidence** in the adaptive model propensity smoothing formula.
Business weight (+/-)	pyWeight	Assign weight to the action. The action weight is used by next best action to calculate propensity, according to the settings defined on the **Arbitration** tab of Next-Best-Action Designer. For more information, see Prioritizing actions based on customer relevance and business priority.
Business value	pyValue	Assign a financial value to the action. The action value is used to prioritize actions. You can also configure the *Calculate Business Value* and *Calculate Value Extension* strategies to implement custom calculation logic for action values. For more information, see Prioritizing actions based on customer relevance and business priority

Availability

Options in the **Availability** section determine if an action is active (enabled) or inactive (disabled). The next-best-action strategy framework selects an active action but filters out an inactive action.

Pega Customer Decision Hub

The following table describes the fields contained in this section:

Input Field	OfferData Property	Description / Purpose
Always	*pyIsPropositionActive*	Action is active. The next-best-action strategy framework selects the action.
Never	*pyIsPropositionActive*	Action is inactive. The next-best-action strategy framework filters out the action.
Within a defined time period	Not applicable	Action is active only within a time period specified by the **Start Date** and **End Date** fields.
Start Date (displayed when **Within a defined time period** option is selected)	*StartDate*	Date the action becomes active.
End Date (displayed when **Within a defined time period** option is selected)	*EndDate*	Date the action is no longer active.
Relevant description	*WhyRelevant*	Description of the circumstances under which this action is relevant for a customer.
Eligibility description	*EligibilityDescription*	Description of the circumstances under which this action is eligible for a customer.

Financials

The following table describes the data fields contained in the **Financials** section:

Pega Customer Decision Hub

Input Field	OfferData Property	Description / Purpose
Original cost	OfferValue	MSRP or the full cost of this action
Marketing cost	BudgetedCost	Costs associated with the marketing of this action
Internal cost	InternalCost	Costs to the organization when a customer accepts this action. This amount is reflected in the Retention Negotiation budget.
Customer cost	Cost	Actual cost to the customer for this action
Agent compensation	AgentCompensation	Compensation amount for the agent if a customer accepts this action
Price description	Pricing	Description of the customer cost of this action (e.g. $45 per month)

Expected Outcomes

The following table describes the data fields contained in the **Expected outcomes** section:

Pega Customer Decision Hub

Input Field	OfferData Property	Description / Purpose
Expected response rate	ExpectedResponseRate	Expected rate of customer response to this action
Expected revenue	Expected revenue	Expected earnings from this action

Bundle Attributes

Bundles attributes are used to specify the bundling settings for the action. Refer to the Action Bundles chapter for more details.

Pega Customer Decision Hub

Custom Attributes

The **Custom Attributes** section lists other proposition fields defined for use by implementers of the application. The following sets of attributes display in this section:

Pega Customer Decision Hub

- All top level custom attributes
- All Issue level custom attributes belonging to the Issue for this action
- All Group level custom attributes belonging to the Group for this action

If the action belongs to a versioned Group, implementers can customize the display of attributes. In such a case, this section will utilize the customized form for specification of these attributes. For guidelines on creating custom forms for Decision Data rules, see Customizing the Decision Data rule form.

Creating a New Attribute

In order to specify a new action detail, the backing proposition attribute must first be created.

Pega Customer Decision Hub

To add a new custom attribute:

1. Go to the Proposition Management landing page. This page is accessible in the Pega Customer Decision Hub portal via: **Configuration Proposition Management** .
2. Use the controls in the **Hierarchy** tab to perform the desired action, as described in the following table.

To	Do
Create a top level attribute	Ensure **Top Level** is selected under the **Business issues & groups** tree. In the **Add** list, select **New property** to launch the New form.
Create an issue or group level attribute	Select the appropriate issue and/or group in the **Business issues & groups** tree. In the **Add** list, select **New property** to launch the New form.

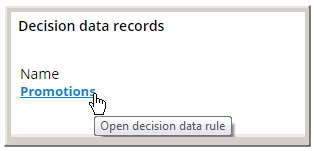
3. Specify the **Name** and **Type** of the new attribute in the **Create Property** form and click **Create**.

The next steps vary based on whether the application utilizes Decision Data rules to version propositions. Refer to Defining versioned propositions for more details.

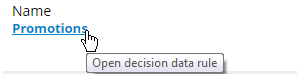
- Application doesn't version propositions - No further action is needed.
- Application versions propositions - The new attribute must be added to the Decision Data rule of each group where it needs to be utilized. This process is described below.

Adding an Attribute to a Group's Decision Data Rule

1. Open the Decision Data rule for the group. This rule can be opened by clicking the link in the **Decision data records** grid after selecting a group in the Hierarchy tab of the Proposition Management landing page.



Decision data records



2. Check out the Decision Data rule.
3. Switch to the **Form** tab.
    1. If the Decision Data rule uses a custom form, add the new attribute to the desired location in the custom form following Pega Platform development practices.
    2. If the Decision Data rule uses the standard form, click **Add field** in the **Form fields** section. In the new row, find and select the newly created attribute in the **Identifier** column.
4. Save the Decision Data rule and check it in.

Defining action flows

The Flow contains the exact sequence of actions to be performed in the lifecycle of a particular action. A Flow consists of a variety of shapes and the connectors which connect these shapes.

Pega Customer Decision Hub

The Flow editor provides the user with an intuitive drag-and-drop mechanism for designing their offerings. Users can configure the details of each step directly on the step's shape using the shape's Properties panel.

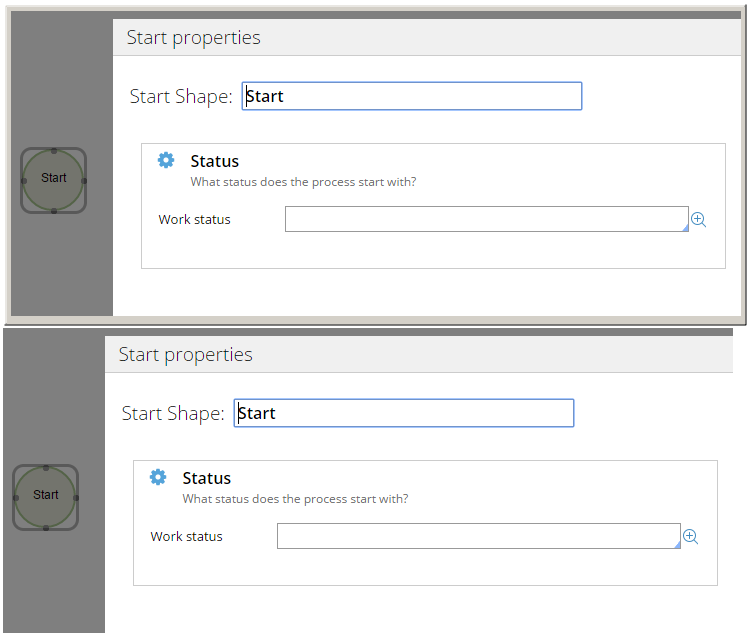
To apply a template, click the icon by the **Dynamic template** field, and then select the action which you want to use as template. The action diagram is updated to display the flow from the action selected as template.

- Understanding the action flow shapes
- Understanding the action flow connectors
- Branching the action flow based on the channel
- Grouping offerings into action bundles
- Configuring an action flow to write information to an external system
- Validating action flow configuration
- Testing the action flow
- Importing and exporting actions

Understanding the action flow shapes

The Start shape denotes the beginning of the Flow. Each Flow must have one and only one Start shape.

Pega Customer Decision Hub



- Configuring the delivery of an email treatment with the Send Email shape
- Configuring the delivery of an SMS treatment with the Send SMS shape
- Configuring the delivery of a generic treatment with the Send Generic shape
- Configuring the delivery of multiple treatments with the Send Multiple shape
- Configuring the delivery of passbook treatments with the Send Passbook shape
- Configuring the delivery of push notifications with the Push Notification shape
- Specifying the context for inbound applications with the Inbound Call Center shape
- Pausing the flow execution with the Wait shape
- Specifying the status of an action with the Update Status shape
- Launching background processes with the Hand Off shape
- Branching the action flow with the Decision shape
- Scheduling follow-up work with the Schedule Appointment shape

Configuring the delivery of an email treatment with the Send Email shape

The Send Email shape enables the user to configure an email message for delivery.

Pega Customer Decision Hub

The following general configuration options are available for this shape:

Treatment

Name	Purpose
Use associated Treatment	Select this option to use the treatment that you selected in the **Treatments** tab. To ensure that this option works correctly, enable channel treatment processing in **Next-Best-Action Designer settings**.
Specify Treatment	Select this option to manually specify a treatment other than the one selected in the **Treatments** tab.
Treatment Name	This option is available after you select the **Specify Treatment** option. The property which holds the name of the Email Treatment to send. In the case of Online Delivery, the Treatment Name specified must be backed by an existing Email Treatment rule since this is used as the content of the email to be sent.
Key Code	This option is available after you select the **Specify Treatment** option. Marketing code used to identify the treatment. This value can be output in the sent email and/or the output file/db, and can be used to track the performance of different treatments. If a key code value is specified on the Email Treatment being referenced, it will be populated into this field upon selecting the treatment.
Deliver Online	This option is available after you select the **Specify Treatment** option. Determines whether an actual email should be sent using the content in the specified Treatment Name. Clear this check box if you want Pega Customer Decision Hub to write the email to a file or database instead of sending it directly.
Use Subject From Treatment	This option is available after you select the **Deliver Online** option. Denotes that this shape should directly use the Email subject specified in the referenced Email Treatment. This subject is shown as read-only in the **Email Subject** field.
Specify Subject	This option is available after you select the **Deliver Online** option. Denotes that this shape should use the specified **Email Subject** instead of the subject specified in the referenced Email Treatment.
Email Subject	This option is available after you select the **Specify Subject** option. Denotes the subject of the email to be sent. For more information, see .
Write To File	This option is available after you select the **Specify Treatment** option. Determines whether a record should be written to an output file.
Write To DB	This option is available after you select the **Specify Treatment** option. Determines whether a record should be written to an output database table.
Select Template	This option is available after you select the **Write to File** or **Write to DB** option. Denotes the file/DB Template rule which contains the specifics of what should be output to the file/table. This includes the file/table name, location, header information, and the specific data fields that need to be output.
Wait after sending	Determine whether the flow should wait before moving on to the next step after sending an email. Enable this option if you expect a customer response to the email (such as accepting or rejecting an action). The wait options available are the same for all wait-based shapes and are described in the section on the Wait shape.

Configuring the email subject

The **Email Subject** field supports the following three formats:

1. Plain text – If the subject contains plain text (with no quoted words); simply enter the text in the input field. The system will wrap the entire text around quotes.
2. Quoted text – If the subject needs to include text in double quotes (e.g. The brand new “ProductName” is here!), then the entire subject must be in quotes with the desired quotes being escaped with a backslash. The example above should be entered as: “The brand new \”Product Name\” is here!”
3. Property references – If the subject needs to contain property references (such as the customer name), then the property reference needs to be concatenated to the remaining subject by the **plus (+)** symbol. Each block of text that is not a property reference should be wrapped in double quotes. This is exemplified below:

- Desired Subject: Hi <Customer's Name>, a new "Product" awaits you!
- Entered Subject: "Hi " + .Customer.pyFullName + ", a new \"Product\" awaits you!"

If an email subject does not meet the supported formats, the following error is displayed to the user: Invalid value specified for Email Subject. Value doesn't adhere to the Validate: BalancedQuotationMarks.

Email Account

Users also have the option to specify the Outbound Email account to use for sending the email message. These accounts are typically configured by the administrator. Refer to the *Email and SMS Account Configuration* chapter for more details.

The following options are available in the **Email Account** sub-section under the **Advanced** section of the Send Email Properties modal:

Name	Purpose
Use Default Email Account	Determines whether the Default Email account should be used for sending this email.
Specify Email Account	Determines whether a user-specified Email account should be used for sending the email.
Account	Name of the Outbound Email account to use for sending the email.
From/Reply To:Use Email Account Settings	Denotes that the sender's name (from) and reply-to address in the email being sent should be taken from the values specified on the Email account.
From/Reply To:Use Operator Settings	Denotes that the name and email address of the operator that initiates this email should be used for the sender's name and reply-to address.
From/Reply To:Override Settings	Denotes that user-specified values should be used for the sender's name (from) and reply-to address in the email being sent.
From	Name that will be used as the sender's name in the email being sent.
Reply To	Email address that will be set as the reply-to address in the email being sent.

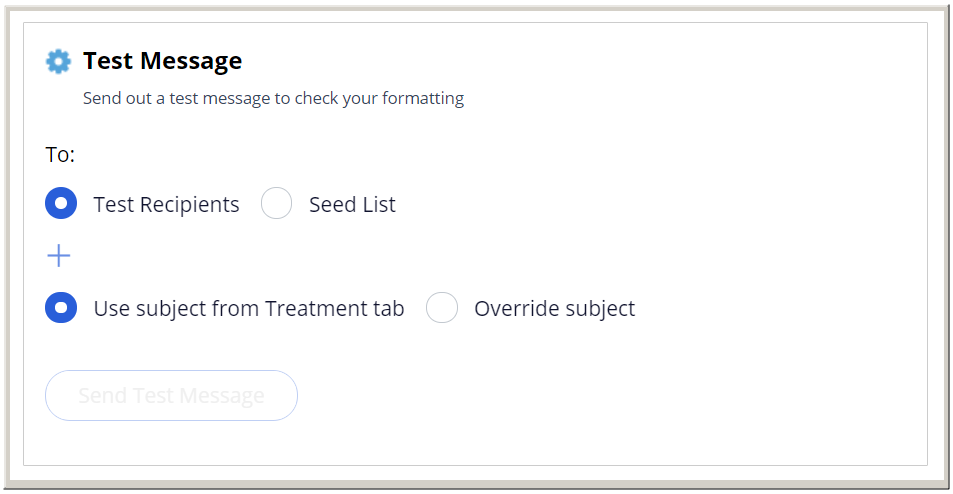
Testing the Send Email shape

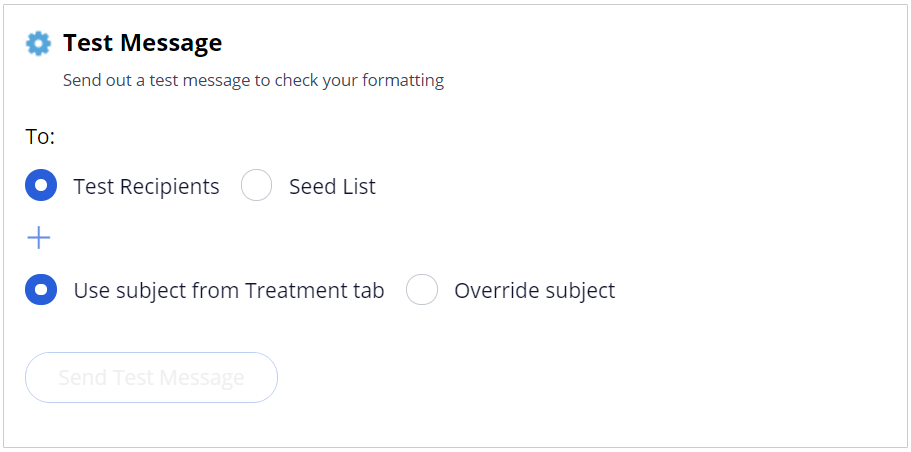
The Send Email shape provides an easy mechanism to test its outcome. Users can employ this test mechanism to test the online delivery of the Email Treatment. This can be beneficial in verifying both the content of the email being delivered and any email configurations that were applied (such as the subject, sender's name, and reply-to address). Finally, this also enables the user to verify any data references (both Action Data and Customer) in the Email Treatment that were used to personalize the email message.

The **Test Message** section lists the options available for configuring the test email message. This includes the following:

- **To** – Recipients of the test email. Select either **Test Recipients** or **Seed List**.
  - **Test Recipients** - Enter list of recipients. Each entry can either be the Operator ID of an existing user in the system or a valid email address. If the operator ID is entered, operator information such as the name and email is populated as the customer data and the test email is sent to the operator's email address.
  - **Seed List** - Select a predefined Seed List as the intended recipients of the test email. The data for each seed is populated as the customer data and is visible (if referenced) in the test email. In this case, the Email1 column in the Seed List (**.pyEmail1 property**) specifies the email address to which the test email is sent. Refer to the *Seed Lists* chapter for information on configuring a Seed List.
- **Subject** – Subject of the test email. This value is pre-populated with the subject specified in the Treatment tab. The user can choose to specify a different subject by selecting the **Override subject** option. Refer to the *Configuring Email Subject* section above for options available for this field.

After the test message has been configured, it can be sent via the **Send Test Message** button.





Configuring the delivery of an SMS treatment with the Send SMS shape

The Send SMS shape enables the user to configure an SMS message for delivery.

Pega Customer Decision Hub

The following General Configuration options are available for this shape:

Name	Purpose
Treatment Name	The property which holds the name of the SMS Treatment to be sent.
Key Code	Marketing code used to identify the treatment. This value can be output in the sent SMS message and/or the output file/db, and can be used to track the performance of different treatments. If a key code value is specified on the SMS Treatment being referenced, it will be populated into this field upon selecting the treatment.

Delivery Options

SMS treatments can be delivered in various ways - they can be sent by the system (Online mode) or they can be written to a data warehouse for processing by other systems (File and DB modes). Users can enable one or more of these modes at the same time.

The following Delivery Configuration options are available for this shape:

Name	Purpose
Deliver Online	Determines whether an actual SMS message should be sent using the content in the specified Treatment Name.
Use Existing SMS (Online mode)	Determines whether an existing treatment, specified by the Treatment Name property, should be used to get the content of the SMS message.
Quick Create SMS(Online mode)	Determines whether to use the text specified in the **Write Message** box as the content of the SMS message.
Write Message(Online mode)	Content of the message to be sent in **Quick Create SMS** mode.
Write To File	Determines whether a record should be written to an output file.
Write To DB	Determines whether a record should be written to an output database table.
Select Template (File/DB modes)	File/DB Template rule which contains the specifics of what should be output to the file/table. This includes the file/table name, location, header information, and the specific data fields that need to be output. Refer to the Templates chapter for more information.

Wait After Sending

Users can determine whether they would like to wait after sending an SMS message or proceed directly to the next step in the Flow. The wait options available are the same for all wait-based shapes and are described in the section on the Wait shape.

SMS Account

Users also have the option to specify the SMS account to use for sending the SMS message. These accounts are typically configured by the system administrator. Refer to the *Email and SMS Account Configuration* chapter for more details.

The following Account Configuration options are available in the **SMS Account** sub-section under the **Advanced** section of the Send SMS Properties modal:

Name	Purpose
Use Default SMS Account	Determines whether the Default SMS account should be used for sending this SMS message.
Specify SMS Account	Determines whether a user-specified SMS account should be used for sending the SMS message.
Account	Name of the Outbound SMS account to use for sending the SMS message.

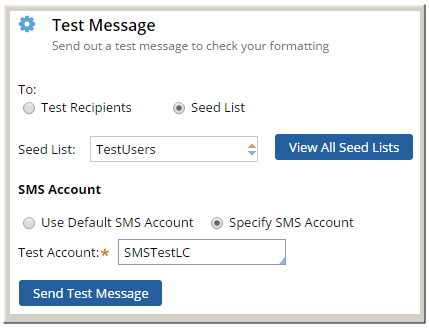
Testing the Send SMS shape

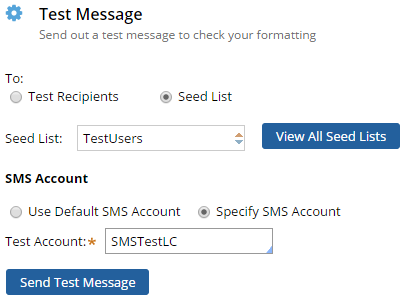
The Send SMS shape provides an easy mechanism to test its outcome. Users can employ this test mechanism to test the online delivery of the SMS Treatment. This can be beneficial in verifying the content of the SMS being delivered. This also enables the user to verify any data references (both Action Data and Customer) in the SMS Treatment that were used to personalize the SMS message.

The **Test Message** tab lists the options available for configuring the test SMS message. This includes the following:

- **To** – Recipients of the test message. Select **Test Recipients** or **Seed List**.
  - **Test Recipients** - Enter list of recipients. Each entry can either be the Operator ID of an existing user in the system or a valid mobile phone number. If the operator ID is entered, operator information such as the name and phone is populated as the customer data and the test SMS message is sent to the operator's phone number.
  - **Seed List** - Select a predefined Seed List as the intended recipients of the test SMS message. The data for each seed is populated as the customer data and is visible (if referenced) in the test message. In this case, the Mobile Phone column in the Seed List (**.pyMobilePhone property**) specifies the phone number to which the test SMS message is sent. Refer to the "Seed Lists" chapter for information on configuring a Seed List.
- **Test SMS Account Options** – These settings specify the SMS account (Default or custom) used to send the test SMS message.

After the test message has been configured, it can be sent via the **Send Test Message** button.





Once configured, the test message can also be directly sent via the “Test this SMS” right-click menu item on the Send SMS shape. Upon sending a test message, the status of the test is displayed. Any errors in the test delivery are displayed in case of a failure. These can be corrected and the test SMS can be resubmitted for delivery.

Configuring the delivery of a generic treatment with the Send Generic shape

Use the Send Generic shape to configure and send a Generic treatment.

Pega Customer Decision Hub

This shape provides a hook to process actions via outbound delivery mechanisms, other than the ones currently provided by the system (such as email or SMS). Use this shape to output relevant data to an output file or database table for processing by legacy systems.

The following General Configuration options are available for this shape:

Name	Purpose
Treatment Name	Name of the treatment. This can be used to track and organize the output data.
Key Code	Marketing code used to identify the treatment. This value can be output in the output file or table and can be used to track the performance of different treatments.

Delivery Options

Generic treatments can be written to a data warehouse (file or database table) for processing by other systems. Users can enable one or more of these modes at the same time.

The following Delivery Configuration options are available for this shape:

Name	Purpose
Write To File	Determines whether a record should be written to an output file.
Write To DB	Determines whether a record should be written to an output database table.
Select Template (File/DB modes)	File or Database Template rule which contains the specifics of what should be output to the file or table. This includes the file or table name, location, header information, and the specific data fields that need to be output. Refer to the Templates chapter for more information.

Wait After Sending

Users can determine whether they would like to wait after sending a generic treatment or proceed directly to the next step in the Flow. The wait options available are the same for all wait-based shapes and are described in the section on the Wait shape.

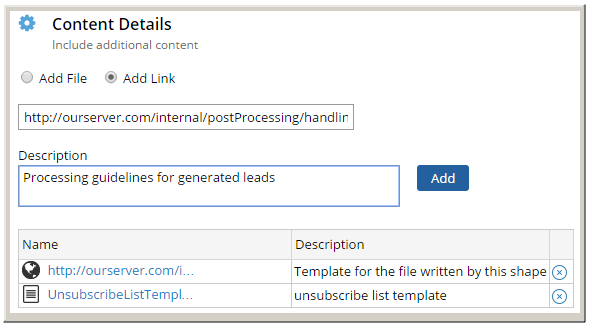
Field Marketing Options

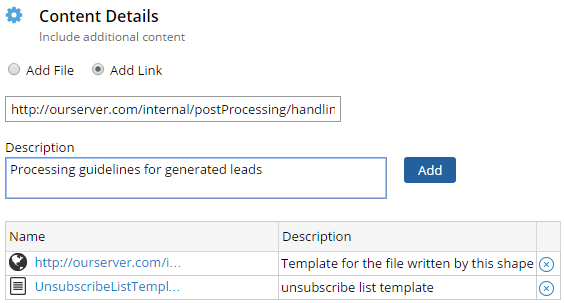
This set of options is available when the application includes Pega Field Marketing. Corporate marketers can enable the checkbox to include this shape in the list of engagement steps that field marketers see in Field Marketing Campaigns. Additionally, they can also provide a meaningful description to aid the field marketer in understanding this shape’s purpose.

Content Details

The **Content Details** section allows the user to upload related content for this shape. If this shape is configured to display in Field Marketing Campaigns, then field marketers will be able to view the uploaded content when they review this shape in their Campaigns.

Both links and files can be uploaded using this section. The user can also enter a description for each item. Once the file or link is added, it displays in the content grid at the bottom of this section.





Configuring the delivery of multiple treatments with the Send Multiple shape

The Send Multiple Treatments shape enables the user to configure treatments to be delivered across different channels (currently Email, SMS, or Passbook). The user also specifies the logic which determines the channel to be used for delivery. When the Flow executes, the correct delivery channel is determined and the treatment specified for this channel is delivered.

Pega Customer Decision Hub

Treatment Specification

The following configurations options are available for specifying the treatments and channels for this shape:

- **Treatment Origin** – This setting determines how the channel for delivery is determined. The two options are:
  - **Strategy/Action** – In this scenario, the channel value determined by strategy execution (as part of a run) is used as the delivery channel. The user can alternately set the channel value (**.OfferData.pyChannel property**) directly in the Flow prior to this shape.
  - **Decision Rule** – In this scenario, the channel value is determined by executing a decision rule. The user must also specify the Decision Type (decision table or decision tree) and the name of the Decision Rule.
- **Treatments** – This group of settings is used to specify the treatments for delivery. The user can select the channels available for delivery by enabling the Outbound Email, Outbound SMS, and Passbook check boxes. The configuration options available for each channel are similar to those present on the channel-specific delivery shapes. Refer to the descriptions provided for each of these shapes for more details:
  - **Outbound Email** – Send Email shape
  - **Outbound SMS** – Send SMS shape
  - **Passbook** – Send Passbook shape
- **Default Treatment** – This setting determines the default channel value to use in case the channel returned by the Treatment Origin setting is either undefined or not currently supported. The values available for this setting are predicated by the channels that are enabled in the **Treatments** group.

Wait After Sending

Users can determine whether they would like to wait after the processing of this shape or proceed directly to the next step in the Flow. The wait options available are the same for all wait-based shapes and are described in the section on the Wait shape.

Delivery Options

Emails and SMS messages can be delivered in various ways - they can be sent by the system (**Online mode**) or they can be written to a data warehouse for processing by other systems (**File and DB modes**). Users can enable one or more of these modes at the same time.

The configuration options available are similar to the ones available for the Send Email and Send SMS shapes. These options are situated in the following areas of the Send Multiple Treatments shape:

- **Write to File/DB options** – In the **Output** section
- **Email Account options** – In the **Advanced** section
- **SMS Account options** – In the **Advanced** section

Configuring the delivery of passbook treatments with the Send Passbook shape

The Send Passbook shape enables the user to configure a Passbook Treatment for delivery.

Pega Customer Decision Hub

The following general configuration options are available for this shape:

Name	Purpose
Passbook Style	Style of the Passbook Treatment to send. Currently available options are **Coupon** and **Generic**.
Passbook Treatment	Name of the Passbook Treatment (pass) to send. Refer to the *Passbook Treatments* section in the *Treatments* chapter for details on creating and configuring Passbook Treatments.
Key Code	Marketing code used to identify the treatment. This value can be output in the sent email and/or the generated pass, and can be used to track the performance of different treatments. If a key code value is specified on the Passbook Treatment being referenced, it will be auto-populated into this field upon selecting the treatment.
Email Treatment	Name of the Email Treatment to use to deliver new passes and updates to existing passes. The pass is delivered as an attachment to the email message.

Delivery Options

New passes are delivered as attachments to emails. On iOS devices, the customer can open the attached pass and can add it to their Passbook.

The following delivery configuration options are available for this shape:

Name	Purpose
Email Subject	Subject of the email containing the new/updated pass. Refer to Configuring the delivery of an email treatment with the Send Email shape for more details.

On Update

Users have multiple options for updating passes that have previously been sent. When the same Passbook Treatment is used in multiple shapes in the same flow, the first delivery is treated as a new pass. Subsequent deliveries (shapes) are treated as updates to the existing pass. Furthermore, if the Passbook Treatment is configured as a global treatment (**“Keep updating the pass...”** option is enabled on the treatment’s Advanced tab), even the first passbook shape will result in an update scenario if the pass has already been sent to the customer previously (via a different action or campaign).

The following configuration options are available for updating passes:

Name	Purpose
Push Notification	Select this option to utilize the Apple Push Notification service (APNs) to signal the device that an update is available for the pass. The device will then make a web service call to get the latest pass.
Email Notification	Select this option to send the updated pass as an attachment to the email (specified in the Email Treatment field).

Note: If neither notification option (Push or Email) is enabled, the system will generate and store the updated pass internally. This updated pass will be available to pass holders (customers) when they attempt to manually update the pass.

Wait After Sending

Users can determine whether they would like to wait after sending a pass. The wait options available are the same for all wait-based shapes and are described in the section on the Wait shape.

Email Account

Users also have the option to specify the Outbound Email account to use for sending the message. These configuration options are available in the **Advanced** tab of the Send Passbook shape and are described in the Configuring the delivery of an email treatment with the Send Email shape sub-section of the **Send Email** shape.

Testing the Send Passbook shape

The Send Passbook shape provides an easy mechanism to test its outcome. Users can employ this test mechanism to test the online delivery of the email with the pass attachment. Refer to Configuring the delivery of an email treatment with the Send Email shape for details on configuration options for the test message.

The test email will be sent with the selected pass as an attachment. Refer to the *Reviewing the Test Pass* sub-section in the *Treatments* chapter for details on reviewing and adding the test pass.

- Uploading the required certificates for passbooks

Uploading the required certificates for passbooks

The Passbook landing page allows you to upload the Apple and Organization certificates (required by the Passbook functionality) to the system. To access the page, in App Studio, click **Settings Channels Passbook** .

Pega Customer Decision Hub

Uploading the Apple Certificate

Use the **Upload Apple Certificate** button to launch the dialog to upload the Apple Worldwide Developer Relations Certification Authority (WWDRCA) certificate to the system. This certificate can be downloaded from http://www.apple.com/certificateauthority.

Use the **Browse...** button in the dialog to locate the certificate and the **Upload** button to upload the certificate to the system.

Once the Apple certificate is loaded, the table displays the following information about the certificate:

Column Heading	Description
Certificate Name	Displays the certificate name. This will always be "Apple".
Valid From	Displays the date from which the uploaded certificate is valid.
Valid To	Displays the date to which the uploaded certificate is valid.
Actions	Provides the **Actions** menu with the **Open** item, which can be used to open the Certificate rule.

Note: Once a certificate has been uploaded, the upload button for that certificate type is disabled. To upload a new certificate, open the existing certificate rule (via the Open action item) and delete it.

Uploading the Organization Certificate

Use the **Upload Organization Certificate** button to launch the dialog to upload the Apple .p12 certificate for your organization or group. This certificate is associated with a pass type id and can be generated from the iOS Provisioning Portal, which is accessed via the iOS Dev Center.

Use the **Browse...** button in the dialog to locate the certificate. Also, enter values for the following fields and use the **Upload** button to upload the certificate to the system.

Field	Description
Certificate Password	Enter the password associated with the certificate. The system will validate this password and will not upload the certificate in case the password is invalid.
Organization Name	Optional. Enter the organization name to display on delivered passes. If you do not provide this value, the organization name associated with the certificate will be used.

Once the organization certificate is loaded, the table displays the following information about the certificate:

Column Heading	Description
Certificate Name	Displays the certificate name. This will always be "Default".
Valid From	Displays the date from which the uploaded certificate is valid.
Valid To	Displays the date to which the uploaded certificate is valid.
Pass Type ID	Pass type identifier associated with the uploaded certificate.
Actions	Provides the **Actions** menu with the **Open** item, which can be used to open the Certificate rule.

Note: Once a certificate has been uploaded, the upload button for that certificate type is disabled. To upload a new certificate, open the existing certificate rule (via the Open action item) and delete it.

Configuring the delivery of push notifications with the Push Notification shape

The Push Notification shape in the flow canvas enables the user to configure notification messages for delivery.

Pega Customer Decision Hub

The following table lists the various roles involved in the push notification process along with their responsibilities:

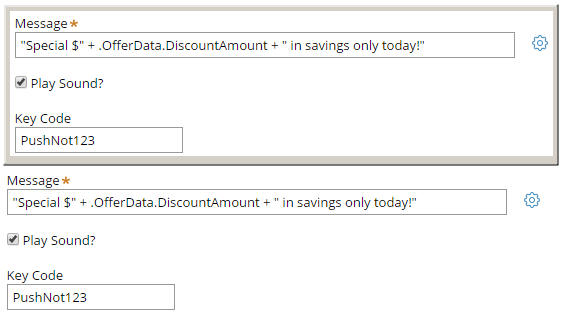
Role	Responsibility	Scope

Role	Responsibility	Scope
App Developer	Creates mobile app, which: Registers and un-registers app users for receiving push notifications from the application, via the provided REST services. See Push Notification User Registration Services for details. Receives and displays push notification messages to app users. Deploys app on app platform (for example, App Store, Google Play). Provides marketing administrator with key registration information for app.	Outside of Pega Customer Decision Hub
Administrator	Registers app for receiving push notification messages from Pega Customer Decision Hub.	In Pega Customer Decision Hub
App User	Downloads and installs app from app store.(Behind the scenes, app registers the user for receiving push notifications).	Outside of Pega Customer Decision Hub
Analyst	Creates actions and utilizes the Push Notification flow shape to send notification messages to registered app users.	In Pega Customer Decision Hub
System	Handles registrations of apps. Handles registrations/un-registrations of app users. Delivers notification messages to registered users for registered apps.	In Pega Customer Decision Hub

The following General Configuration options are available for the Push Notification shape:

Name	Purpose
Message	Notification message to push to the customer's registered device(s). This can be plain text or an expression and can include property references. If the Push notification is defined as a treatment for an Action, the message should be configured to reference .OfferData.PushMessage. Refer to Configuring push notification treatments for more details.
Play Sound?	Associate sound with the notification message. If this setting is enabled, the message payload references the default sound. When the message is received by the device, the device-specific default sound for notifications (if enabled) is played.
Key Code	Marketing code used to identify the treatment. This value can be passed to the device along with the notification message (as additional values).

A sample configuration for the general settings is shown below.



Application Configuration

Users also have the option to specify the application where the notification should be sent.

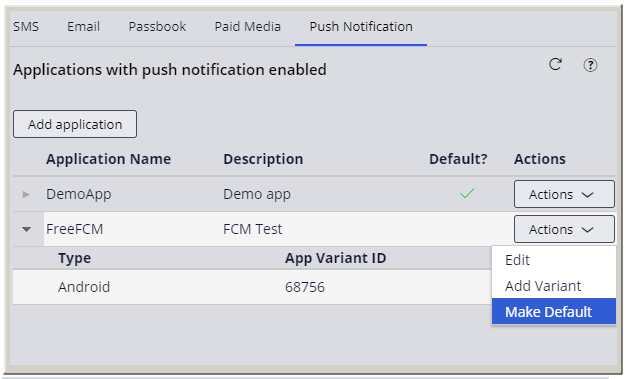
The following Application Configuration options are available in the **Application Configuration** sub-section:

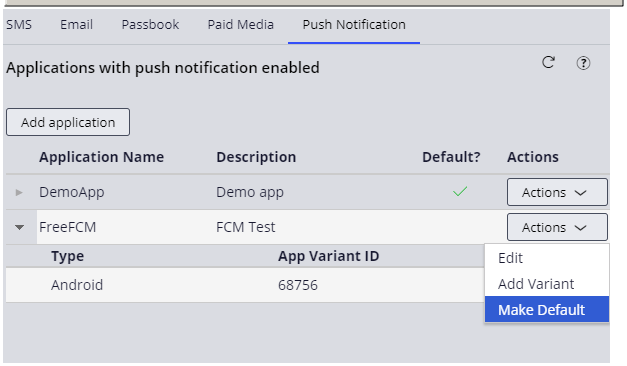
Name	Purpose
Default Application	Determines whether the notification message should be pushed for the default app.
Other Application	Determines whether the notification message should be pushed for the specified app. Select the app to notify from the drop-down of configured apps.

Setting the Default App

Administrators can mark one app as the default. This is beneficial in scenarios where there is only app or where a particular app is predominantly used. The default app is the (default) target for push notification messages in flows. Analysts can choose to override this by selecting a specific app

Use the **Make Default** action from the **Actions** menu to mark an application as the default.





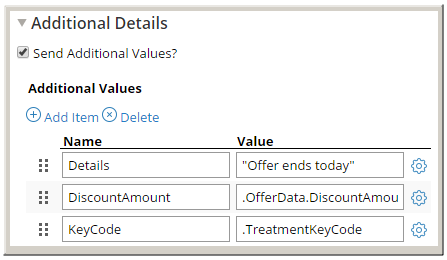
Wait After Sending

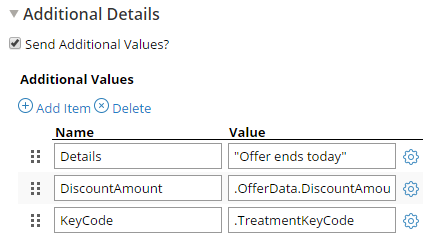
Users can determine whether they would like to wait after sending a Push Notification or proceed directly to the next step in the Flow. The wait options available are the same for all wait-based shapes and are described in the section on the Wait shape.

Send Additional Values

Users can choose to include other information as part of the notification message payload. This data can be used by the app developer. Different platforms place different restrictions on the size of the overall message payload. Users should be aware of this limitation when specifying additional values. A notification message that ends up being larger than the allowed payload size may not be delivered.

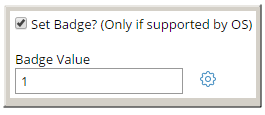
Enable the **Send Additional Values?** check box (in the **Additional Details** section of the properties panel) to pass additional values. Use the grid controls to add fields. For each field, specify a name and value. The name must be plain text, but the value can be an expression. The key code can also be passed as an additional field.

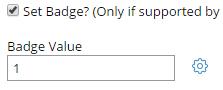




Set Badge

The badge value can be sent in the payload of the push notification and displayed if the user's device's operating system supports this function.





- Configuring application settings for push notifications
- Adding a Variant to an Existing App
- Testing an App Variant
- Push Notification User Registration Services

Configuring application settings for push notifications

Use the Push Notifications landing page to manage push notifications for Android and iOS apps. To access the page, in App Studio, click **Settings Channels Push Notifications**

Pega Customer Decision Hub

Using the Applications tab

The **Applications** tab on the Push Notifications landing page allows you to:

- Configure new apps for receiving push notifications
- View details of existing apps and their variants
- Perform actions on apps, such as adding new variants
- Perform actions on app variants, such as testing push notification delivery

The list of configured apps contains the following columns:

Column Heading	Description
Application Name	Displays the name of the app.
Description	Displays the description of the app.
Default?	Denotes (through a check mark) whether this is the default app for push notifications.
Actions	Provides the **Actions** menu with the following actions: • **Edit** - Perform this action to update the app's name or description. • **Add Variant** - Perform this action to add a new variant for this app. Refer to the Configuring a new app section below. • **Make Default** - Perform this action to make this app the default app for push notifications.

The expand pane for an app displays the configured variants for the app. The list of app variants contains the following columns:

Column Heading	Description
Type	Displays the platform of the app variant.
App Variant ID	Displays the unique identifier for the app variant.
Actions	Provides the **Actions** menu with the following actions: • **Edit** - Perform this action to update the details of the app variant. • **Test** - Perform this action to test the push notification functionality with this app variant. In the **Test Variant** dialog, provide a value for the Test Push Token and click **Test** to send a test push notification message to the app on the test device. The system displays the outcome of the test in the **Test Result** field. • **Remove** - Perform this action to remove this app variant. **Caution:** Removing the last variant of an app will also remove the app from the system.**Note:** This action is disabled for variants of the default app. It is also disabled on a variant once app users have been registered to receive notifications for it.

Configuring a new app

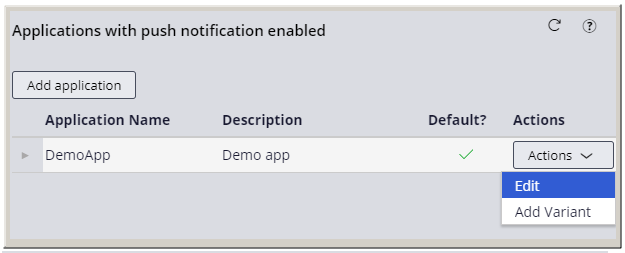
To launch the **Create a New Application** dialog for configuring a new app for push notifications, click **New**. Provide the following information and click **Submit** to create a new app:

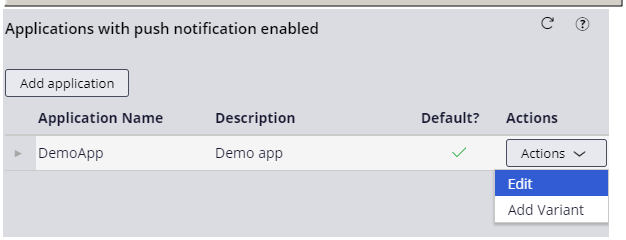
Field	Description
Name	Enter a unique name for the app.
Description	Enter a description for the app.
Platform	Select the platform on which this app is available. Choose from: • **Android** • **iOS**
Unique ID	Enter a unique identifier to associate with this app and platform. This id is used for validating push notification service requests.
Server Key	Android only. Enter the Google server key for this app.
App Certificate	iOS only. Use the **Browse** button to locate the Apple .p12 certificate for this app.
Certificate Name	iOS only. Enter a name for storing the app certificate in the system.
Certificate Password	iOS only. Enter the password associated with the certificate. The system will validate this password and will not upload the certificate in case the password is invalid.
Certificate Type	iOS only. Select the certificate type. Choose either **Production** or **Development**.
Monitor Apple's notification feedback service?	iOS only. Select the desired option: • **Yes, monitor and automatically manage errors** - Select this option to enable the system to monitor the feedback service for this app. When this option is selected, erroneous devices will automatically be un-registered from the system. • **No, another system will monitor feedback** - Select this option to monitor the feedback service through an external system/mechanism. In this case, the external mechanism should invoke the un-registration service to unregister push tokens that are no longer valid.

Adding a Variant to an Existing App

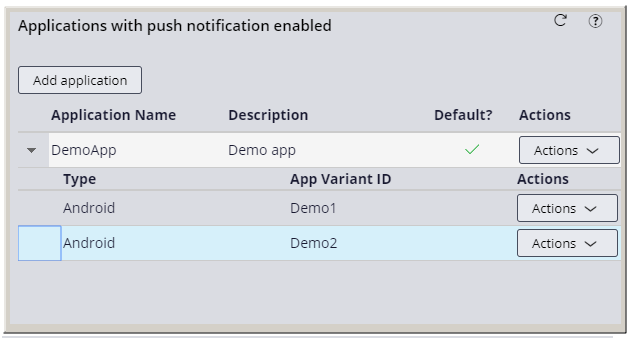
Typically, mobile apps are available on multiple platforms. This notion is represented in the system via App Variants. The Application entry in the Push Notifications landing page grid represents the conceptual app. The actual app instance on a particular platform is represented via an App Variant. Upon creating a new app, the associated variant is also created. Other variants can be added via the **Add Variant** action from the **Actions** menu for a particular application.

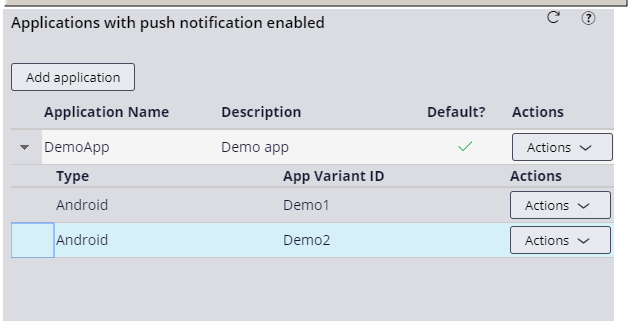
Pega Customer Decision Hub





The configured variants for an app can be viewed via the expand pane icon, located to the left of the app name.



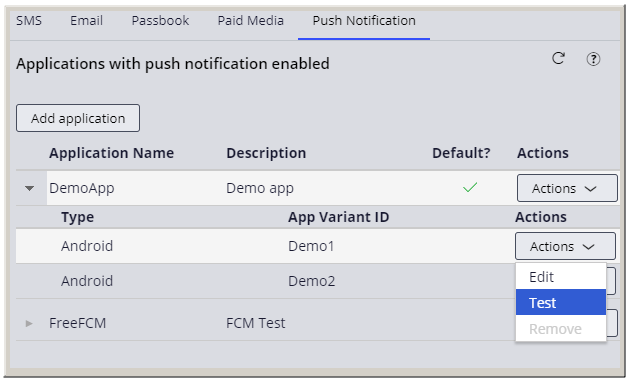


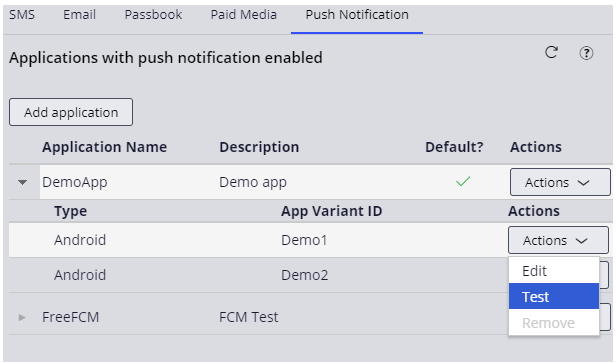
Testing an App Variant

After configuring an app variant, it is recommended that the user tests the configuration of the variant to ensure that push notification messages can be delivered for the app. Testing the variant validates both the variant's configuration and the mobile app itself.

Pega Customer Decision Hub

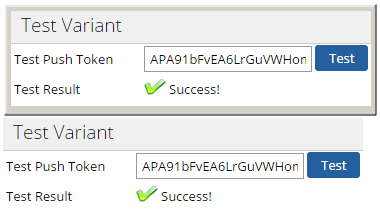
Use the **Test** action from the **Actions** menu for an app variant to launch the test variant dialog.



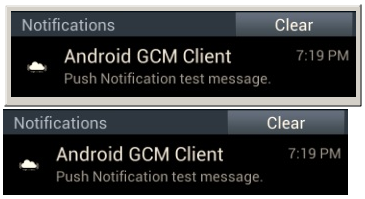


Provide a value for **Test Push Token** and click the **Test** button to send the test push notification message. A push token uniquely identifies the installation of an app on a particular device. The app developer should be able to provide you with the push token associated with the test device.

The system displays the status of the test message in the **Test Result** field.



The test message is delivered (as a push notification) to the app on the test device. The following image shows the test notification, as received for a sample app running on the Android platform:



Push Notification User Registration Services

Pega Customer Decision Hub provides two services for registering app users with the system. Push notifications (from actions) are only sent to registered customers. It is the responsibility of the app developer to code the app to invoke these services, at appropriate points in time. For example, when a customer installs the app, the app must register with the platform push notification service (e.g. APNs, FCM) and should then invoke the marketing registration service.

Pega Customer Decision Hub

- Registration Service
- Un-registration Service

Registration Service

The registration service is a REST service that supports the POST HTTP method. The URL for this service is:

Pega Customer Decision Hub

```
http://<host>:<port>/prweb/PRRestService/PushNotification/Registration/Register
```

For the request, this service expects a JSON object having the following structure:

```
{
   "AppVariantID":"<AppVariantID>",
   "PushToken":"<PushToken>",
   "UserID":"<UserID>"}
```

The tokens included in the above request are:

- *AppVariantID* - Unique ID for the app variant configured in the system. This is specified when the app variant is created in the system and is used as a means for validating genuine requests.
- *PushToken* - Token assigned by the platform messaging service that is used for pushing notification messages for the selected app on the selected device.
- *UserID* - ID associating the app user with a customer in the system. In the out-of-the-box implementation, the expected value for this field is the CustomerID property. Custom implementations can override the mechanism by which this field is associated to a customer in the system.

For the response, this service sets the appropriate HTTP status code. In case of errors, an appropriate message is also returned as part of the error stream.

Un-registration Service

The un-registration service is a REST service that supports the POST HTTP method. The URL for this service is:

Pega Customer Decision Hub

```
http://<host>:<port>/prweb/PRRestService/PushNotification/Registration/Unregister
```

For the request, this service expects a JSON object having the following structure:

```
{
   "AppVariantID":"<AppVariantID>",
   "PushToken":"<PushToken>"}
```

The tokens included in the above request are:

- *AppVariantID* - Unique ID for the app variant configured in the system. This is specified when the app variant is created in the system and is used as a means for validating genuine requests.

- *PushToken* - Token assigned by the platform messaging service that is used for pushing notification messages for the selected app on the selected device.

For the response, this service sets the appropriate HTTP status code. In case of errors, an appropriate message is also returned as part of the error stream.

Note: Un-registering a push token removes the push token registration from the system. The Data-PushNotification-Delivery-Registration.PreRegistrationDelete activity is provided as an extension point for custom implementations. This activity can be used to insert desired behavior before the registration entry is deleted. This activity can also be used to bypass the deletion process. For more details, refer to the usage guidelines specified in this activity.

Specifying the context for inbound applications with the Inbound Call Center shape

The Inbound shape provides a means to specify the content that should be returned when an inbound application, such as Next-Best-Action Advisor (NBAA), requests this action to display within their application.

Pega Customer Decision Hub

The following configuration options are available for this shape:

Name	Purpose
Section	Name of the Section rule to use. The content of this section is returned when an inbound application (such as NBAA) requests action content from the system.
Channel	Inbound channel for the Treatment (Default value for this shape is CallCenter).
Work Status	Status of the action after executing this step in the flow.

Note: There should be only one Inbound shape in a flow. Having multiple Inbound shapes in the same flow could cause non-deterministic behavior in resolving the content that should be returned to the inbound application.

The Inbound shape, by itself, simply serves as a determining point for the content to be returned for external requests. However, this shape can be (and typically is) combined with the InboundTicket ticketing mechanism to provide a branch of flow execution for processing an inbound request. Refer to Branching the action flow based on the channel for more details.

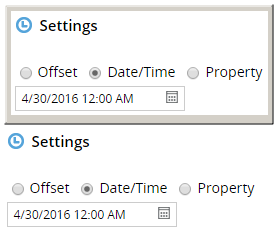
Pausing the flow execution with the Wait shape

The Wait shape enables the user to pause the flow execution for a specified duration. This shape can be used when an unconditional wait is required in the flow. When the specified wait duration has lapsed, flow execution automatically resumes with the next shape in the flow.

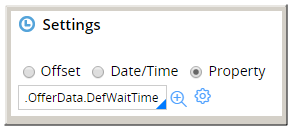
Pega Customer Decision Hub

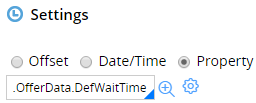
This shape supports the following modes for specifying the wait duration:

- **Offset** - Use this mode to set the wait time to a specified duration from when this shape is executed. Use the Days, Hours, Minutes, and Seconds fields to specify the offset. Optionally, enable the Business Days check box to apply the specified offset only using business days.
- **Date/Time** - Use this mode to set the wait expiration time to a specified future date and time. Specify the desired date and time.



- **Property** - Use this mode to set the wait expiration time using the value of a date time property or expression. Select the desired property or specify the desired expression.





Note: Ensure that the specified property (or expression) will have a valid date and time value during action runtime.

Users can also enable a wait in flow execution on wait-based delivery shapes. This includes the Send Email, Send SMS, Send Passbook, Send Generic, Send Multiple Treatments, and Push Notifications shapes. To enable the wait and configure the wait options (listed above), enable the **Wait after sending** check box on these shapes.

Specifying the status of an action with the Update Status shape

The Update Status shape is used to specify the status of the action as it works its way through various stages of the Flow.

Pega Customer Decision Hub

This status can be used to group actions for reporting and other aggregations. This status can also be used to determine the appropriate path to be taken down a subsequent branch in the Flow execution. The Update Status shape also enables the user to record Behavior and Response information for the action into Interaction History.

The following Configuration options are available for the Update Status shape:

Name	Purpose
Work Status	Status of the action. The out-of-the-box values for the action status include: • New • OfferSent • Resolved-OfferAccepted • Resolved-OfferExpired • Resolved-OfferRejected

Name	Purpose
Behavior	Denotes the behavior of the customer to the particular action. This information gets recorded in History. The out-of-the-box values for Behavior include: • Negative • Neutral • Positive
Response	Denotes the response of the customer to the particular action. This information gets recorded in History. The out-of-the-box values for Response include: • Accepted • NoResponse • Pending • Rejected

Launching background processes with the Hand Off shape

Use the Hand Off shape to launch other processes (in the background) from the Flow. This serves as a launching point for external processes which can be triggered at the appropriate time in the action execution. For example, use the Hand Off shape to initiate a Fulfillment process when a customer accepts an action.

Pega Customer Decision Hub
Note: Initial processing of the handed-off work object is sequential with the Flow execution until the new work object reaches a stopping point (such as an Assignment). At this point, Flow execution resumes and the handed-off work object lives and executes separately on its own.

The following Configuration options are available for the Hand Off shape:

Name	Purpose
Work Class Name	Name of the class which contains the process (Work Object flow) to be started.
Work Flow Name	Name of the process (flow) to be started.
Work Status(Status section)	Status of the action after the new process has been initiated.

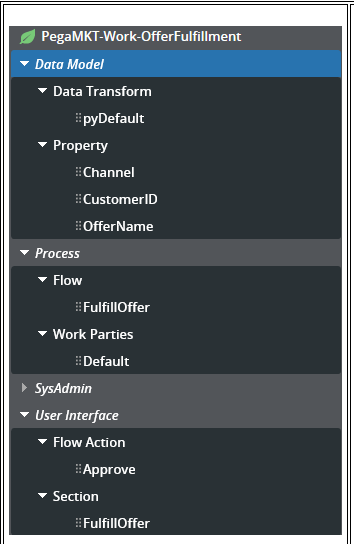
As part of the Hand Off shape, the entire contents of the action work object (including action details) are initially made available to the newly created work object. This information is stored in the top level OfferWorkPage clipboard page. This information is available whilst the new work object shares the execution with the action work object (see Note at the beginning of this section). If the new work object requires any of this data for its processing, it is recommended to copy relevant data from the OfferWorkPage onto appropriate locations on the new work object. This could be done via the use of the default Data Transform rule (pyDefault) or by other mechanisms in the work process flow before it enters a waiting point (such as an Assignment).

In a similar vein, a reference to the newly handed-off work object is stored on the original action work object. This is made available under the pyReferencedInsKeys construct. This information can be used to provide a reference code to the customer in an email or for any other cross-referencing purposes, as deemed fit by the user.

Note: In order for the system to successfully create and launch the handed-off work object, all necessary rules (such as Flow, Data Transform, Work Parties, etc.) should be properly configured for the desired work object.

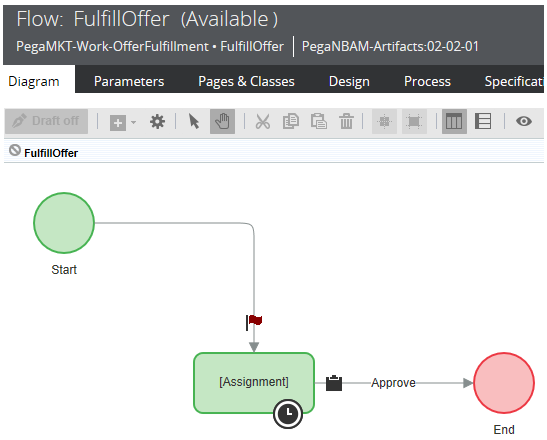
Hand Off Example

Some of the salient pieces involved in a handed-off work object are demonstrated in the following OfferFulfillment example:



- Data Transform:
  - pyDefault
- Properties:
  - Channel
  - CustomerID
  - OfferName
- Flow:
  - FulfillOffer
- Work Parties:
  - Default
- Flow Action:
  - Approve
- Section:
  - FulfillOffer

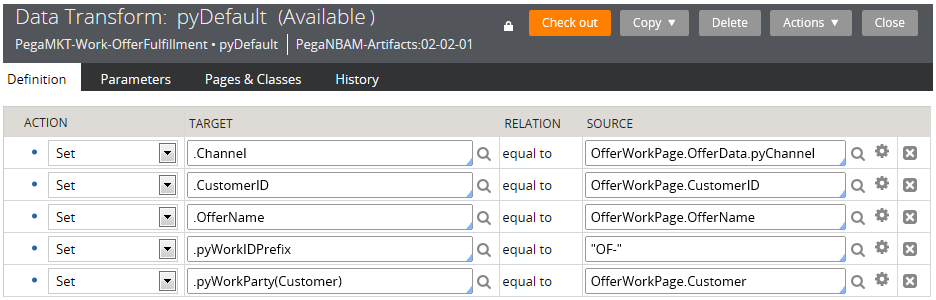
This work object involves a minimal process flow where the object is placed in a workbasket and is resolved upon approval. The FulfillOffer Flow rule is shown below:



The above flow is configured to create a new work object since we expect it to have its own lifecycle and want it to be actionable after it is stored in the workbasket. For simpler flows with straight through processing (no assignment shapes), the flow can be configured with the temporary object setting in the Process tab.

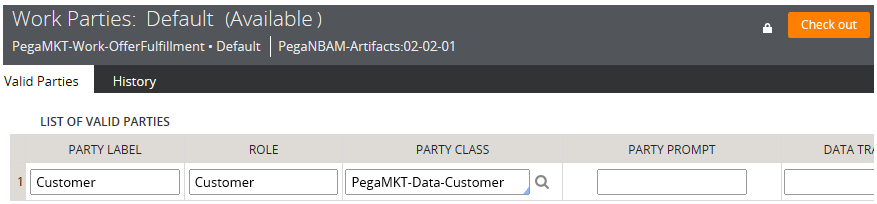
As mentioned previously, both customer and action data are available to this flow until it hits the first assignment. After this assignment, only data that has been saved onto the work object will be available.

The following (pyDefault) Data Transform demonstrates the copying of (both action and customer) data from the action page onto the work object, upon its initial creation:

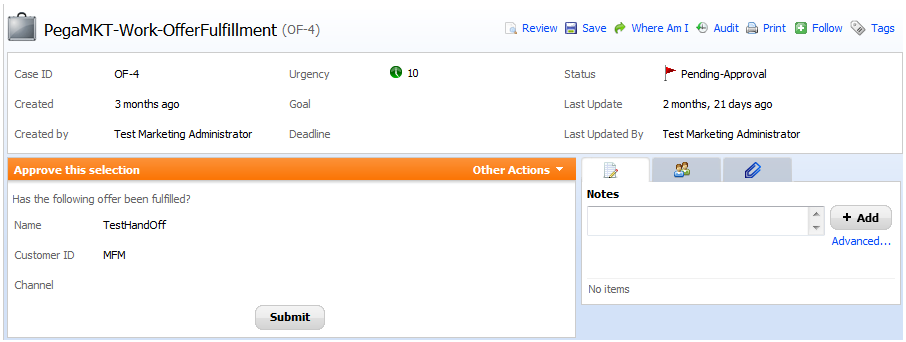


In the above Data Transform, the Customer object is added as the Customer work party on the new object.

A sample Work Parties rule for this work object is shown below:

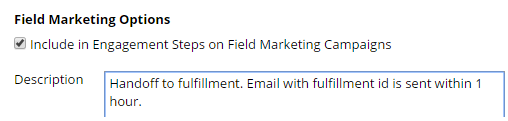


The sample Flow Action and Section use basic PRPC concepts to display some of the copied over data and present an action to resolve the work object. This is demonstrated by the following instance of a handed-off Offer Fulfillment work object:



Field Marketing Options

This set of options is available when the application includes Pega Field Marketing. Corporate marketers can enable the checkbox to include this Hand Off shape in the list of engagement steps that field marketers see in Field Marketing Campaigns. Additionally, they can also provide a meaningful description to aid the field marketer in understanding this shape's purpose.



Branching the action flow with the Decision shape

The decision shape enables the user to execute a variety of different decision constructs (such as decision trees, decision tables, etc.) to provide a basis for branching in the execution of the Flow.

Pega Customer Decision Hub

The following Decision types are currently supported:

- Boolean Expression
- Decision Table
- Decision Tree
- Fork
- Map Value
- Predictive Model
- Scorecard Model

More information on the various options for configuring each of the decision types (listed above) can be found in Pega Platform documentation.

Scheduling follow-up work with the Schedule Appointment shape

The Schedule Appointment smart shape enables the user to schedule and configure follow-up work to be performed at a particular point in the flow lifecycle.

Pega Customer Decision Hub

The user can utilize the “Instructions” field to specify any instructions to aid in processing the scheduled appointment.

Select Assignee

The user can choose from the following options:

- **Current Operator** – Routes the appointment to the operator that initiates this action (e.g. by using it in a Campaign and executing the Campaign).
- **Specify Operator** – Routes the appointment to the operator specified in the **Operator** field.
- **Workbasket** – Routes the appointment to the workbasket specified in the **Workbasket** field.

Schedule Options

The user can choose from the following options:

- **Relative Date** - Sets the appointment time relative to the time when this shape is processed in the action. The user must specify the relative number of days, the appointment time, and the time zone. The user can also enable the **Business Days?** checkbox to denote that the relative number of days should be applied using only business days.
- **Specify Date** – Sets the appointment time to the specified date and time.

Understanding the action flow connectors

The various shapes outlined in the prior sections can all be connected to one another. Each connector connects one shape to another. A shape can have multiple connectors emanating from it and/or terminating at it. The various connectors represent possible paths of Flow execution.

Pega Customer Decision Hub

Connectors can be categorized into the following types:

- Simple Connectors
- Decision Connectors
- Wait Connectors
- Response Received Connector

- Understanding simple connector
- Understanding Decision connectors
- Understanding Wait connectors
- Understanding Response Received connectors

Understanding simple connector

These connectors simply connect one shape to another. There is no logic path determination behind these connectors.

Pega Customer Decision Hub

The following shapes have simple connectors emanating from them:

- Inbound
- Update Status
- Hand Off
- Start

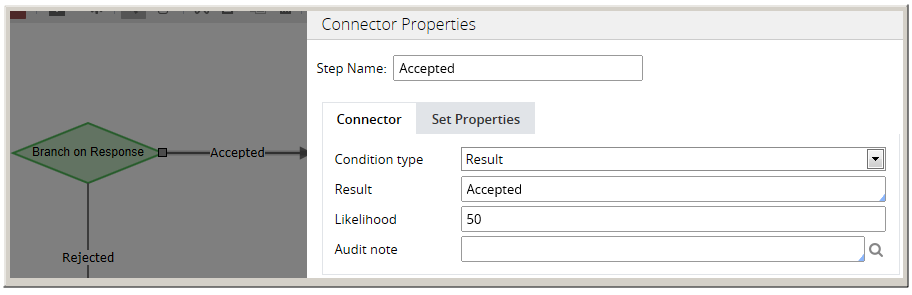
Understanding Decision connectors

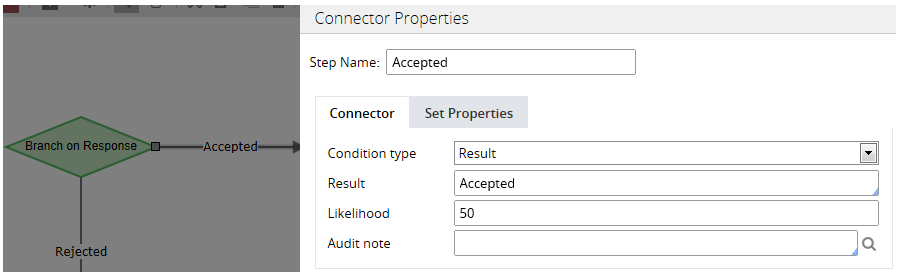
These connectors emanate from the Decision shape. Decision connectors can be sub-categorized based on the condition type – Result, Always, When, and Else.

Pega Customer Decision Hub

Result Condition Type

This condition type is used to represent the result of executing the Decision rule specified in the Decision shape. The system automatically analyzes the Decision rule and populates the Result drop-down with the possible set of outcomes from the decision.





Always Condition Type

This condition type represents a connector that is always taken.

When Condition Type

This condition type represents a connector that is taken when the expression/condition specified in the When input box evaluates to true.

Percentage Condition Type

Use the Percentage condition type to specify the percentage (relative amount) of time that the Flow should take this path during its execution.

Else Condition Type

The Else condition type serves as the catch-all connector and is taken when none of the other connectors originating from the Decision shape can be taken. It is a best practice to have an else path whenever the other execution paths aren't collectively exhaustive (e.g. with percentage and when condition types).

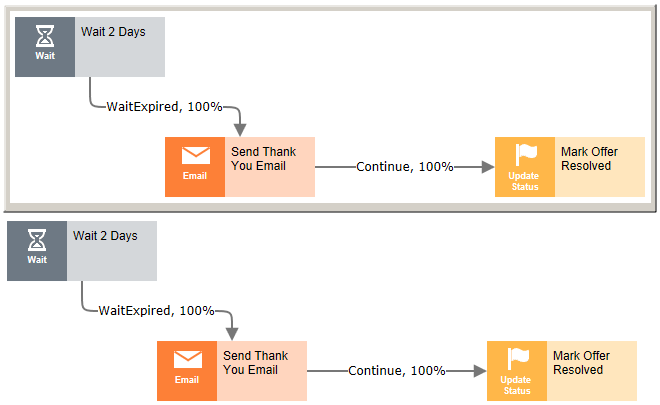
Understanding Wait connectors

The wait connector emanates from shapes that have a waiting mechanism tied to them (wait-based shapes). This includes the following shapes:

Pega Customer Decision Hub

- Send Email
- Send SMS
- Send Generic
- Send Passbook
- Send Multiple Treatments
- Push Notification
- Wait

There are two types of the Wait connector – Continue and Wait Expired. Only one Wait connector can originate from a wait-based shape. If waiting is enabled for the shape, the Wait Expired connector is used; otherwise, the Continue connector is used. The appropriate Wait connector (Continue vs. Wait Expired) is automatically generated when you connect a wait-based shape to another shape.



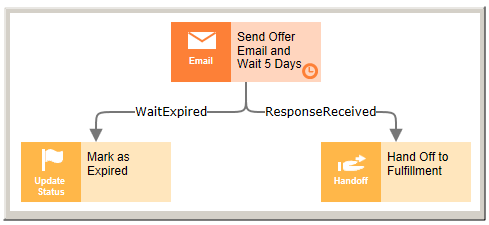
Note: Since the system manages the Wait connectors, these should not be changed manually. If you wish to update the type of Wait connector originating from a shape, toggle the Wait field on the shape. If you delete the Wait connector by accident, simply draw a new connector coming out of the shape and the appropriate Wait connector will be recreated for you.

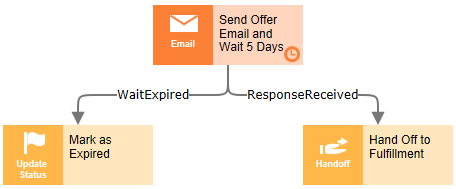
Understanding Response Received connectors

Use the Response Received connector to denote the Flow path to take when a response is received for a particular action. This is typically used as a connector originating from a Send Email or Send SMS shape after an action email or SMS is sent to a customer.

Pega Customer Decision Hub

This connector should be used in tandem with the Wait Expired connector, as shown below:



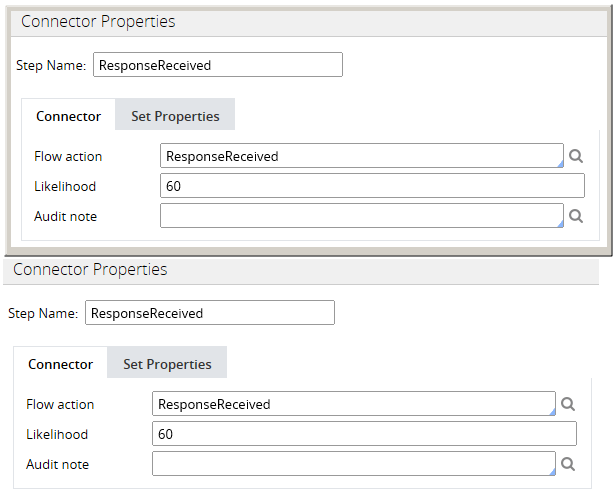


In the Flow depicted above, the System puts the action in a wait state for the duration specified (5 days). If no response is received within this wait duration, Flow execution will resume along the Wait Expired path. However, if a response is received before the wait duration has expired; the action will instantly get out of the wait state and go down the Response Received path.

Note: The ability of the system to go down the Response Received path is tied to the built-in response mechanism for Email and SMS treatments. Refer to Embedding a Response Link in the Treatments chapter for adding this mechanism to Email treatments. For SMS treatments, this mechanism is handled via the configuration of Inbound SMS accounts. For more details, refer to Configuring an Inbound SMS Account in the Email and SMS Accounts Configuration chapter.

To add a Response Received connector to a Send Email or Send SMS shape:

1. Enable Wait on the shape and configure the wait duration. Then, draw a connector from the shape to the backing shape for your Wait Expired action. The system will automatically create the Wait Expired connector for you.
2. Draw a second connector from the shape to the backing shape for the Response Received path. Open this connector's properties panel and select **ResponseReceived** as the value for the Flow Action smart prompt and provide a suitable Name for the connector (e.g. Response Received).



Branching the action flow based on the channel

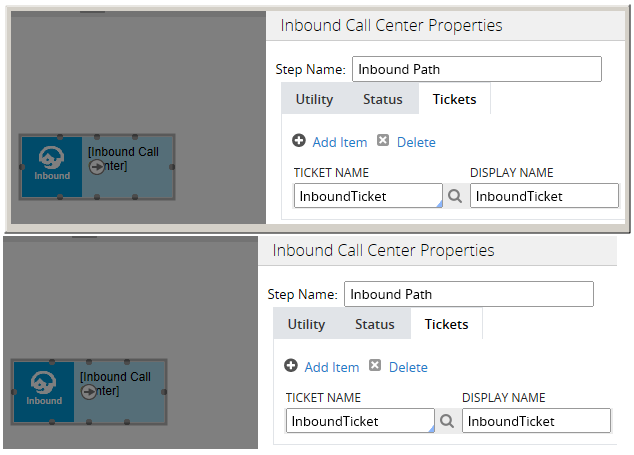
The system enables the use of the same action rule for both inbound and outbound channels. You can build distinct paths of flow execution based on the channel. This functionality facilitates the use of the system by inbound applications (such as NBAA) to process their customer offerings.

Pega Customer Decision Hub

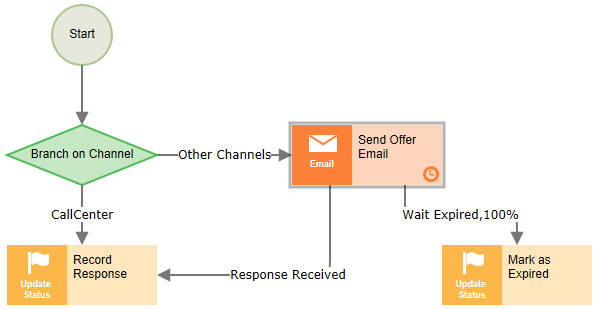
The system enables this inbound channel support functionality in the following two ways:

1. When an inbound request is received (for a specific action and customer), the system first determines whether the specified action is already in flight for the specified customer in the specified context. If so, the particular action is loaded and flow execution is forced down the Inbound path. This is done by the use of the InboundTicket construct. The InboundTicket can be associated with most flow shapes (with the exception of the Start shape and the Wait shape). This ticket denotes the place in the flow where processing should jump (switch to) when an inbound request is received for an in-flight action.

   The InboundTicket is enabled by specifying it as the value for the **Ticket Name** in the **Tickets** section (under **Advanced**) of the appropriate flow shape, as shown below.



   Note: To avoid ambiguity in flow execution, the InboundTicket should be specified on only one shape in a flow.
2. On the other hand, if the specified action is not currently in-flight for the specified customer and context, the system creates a new action object and starts executing the flow on it, in the context of the channel that was part of the inbound request. If there's a need to have separate processing paths for different channels (say Inbound vs. Outbound, or Call Center vs. ATM), this can be achieved by branching the flow on Channel at the appropriate place.



In the flow snippet above, the flow execution is initially branched based on the channel. If the channel on the input request is Call Center, the flow goes down the Record Response route – in this case, the customer is responding to the action in the Call Center, and there is no need to send them the action email. For all other channels, we send the action email and wait for either the customer to respond or the wait duration to expire.

Both the scenarios, outlined above, should be handled in an action if it needs to support requests on an inbound channel. Thus, such an action should have both:

- A shape designated as the resumption point for inbound requests by having the InboundTicket set on it
- A branch in flow execution for channel-specific paths (as needed)

Note: When integrating with an inbound application (such as NBAA), it is typical to use the Inbound shape to hold the InboundTicket.

Grouping offerings into action bundles

In typical marketing scenarios, it is often desirable to group offerings into bundles. There may be varied reasons for doing so, such as incentives, logistics, etc. To address this marketing need, the system provides support for grouping actions into bundles.

Pega Customer Decision Hub

- Action Roles
- Action Bundle Types
- Marking an action as being part of a bundle
- Designing Flows for Bundles
- Configuring email treatments for actions which are grouped together as a bundle

Action Roles

Actions in an action bundle can be categorized into the following two roles:

Pega Customer Decision Hub

- Parent action - Each bundle must have one parent action. Typically, the flow for the parent action will propagate the offering of its member actions. For example, the parent flow will send an email with details about its member actions and action links to respond to the actions.
- Member action - Each bundle can have multiple member actions.

Action Bundle Types

The system supports three modes of action bundling. These modes are henceforth referred to as bundle types.

Pega Customer Decision Hub

- Fixed Bundle - This bundle type should be used when customers must purchase the bundle in its entirety (all members). In this mode, the parent flow is the only flow that is processed. All of the recorded response (Interaction History, Adaptive Decision Manager, and Visual Business Director responses) are recorded for the parent bundle and all of its members. Any response will progress the parent flow.
- Flexible Bundle - This bundle type should be used when customers can purchase the entire bundle or can pick individual members to purchase. In this mode, the parent flow is used as a container for distributing communication about the collection of offerings. A response to an individual offering will only start that member's flow. Any recorded response in the member flow will record information for the member action and the parent action. Responding directly to the parent bundle action will trigger all member actions that haven't been started.
- Grouped actions - This bundle type should be used when the user has a selection of actions they wish to communicate together to the customer. The individual actions themselves may not have any correlation to each other. In this mode, the parent flow is used as a container for distributing communication about the collection of offerings. A response to an individual offering will only start that member's flow. However, any recorded response in the member flow will only record information for the member action.

The differences in the behaviors of the three bundle types are summarized in the table below:

Scenario	Fixed Bundle	Flexible Bundle	Grouped Actions
Action initiation	Same behavior for all three bundle types: • Start the parent flow • Record initial IH entry for parent and all members		
Respond to Email link for bundle (all members)	- Resume the parent flow	- Start all member flows that haven't been started - Parent flow does not resume	- Resume the parent flow
Respond to Email link for individual action	- Resume the parent flow	- Start or resume the member flow	- Start or resume the member flow
Update Status shape in parent action	- Record IH response for parent and all members	- Record IH response for parent and member actions that haven't been started	- Record IH response for parent and member actions that haven't been started
Update Status shape in member action	N/A (since member action is never started)	- Record IH response for member and its parent	- Record IH response for member only
Parent action completion	- Complete parent and all member actions	- Complete parent and member actions that haven't been started	- Complete parent and member actions that haven't been started
Member action completion	N/A (since member action is never started)	- Complete member action only	- Complete member action only

Marking an action as being part of a bundle

There are two ways to specify the action bundle settings for an action.

Pega Customer Decision Hub

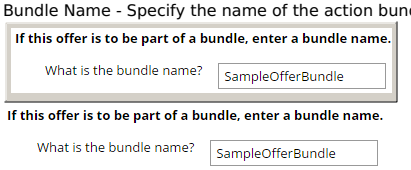
- Using the Details Tab on the Action Rule Form
- Specifying Bundle Characteristics in the Strategy

Using the Details Tab on the Action Rule Form

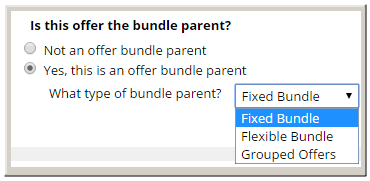
The **Details** tab on the action rule form contains the **Bundle Attributes** section which can be used for specifying the following bundle settings.

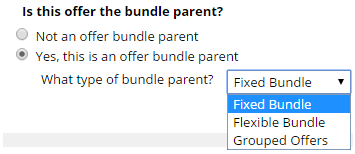
Pega Customer Decision Hub

- Bundle Name - Specify the name of the action bundle. This name should be the same for all actions (both parent and members) in the bundle.



- Action Role - Specify the role this action will play in the bundle. Select **Yes** if this action is a parent action, otherwise select **No**. Also, select the bundle type for the parent action.



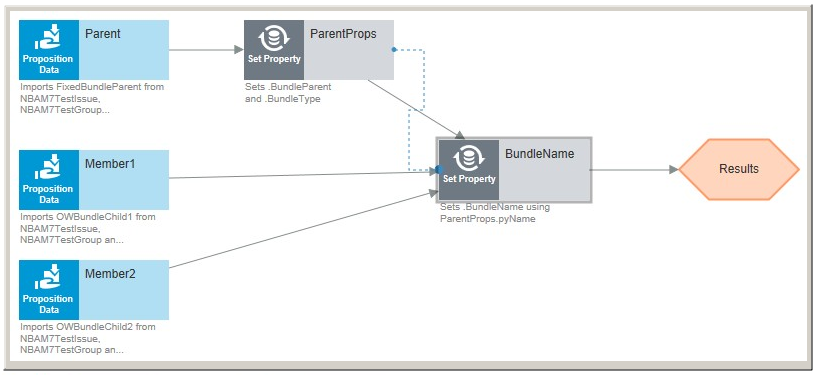


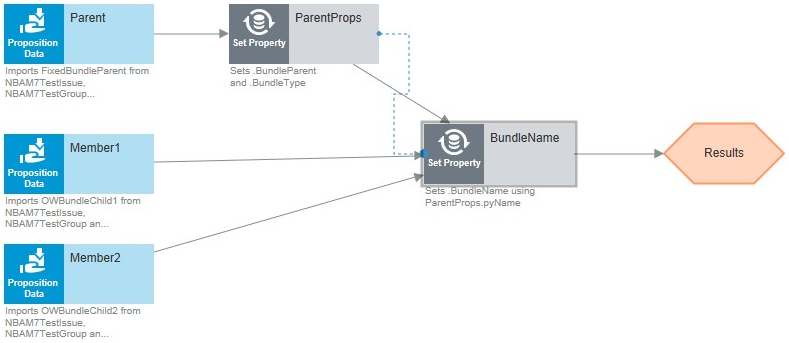
Specifying Bundle Characteristics in the Strategy

Bundle characteristics for actions can also be directly specified in Strategies. Strategies can also be used to override the bundle settings specified on the action. The settings on the Details tab can be considered as defaults, which can be overridden in Strategies (if desired).

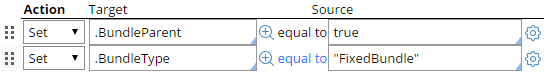
Pega Customer Decision Hub

The following example illustrates setting bundle characteristics using a Strategy rule.

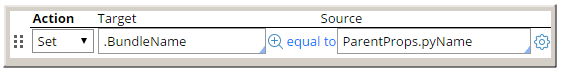


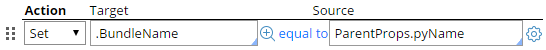


For the parent action, the bundle type is specified (FixedBundle, FlexibleBundle, or GroupedOffers) and the bundle parent flag is set to true.



For all actions (parent and two members), the bundle name is specified. In the above example, the parent action's name is used as the bundle name.





Designing Flows for Bundles

When designing flows for bundled items, the user must take into consideration the role of the action. For the parent action, the flow can also be tailored differently based on the bundle type.

Pega Customer Decision Hub

- Configuring the bundle parent action
- Configuring the bundle member actions

Configuring the bundle parent action

The parent action flow is responsible for sending out the email containing details of the bundle and its members. This email can also contain links to respond to the bundle and/or individual actions. This initial responsibility is the same across all three bundle types. Once the email has been sent, the behavior of the parent action flow typically varies slightly based on the bundle type.

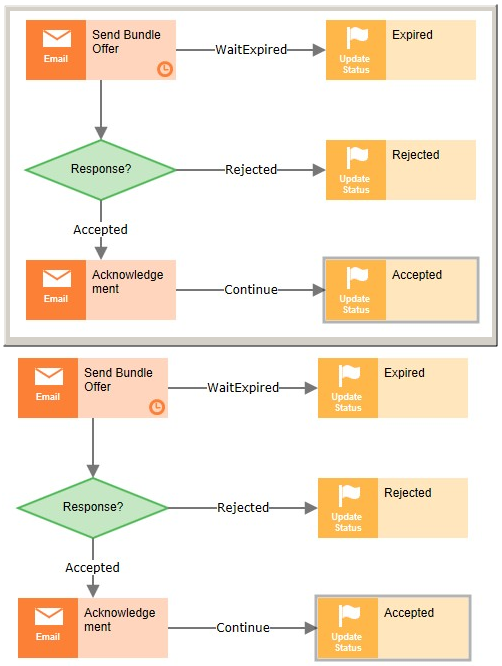
Pega Customer Decision Hub

- Fixed Bundle
- Flexible Bundle
- Grouped Actions

Fixed Bundle

In a fixed bundle, the parent flow is responsible for handling responses received for the bundle, as shown in the flow snippet below.

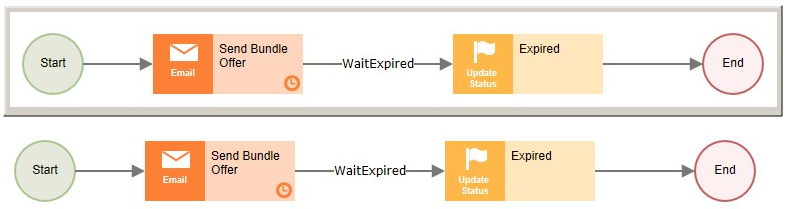
Pega Customer Decision Hub



Flexible Bundle

In a flexible bundle, the parent flow is only responsible for the initial communication of the action bundle. Responses received for the bundle will not progress the parent flow. A simple parent flow is shown below.

Pega Customer Decision Hub



Grouped Actions

In a grouped bundle, the parent flow is responsible for the initial communication. Typically, bundle-level response links are not provided in this mode. If a bundle-level link is made available, the parent flow will be resumed. This could be used to trigger a follow-up email with more information and/or update Interaction History records for the parent and all member actions that haven't been started.

Pega Customer Decision Hub

Configuring the bundle member actions

The flow for a bundle member action differs slightly from typical action flows since the initial communication of the action has already been made by the bundle parent action. The flow also differs based on the bundle type.

Pega Customer Decision Hub

- Fixed Bundle
- Flexible Bundle and Grouped Actions

Fixed Bundle

In a fixed bundle, the member flow is never started. Thus, the member flow can simply be an empty flow.

Pega Customer Decision Hub



Flexible Bundle and Grouped Actions

In both these modes, the member flow is started (or progressed) when a response is received. With the exception of the initial communication, these flows can be designed just like standard action flows.

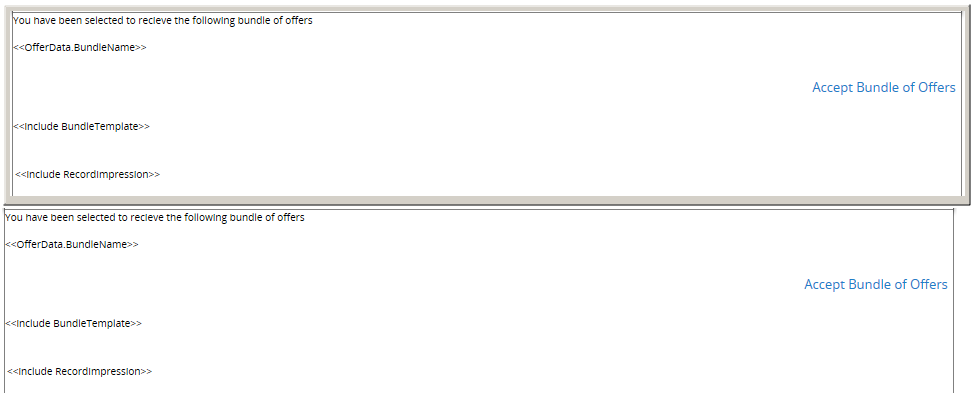
Pega Customer Decision Hub

Configuring email treatments for actions which are grouped together as a bundle

When designing Email Treatments for bundles, the user needs to consider the type of the bundle. A typical Email Treatment for an action bundle includes an overall description of the bundle and details for each of its members. Additionally, based on the bundle type, response links can be provided at the bundle and/or the member level.

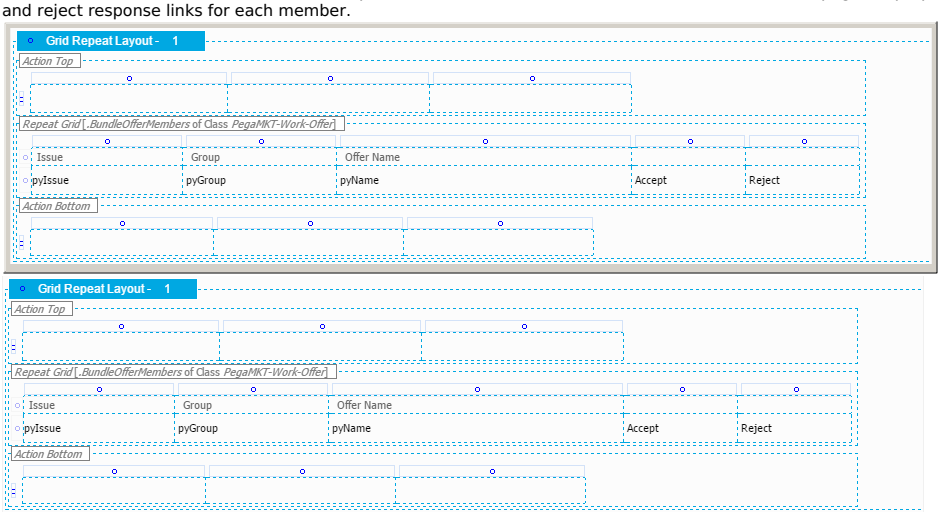
Pega Customer Decision Hub

The top level BundleTemplate Email Treatment is provided as an out-of-the-box sample.



This treatment includes the following salient pieces of a bundle Email Treatment:

- Bundle name - References the .OfferData.BundleName property
- Response link for bundle – Added using the Insert Button functionality
- Impression capture - References the RecordImpression section. With bundles, it is typical to include the impression capture section at the parent level.
- Member details - References the BundleTemplate section. This section uses the BundleOfferMembers page list property to display the member actions in a grid with accept and reject response links for each member.



The BundleOfferMembers property is populated at runtime to contain details of the various bundle members. This can be used in dynamic containers such as the repeating grid. Users can also make static references to member actions in this page list. For example, to display the name of the first action in the bundle, reference the .BundleOfferMembers(1).OfferData.pyName property.

Another common usage pattern is to update the grid in the BundleTemplate (or equivalent) section at the top level (PegaMKT-Work-Offer) to include a sub-section for displaying relevant action information, say MemberOfferDetails. Users can then create a top-level version of this section to serve as a base for all member actions. To customize the details for a particular issue or group, the user simply needs to override the details section in the appropriate issue or group class.

Configuring an action flow to write information to an external system

Create Output Template rules to define the content to write to an external source as part of making an action. Pega Customer Decision Hub supports writing to both a file and a

database table.

Pega Customer Decision Hub

You can use data exports out of Pega Customer Decision Hub to support and extend your customer engagement efforts. For example, as part of your customer retention efforts, you may prepare an engagement message for customers whose contracts are due to expire, encouraging them to renew their contracts. After the message is emailed to the customers, you may want to follow up by sending them the required renewal forms by direct mail. To automatize this process, you can configure Pega Customer Decision Hub to automatically write the customers' names and mailing addresses to a file.

Understanding the process of using an output template

The process of writing information to an output template consists of the following stages:

1. *Configuration* - In this stage, an operator with access to the **Treatments** landing page creates an Output Template rule, configures the output contents, and specifies the finalization schedule. For more information, see the following topics:
   - Creating file templates
   - Creating database templates
   - Configuring the contents of a file or database template
   - Configuring the finalization schedule for a file or database template
2. *Runtime* - In this stage, the records are staged for writing to the target file or database table. The runtime stage for an output template starts when an action flow reaches the flow shape that references the template.
3. *Finalization* - In this stage, the target file or database table is populated with the staged records. The finalization stage can be triggered manually, automatically at the end of each run, or according to a predefined schedule. For more information, see Configuring the finalization schedule for a file or database template.

You can view the list of output templates, their status, and other details on the File and Database landing pages. For more information, see File and Database landing page details.

- Creating file templates
- Creating database templates
- Configuring the contents of a file or database template
- Configuring the finalization schedule for a file or database template
- File and Database landing page details

Creating file templates

To create a new template for file exports, complete the following steps:

Pega Customer Decision Hub

1. In the Pega Customer Decision Hub portal, click **Content Treatments** .
2. On the **Treatments** landing page, click **Create File** .
3. Enter a description of the template.
4. Select the issue and group for storing the output template rule.
   To create the rule at the top level of your next-best-action business structure, leave the **Issue** and **Group** fields empty. The level at which you create the template determines the attributes (action details) that are available for use within the template.
5. Click **Create and open**.
6. Provide the following information:
   a. In the **Repository name** field, enter the name of the repository where you want to store the file.
   b. In the **File path** field, enter the path on the repository where you want to store the output file, and the name of the file where you want to store the output content. The format for the file path and name is /folder/filename. The file name that you provide is automatically appended with a unique identifier when the file is generated.
   For example: actions/output/mailing-list-kjssbwuamqoo91335643959855560.csv.
   Note: If you configure multiple output templates,each file should have a unique name.
7. Specify whether you want to **finalize the template at the end of each outbound run**.
   By finalizing a template at the end of each run, you ensure that all data in the file is the result of a single run. That is, the final output includes all data related to the run that triggered the finalization. If want the data from multiple campaigns to be written to the same file, leave this option unselected.
8. If you chose to finalize the template at the end of each run, select **Include run ID in the file name** if you want the file name to include the run ID for easier identification.
   For example: actions/output/mailing-list_PR1-kjssbwuamqoo91335643959855560.csv.
9. Specify whether you want to include the date and time in the file name when you finalize the template.
   When the template is finalized, the name of the output file includes the date and time configured in the **MKTFileNameDateMask** dynamic system setting. By default, the format is YYYY-MM-DD-hhmmss
   For example: actions/output/mailing-list_PR1_2020-02-03-111609-kjssbwuamqoo91335643959855560.csv.
   Tip: As a best practice, if you chose to finalize the template at the end of each run, configure the file name to include the run ID, the data and time of the template finalization, or both.
10. In the **Delimiter type** field, select the delimiter for the output CSV file.
    Use a delimiter which is recognized by the target repository for the file. Common delimiters include commas (,) and semicolons (;).

Creating database templates

To create a new template for database exports, complete the following steps:

Pega Customer Decision Hub

1. In App Studio, click **Content Treatments** .
2. On the **Treatments** landing page, click **Create Database** .
3. Enter a description of the template.
4. Select the issue and group for storing the database template rule.
   To create the rule at the top level of your next-best-action business structure, leave the **Issue** and **Group** fields empty. The level at which you create the template determines the attributes (action details) that are available for use within the template.
5. Click **Create and open**.
6. Provide the following information:
   a. In the **Database** field, specify the name of the Database rule (*Data-Admin-DB-Name*) that points to the database where the system writes output records for the output template.
   b. In the **Table** field, enter the name of the table where you want to write the output content for this template.
7. Specify whether you want to finalize the template at the end of each outbound run.
   By finalizing a template at the end of each outbound run, you ensure that all actions from the run are written to the database at the same time. That is, the final output includes all data related to the run that triggered the finalization. .**Note:** If you choose to finalize the template at the end of the run, all data from the run is appended to the existing data in the table.
8. If you did not choose to finalize the template at the end of each run, specify how you want to write the information to the destination database:
   - To replace the data in the destination table with new data, select the **Overwrite existing data** check box.
   - To add new data to the destination table, select the **Append to existing data** check box.

Configuring the contents of a file or database template

Configure the output content, including both data attributes and static text, for this template.

Pega Customer Decision Hub

1. On the **Fields** tab of a template, click **Add** to add a new field to the template contents.
2. Enter a name for the field.
   Note: The system uses this value as the name for the database table column or file column header. Do not use database reserved words such as Group for the names, as this will cause errors during the template finalization.
3. Select whether the field is a **Data Field** or **Text**.
4. In the **Content** field, select the data field to output, or type in the text.

5. For file templates, if the field that you added is a data field, select the format to apply to the field. **For example:** Depending on the data field that you selected, formats can include date formatting, currency formatting, decimal separators, and others.

Configuring the finalization schedule for a file or database template

Configure the finalization schedule, notifications, and post-finalization activity for the output template.

Pega Customer Decision Hub

Finalizing a template writes any pending data to the output file or table. For file templates, finalization converts the staging output template *<container name>*_LOCK table) to its proper version (without lock in its name). For database templates, this happens only if you select the the **Overwrite existing data** writing option.

Pega Customer Decision Hub uses the standard queue processor for template finalization. If you want to use a dedicated queue processor, add a new dynamic system setting called CDHDedicatedFileDBQueueProcessor, and set its value to true.

Tip: If you configure the output template to be finalized at the end of each run, you do not need to enable scheduled finalization. As a best practice, enable scheduled finalization for templates that capture data for actions triggered by mechanisms other than outbound runs, such as real-time events or containers.

1. In the **Schedule** section, select the **Enable schedule** check box to configure when and how frequently to finalize this output template.
   Tip: As a best practice, do not schedule the template to be finalized more often than every 30 minutes. Finalizing the template too frequently can result in unexpected behavior.
2. **Optional:** In the **Delivery options** section, configure the notification that you want to send when the template is finalized.
3. **Optional:** In the **Advanced** section, configure a post-finalization activity for this output template:
   a. In the **Task name** field, enter a name for the task.
   b. In the **Task description** field, enter a description for the task.
   c. In the **Activity class** field, select the **Applies To** class of the post-finalization activity from the list.
      Note: The post-finalization activity must reside in the *PegaMKT-Work-Offer* class or in one of its descendant classes.
   d. In the **Activity name**, select the post-finalization activity (*Rule-Obj-Activity*) from the list.

What to do next: After you specify the contents of an output template, you can configure the following outbound shapes in an action flow to write to the template:

- Send Email
- Send SMS
- Send Generic
- Sent Multiple Treatments

File and Database landing page details

The File and Database landing pages display detailed lists of File and Database rules currently defined in your system.

Pega Customer Decision Hub

You can access the landing pages in the Pega Customer Decision Hub portal by clicking **Content Treatments File** or **Database**.

The list of file or database templates contains the following columns:

Column Heading	Description
Status	Displays the current status for the output template. The possible status values are: • **Online** - The output template can be used to write records and is available for finalization. • **Finalized** - Finalization is currently in progress for this output template. • **Failed** - An error occurred while writing records or finalizing the output template. • **Unknown** - The status for this output template cannot be retrieved. Usually, this is because the Decision Data Store service is unavailable.
Template name	Displays the name of the output template.
Description	Displays a short description of the template.
Issue	Displays the Issue to which the output template belongs.
Group	Displays the Group to which the output template belongs.
Last updated	Displays the date and time when records were last written to the output template.
Updated by	Displays the name of the operator who last updated the template.
Actions	Click the **Actions** icon to select an action to do on this template. For templates in the Online status, you can select the following actions: • **Finalize** - Triggers the finalization of the output template to the destination file or database table. • **Preview** - This action is available only for Database templates. It generates a CSV file containing a subset of the records in the database table. The number of records returned in the preview file is configurable by the application administrator.

To view more details about a template, click the corresponding row in the table. This displays the following additional details:

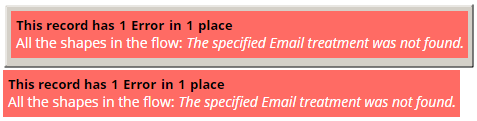
Field	Description
Repository name	Available for file templates. The name of the repository to which the file template is configured to write.
File path	Available for file templates. The file path to which the template writes. The file path is relative to the configured repository.
Database	Available for database templates. The name of the database rule to which the database template is configured to write.
Table	Available for database templates. The name of the database table to which the database template is configured to write.
Estimated staged records	Available for both types of templates. The estimated number of records which are currently staged and awaiting finalization for this output template.
Failed records	Available for both types of templates. The number of failed records associated with this output template. Only shown when the template status is Failed.

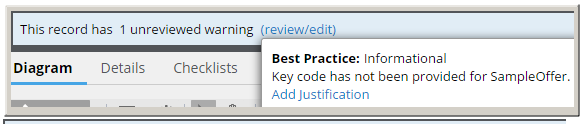
Validating action flow configuration

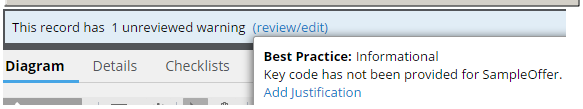
When the action is saved, the system validates multiple aspects of the action and presents the user with any violations that are found. These violations are reported as warnings when the Flow is in Draft mode and the action can still be saved. When Draft mode is disabled, these violations are reported as errors and must be corrected in order for the action to be saved.

Pega Customer Decision Hub

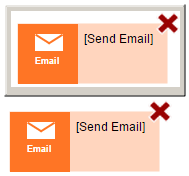
These errors and warnings are displayed underneath the action header.







If a shape has any errors associated to it, an error icon is also displayed on the shape. The tooltip on the error icon displays the associated error message.



Some examples of these violations include the following:

- Missing required fields
- Invalid rule references – e.g. Treatment, Template, Decision, Email Account, etc.
- Missing key codes (warning only)

Note: Being able to successfully save an action with Draft mode disabled is an excellent gauge for ensuring successful flow execution. It is strongly recommended that any actions that are live (can be potentially sent to customers) should be validated by saving them with Draft mode disabled.

Testing the action flow

After you define the action flow, test it against seed customers to ensure that it works as expected. The action is tested against a seed list, that is, a list of dummy customers created for the purpose of testing.

Pega Customer Decision Hub

1. Click the **Test** tab of an action where you want to test the flow.
2. In the Select Recipients section, select a seed list for the action test.

   The test will be executed for each dummy customer in the seed list. All the data associated with the dummy will be available as customer data in the action test execution and will be used in the various flow shapes.

3. In the **Select Starting Point** section, select the flow shape from which the action test should start.
   You can start the test from any of the flow shapes, except the End shape.
4. In the **Select a Channel and Direction** section, specify the channel and direction of communication which should be tested.
5. Click **Test action**. **Result:** The test starts and any treatments associated with the action are sent to recipients. You can test the responses by clicking the links in the treatments. Results from the tests are not saved in Interaction History, and thus do not impact the performance of AI models which govern Next-Best-Action.
6. To monitor the progress of the test, see the **Track Results** section.
   Any errors in the test are reported at the top of the **Test** tab. Correct them before you launch the test again.

Importing and exporting actions

In addition to creating actions manually, you can also import actions created outside of Pega Customer Decision Hub by uploading a .csv file with the action data.

Pega Customer Decision Hub

1. Prepare a valid .csv file with the list of actions that you want to import. The .csv file must contain the name and status (*.pyIsPropositionActive*) value of the action. Possible values for the *.pyIsPropositionActive* are Always, Never, and the expiration date of the action.
   Note: To easily create a valid .csv file, export a template for the Issue and Group where you want to create the actions. For more information, see Exporting Actions.
2. In the Actions landing page, click **Create Create actions from file** .
3. Click **Choose File** and select a valid .csv file.
   The .csv file must contain the name and status (.pyIsPropositionActive) value of the action. Possible values for the .pyIsPropositionActive are Always, Never, and the expiration date of the action.
4. Select the Issue and Group for the actions in the import file.
5. In the **Template Type** section, select whether Pega Customer Decision Hub should apply an flow template to the imported actions.
   - **None** - Actions are imported with a default flow. The default flow consists of a Start and an End shape.
   - **Reference flow from another action** - For each action in the .csv file, you can specify an existing action to be used as template for the imported action. The imported action is created with the same flow as the template action. If the template action is not found, the new action is imported with a default flow. The default flow consists of a Start and an End shape.
   - **Copy flow from another action** - If your .csv file does not specify individual templates for each action, you can select one action to use as template for all actions in the .csv file. The imported actions are created with the same flow as the template action.
6. **Optional:** If you want to completely overwrite existing actions with actions from the .csv file, select **Delete existing actions that are not in the file** .
   This will delete any action which is not included in the .csv file you are importing.
7. Click **Import**. **Result:** The pop-up window displays the import summary, including the number of actions imported, the number of actions deleted, and the number of errors. If there are any errors in the .csv file, download the error report and resolve each error before you continue.

Exporting Actions

You can export actions created in Pega Customer Decision Hub to a .csv file with the action data. You can also use the Export function to quickly create a valid import template for actions created outside of Pega Customer Decision Hub.

For more information, see Importing and exporting actions.

Pega Customer Decision Hub

1. In the **Issue / Group** filter, select a Business Issue and Group.
2. Click **Apply**.
3. Click **View**. The **Export CSV file** link in the top right corner of the action list is now active.

4. Click **Export CSV file**.

Importing and exporting actions

In addition to creating actions manually, you can also import actions created outside of Pega Customer Decision Hub by uploading a .csv file with the action data.

Pega Customer Decision Hub

1. Prepare a valid .csv file with the list of actions that you want to import. The .csv file must contain the name and status (*.pyIsPropositionActive*) value of the action. Possible values for the *.pyIsPropositionActive* are Always, Never, and the expiration date of the action.
   Note: To easily create a valid .csv file, export a template for the Issue and Group where you want to create the actions. For more information, see Exporting Actions.
2. In the Actions landing page, click **Create Create actions from file** .
3. Click **Choose File** and select a valid .csv file.
   The .csv file must contain the name and status (.pyIsPropositionActive) value of the action. Possible values for the .pyIsPropositionActive are Always, Never, and the expiration date of the action.
4. Select the Issue and Group for the actions in the import file.
5. In the **Template Type** section, select whether Pega Customer Decision Hub should apply an flow template to the imported actions.
   - **None** - Actions are imported with a default flow. The default flow consists of a Start and an End shape.
   - **Reference flow from another action** - For each action in the .csv file, you can specify an existing action to be used as template for the imported action. The imported action is created with the same flow as the template action. If the template action is not found, the new action is imported with a default flow. The default flow consists of a Start and an End shape.
   - **Copy flow from another action** - If your .csv file does not specify individual templates for each action, you can select one action to use as template for all actions in the .csv file. The imported actions are created with the same flow as the template action.
6. **Optional:** If you want to completely overwrite existing actions with actions from the .csv file, select **Delete existing actions that are not in the file** .
   This will delete any action which is not included in the .csv file you are importing.
7. Click **Import**. **Result:** The pop-up window displays the import summary, including the number of actions imported, the number of actions deleted, and the number of errors. If there are any errors in the .csv file, download the error report and resolve each error before you continue.

- Exporting Actions

Monitoring actions

After configuring an action and propagating it to customers via Campaigns, marketers can monitor the action's progress and, if necessary, perform certain actions on active actions. The Wait Volumes overlay provides this capability.

Pega Customer Decision Hub

Marketers can use this overlay to perform the following actions:

- View Waiting Action Counts
- Update Expiration Time for Waiting Actions
- Stop Waiting Actions

Note: New deployments of Pega Customer Decision Hub on Pega Cloud services are configured to use the Decision Data Store for action state storage. Wait Volumes overlay is disabled in this mode. If you want to enable the Wait Volumes overlay feature, see Disabling the Decision Data Store as action state storage on Pega Cloud.

- View Waiting Action Counts
- Update Expiration Time for Waiting Actions
- Stop Waiting Actions

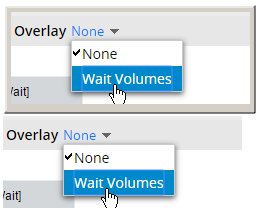
View Waiting Action Counts

Applying the Wait Volumes overlay on a particular action displays the number of waiting instances of the action for the selected Campaign. The overlay displays this count for each wait-enabled shape in the action.

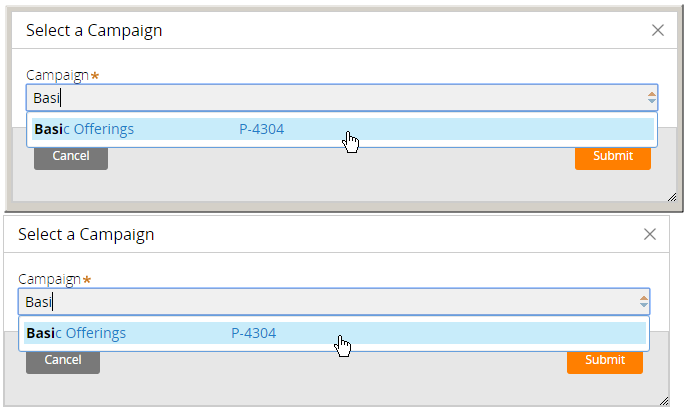
Pega Customer Decision Hub

To view this overlay, perform the following steps:

1. Select the **Wait Volumes** item from the **Overlay** toolbar menu in the action canvas.

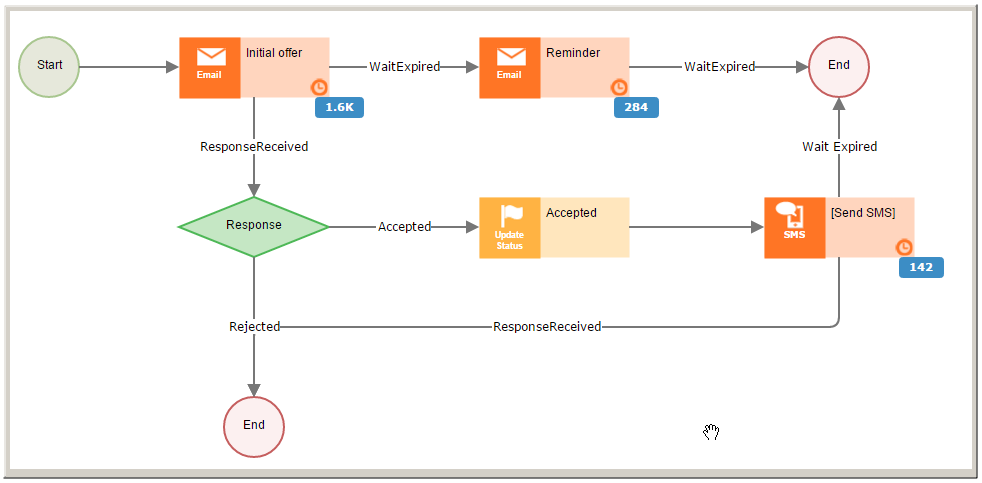


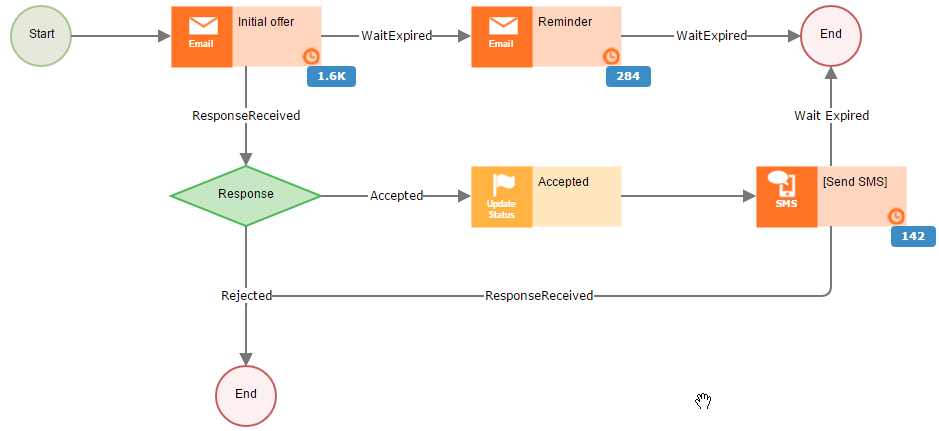
2. Specify the Campaign for which you wish to view waiting action counts. You can specify either the name or the work id of the Campaign.



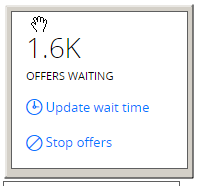
3. Click **Submit**.

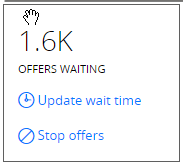
Upon application of the overlay, the count of action waiting is shown on every wait-enabled shape. In addition, the name and work id of the selected Campaign is displayed in the canvas toolbar next to the Overlay menu. The count shown on each shape represents the number of instances of the action that are waiting at the shape and have been initiated as part of the selected Campaign - either via a scheduled run or via a triggered Event.





Hovering over the count on a shape displays an expanded view of the overlay with links to perform the Update Expiration Time for Waiting Actions and Stop Waiting Actions actions.





Note: Action links in the overlay are disabled when no actions are waiting at the corresponding shape or if an action is currently being applied.

To view the wait volumes for a different Campaign, re-apply the overlay (by following steps 1-3 above) and select the desired Campaign.

To remove the applied overlay, select the **None** item from the Overlay menu.

Update Expiration Time for Waiting Actions

The **Update wait time** overlay action enables marketers to update the time left before instances of the action waiting at a particular shape expire. An example use case for this action would be to grant customers more time to respond to an action.

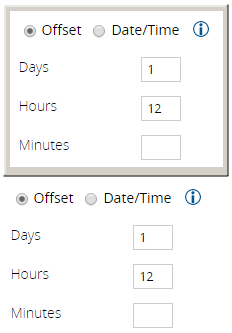
Pega Customer Decision Hub

To initiate this action, click the action link from the overlay detail pane.

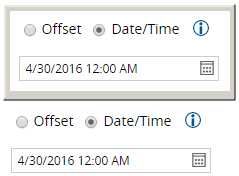


Then, choose one of the following options to set the new wait expiration time:

- **Offset** - Select this option to increment the current expiration time of each waiting action by a specified duration. Use the **Days**, **Hours**, and **Minutes** fields to specify the desired offset.

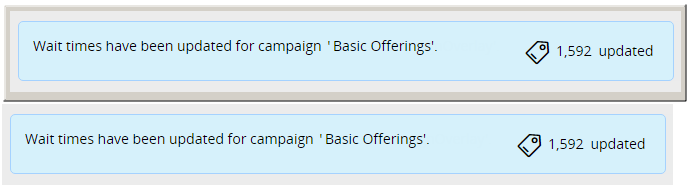


- **Date/Time** - Select this option to set the expiration time of each waiting action to a specified future date and time. Specify the desired date and time.



Click **Submit** to update the expiration time of waiting action. The system performs this update in the background. During this time period, a busy message is shown at the top of the Action rule and all further overlay actions are disabled.

Upon completion of the update, a confirmation message is displayed along with the number of waiting actions that were updated.



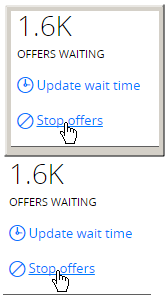
Stop Waiting Actions

The **Stop actions** overlay action enables marketers to stop execution of all instances of the action that are waiting at a particular shape. This provides granular control over the processing of the flow.

Pega Customer Decision Hub

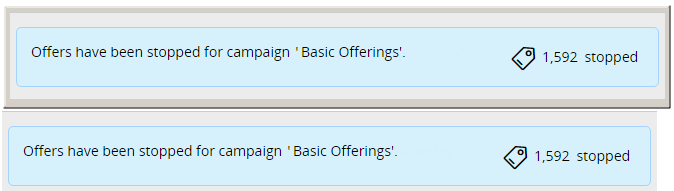
For example, if we had an action with an initial outreach (Email, SMS, etc.) followed by other downstream processing. If the outreach was under-performing or if the action was no longer viable, the marketer could stop all instances waiting at the outreach shape without impacting any customers who had already responded and progressed to a later stage in the flow.

To initiate this action, click the action link from the overlay detail pane.



Confirm the stop request by clicking **Submit**. Upon submission, the system performs the update in the background. During this time period, a busy message is shown at the top of the Action rule and all further overlay actions are disabled.

Upon completion, a confirmation message is displayed along with the number of waiting actions that were stopped.



Reviewing action performance

The side bar on the right side of the **Details** tab displays information pertaining to the performance of the action. In addition to the action image, the side bar contains the **Performance** and **Key Predictors** sections.

Pega Customer Decision Hub

- Performance
- Key Predictors

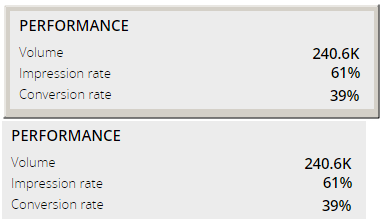
Performance

This section provides an overview of how the action is performing across the following three metrics:

Pega Customer Decision Hub

- **Volume** - Total number of times this action has been made
- **Impression rate** - Ratio (expressed as a percentage) of the number of times this action has been opened (i.e. impressions) to the volume of the action
- **Conversion rate** - Ratio (expressed as a percentage) of the number of times this action has been accepted to the volume of the action

A sample performance summary is shown below.



PERFORMANCE
Volume 240.6K
Impression rate 61%
Conversion rate 39%

Key Predictors

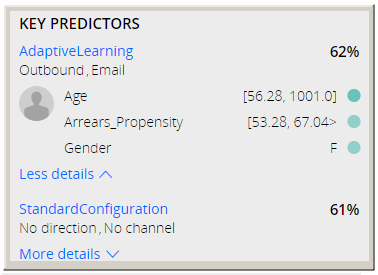
This section provides information on Adaptive Models that include this action. Each model is listed separately and the first model is expanded by default. The information presented for each model provides a quick profile of a customer that is likely to respond to this action.

Pega Customer Decision Hub

The following information is presented for each model:

- Model name - Click to open the behavior report associated with the model.
- Overall performance of the model
- Direction and channel for the corresponding predictor information
- Active predictors for the model. For each predictor, the group with the highest propensity is shown. Hover over the green circle to see this propensity value.

A sample predictors summary is shown below.



KEY PREDICTORS
AdaptiveLearning 62%
Outbound, Email
Age [56.28, 1001.0]
Arrears_Propensity [53.28, 67.04>
Gender F
Less details
StandardConfiguration 61%
No direction, No channel
More details

Model learning for new actions

In Pega Customer Decision Hub, every action is backed by an AI model. Additionally, every treatment within an action is also backed by an AI model. In the rest of this article "actions" mean both "actions and treatments".

Pega Customer Decision Hub

The output of a model is a prediction known as a propensity. Propensity is the likelihood of a customer behaving favorably towards an action. The propensity output of a model could be different for different customers. A model calculates the propensity based on various parameters including but not limited to customer information and their past interaction data. This prediction output of a model represents the relevancy of an action for a customer.

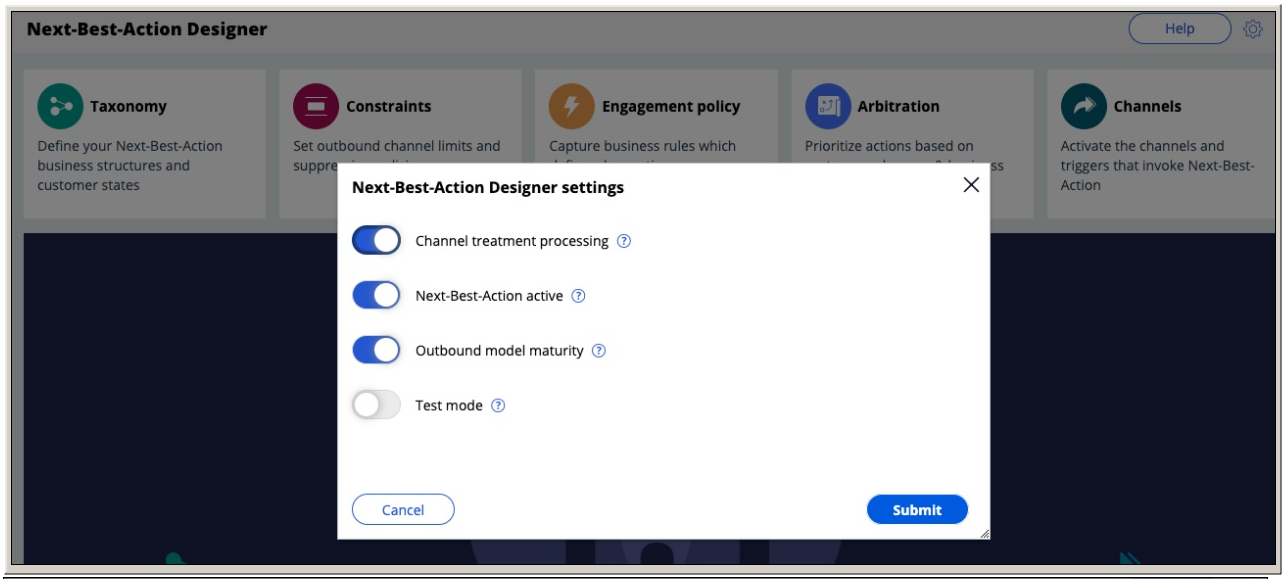
The AI models are self-learning, that is, they improve the quality of their prediction automatically as and when they receive feedback in the form of customer responses. The more customer responses they receive, the more reliable their predictions are. Conversely, the propensity output of models for newly introduced actions is not reliable as they would have too few or no customer responses to learn from.

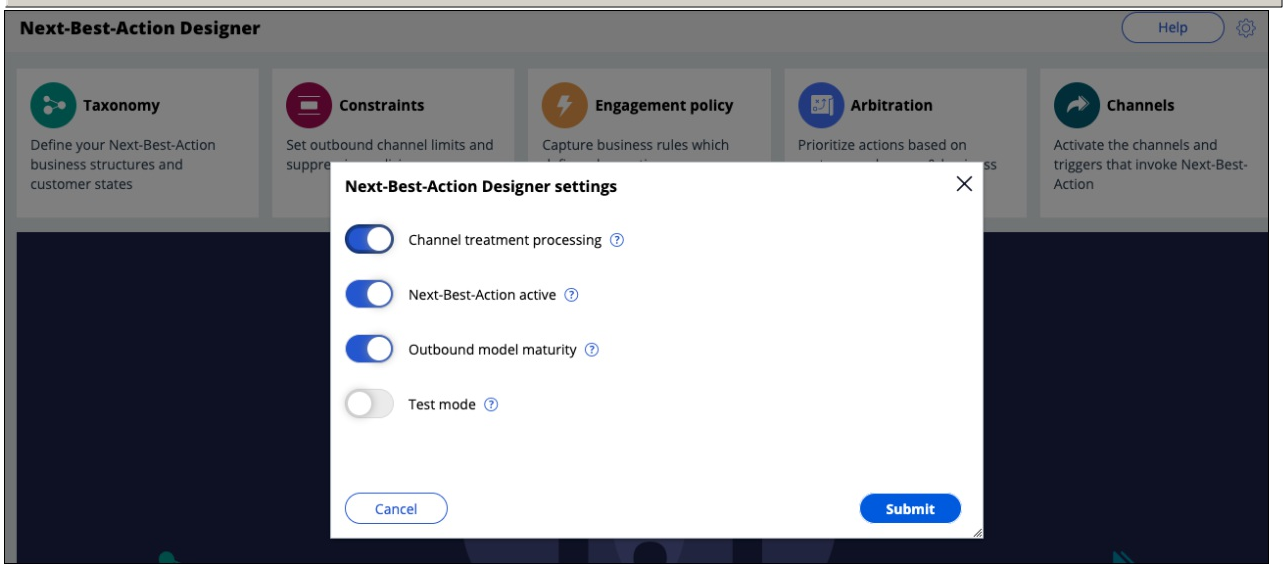
There are conditions in which it is insufficient to rely on the raw propensity returned by the AI models in order to select the right action. For example, when you turn on the system for the first time or launch a new set of actions all at once, all their propensities will be the same. So, the arbitration logic will not be able to differentiate between them. Or, if some of the actions are backed by mature models and you introduce new actions, the new actions will not get an opportunity to be presented.

Pega Customer Decision Hub complements the raw output of AI models with different mechanisms to ensure that customers receive actions that are relevant to them.

Outbound model maturity

The propensity of AI models will be less reliable in the beginning. This could result in customers receiving irrelevant actions. To help ensure that customers continue to receive relevant actions on outbound channels, Pega Customer Decision Hub provides the Outbound model maturity option in Next-Best-Action Designer.





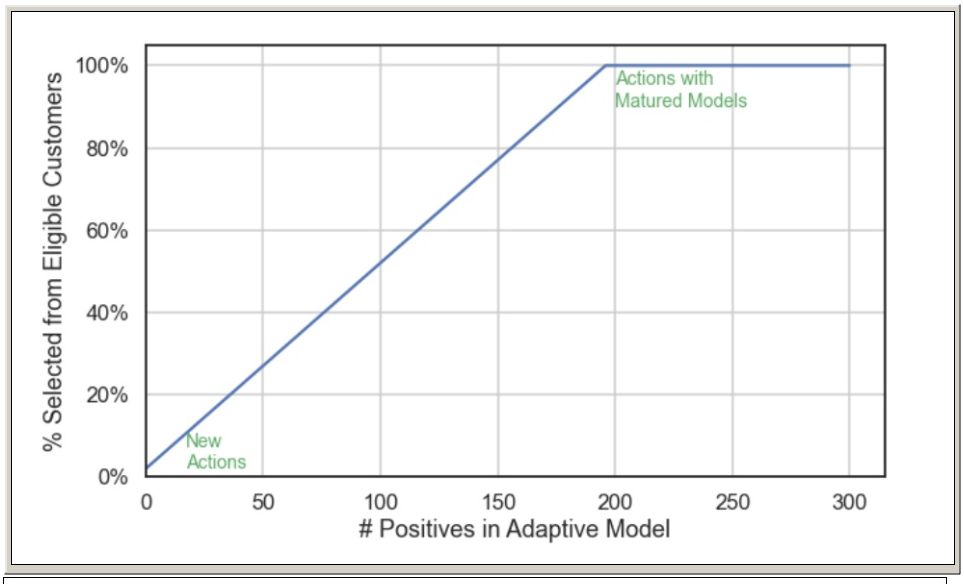
When this option is enabled, Pega Customer Decision Hub initially targets the outbound actions only to 2% of the population who qualify for those actions. The models start to learn as they receive customer responses. As the models start to learn, the target population for corresponding actions is gradually increased. When the models receive about 200 positive responses, 100% of the population is allowed to be presented with those actions. This approach helps in a good distribution of actions as each new action is exposed to a reduced set of audience and every new action gets a fair opportunity to learn from customer responses.

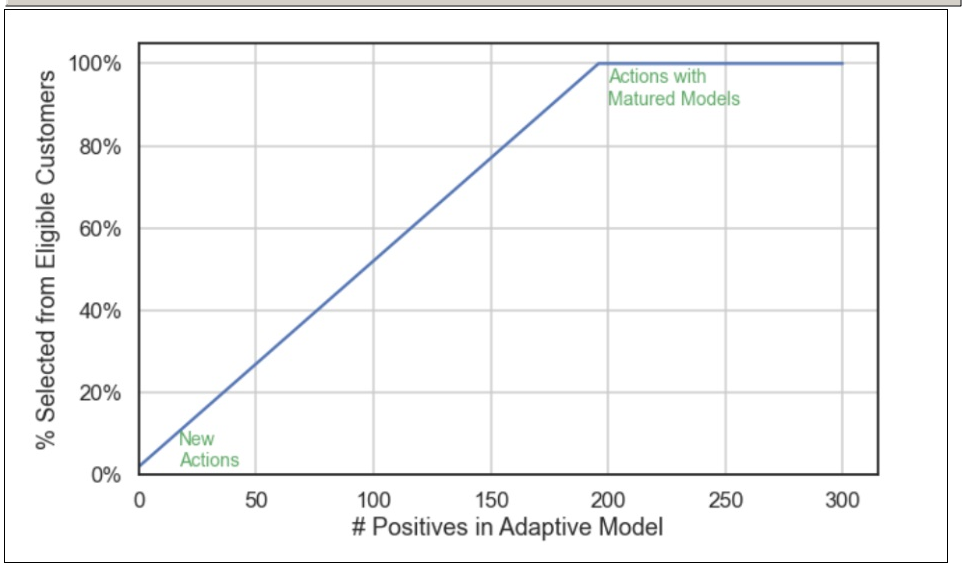
The target population for an action with an immature model is determined using the formula:

$$X = 0.02 + \min\left(0.98, \frac{no.\ of\ positives}{200}\right)$$

$$X = 0.02 + \min\left(0.98, \frac{no.\ of\ positives}{200}\right)$$

Where X is the percentage of starting population considered for targeting the action.





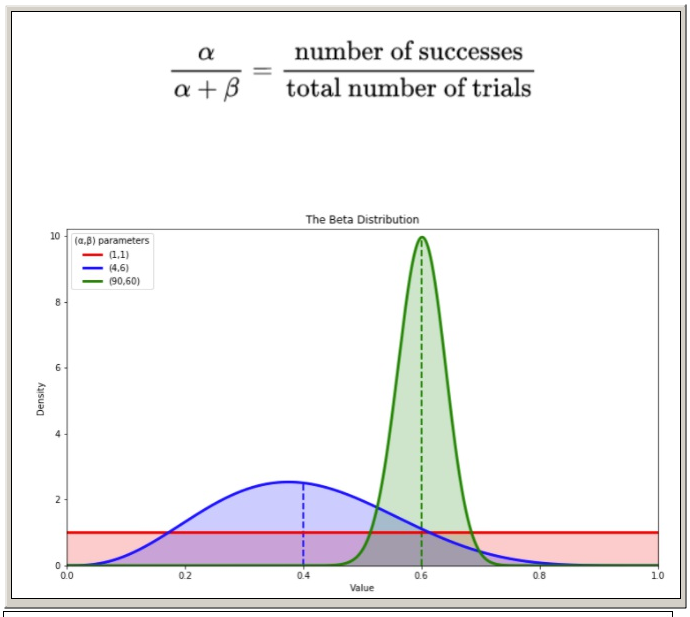
When Outbound model maturity is turned off, no "capping" is applied to the target population for targeting outbound actions with immature models. All (100%) qualified customers are considered, which could lead to spamming the customers with irrelevant actions.

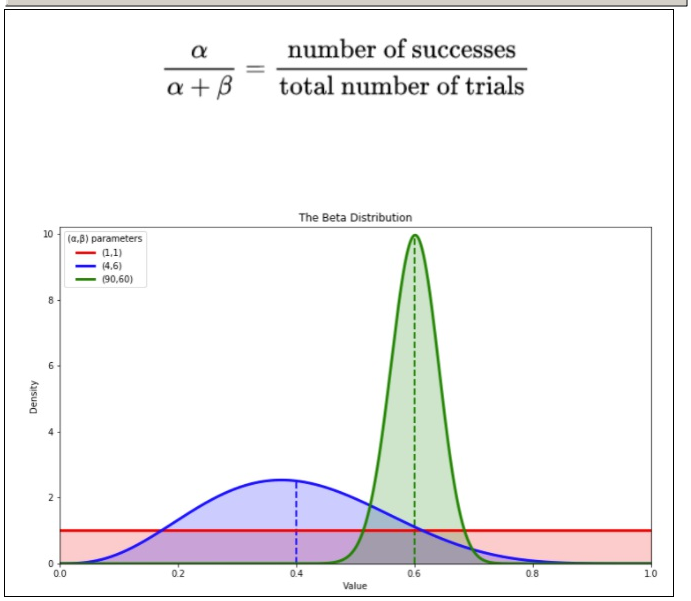
Thompson sampling

The starting propensity of new models will be 0.5 for every customer. When multiple new actions are launched, there will be many actions with the same propensity. This could result in the same actions being selected repeatedly while other actions get suppressed.

To avoid this problem, Pega Customer Decision Hub replaces the original model propensity with the propensity selected from a range that centers around the model propensity. The width of the range depends on the number of responses the model has gathered. For new actions, the propensity is randomly chosen between 0 and 1. When the model gathers more responses, the selected propensity is closer to the model propensity. This can also be described as adding a small amount of noise to the propensity. If there is a lot of uncertainty, more noise is added. If there is little uncertainty (that is, the model has received a lot of responses), there is little noise, and the returned propensity is very close to the model propensity. So, this approach effectively only applies to immature models, namely models that have not received a lot of feedback.

Thompson sampling mechanism samples from the beta distribution around the model propensity when it adds noise to that propensity. The following graph shows the mean of the beta distribution. It demonstrates, how the calculated propensity deviates as a function of the number of positive responses (number of successes) and the number of times an action is offered to customers (total number of trials). The calculated propensity gets closer to the original propensity as the model receives more positive responses.





Note: Unlike Outbound model maturity, the Thompson sampling is always applied and is applicable to both inbound and outbound actions.

AI Models control group

AI models can aid in selecting relevant actions for customers. Lift is the benefit you get by using AI models for action selection versus not using AI models. Pega Customer Decision Hub uses model control group mechanism for lift measurement. This mechanism also aids in continuous exploration to provide the model learning with unbiased samples.

You define the control group when configuring a Prediction. You can either define the control group as a percentage of all customers (the customers are then chosen randomly) or choose the customers for the control group yourself by assigning a specific field value to the customers that you want to include in the control group. For customers in the control group, Pega Customer Decision Hub applies a random propensity to the actions. Whereas, for customers in the test group, customers not in the control group, the original propensity from the model is used.

Note that, all the other conditions defined in engagement policies and arbitration still apply. Therefore, this mechanism acts only as an AI models control group rather than a global control group that tests the end-to-end next best action strategy framework.

Tip: The following mechanisms are implemented in the Predict Action Propensity prediction within the next-best-action strategy framework.

Setting constraints, contact policy limits, and controls

You can restrict the number of contacts a customer receives in a given period of time, or configure an automatic pause for actions after a specific number of outcomes.

Pega Customer Decision Hub

Too many contact attempts over a short period of time can have a negative impact on a customer's willingness to further interact with your actions. They can also be non-compliant with corporate or regulatory guidelines. To prevent that, you can specify the following constraints for contacting your customers:

- **Customer contact limits** - Limit the number of interactions which a customer can receive over a given period of time on a specific channel. For example, you can decide that you do not want your customers to receive more than two emails per week.
- **Contact policy library** - Define suppressions, that is, automatically put an action on hold after a specific number of outcomes are recorded for some or all channels. For example, a contact policy can suppress an action for a particular customer after the customer has seen an ad for that action for five times. Suppressing or pausing an action prevents oversaturation by limiting the number of times that a customer is exposed to the same action.

1. In the Pega Customer Decision Hub portal, click **Next-Best-Action Designer Constraints** .
2. Click **Edit**.
3. **Optional:** To set the customer contact limit, in the **Customer contact limits** section, click **Edit customer contact policy**.
   Note: Pega Customer Decision Hub has a predefined Customer Contact Policy which sets the customer contact limits to one email and one SMS per week. Edit the policy to configure the channels, contact limits, duration, and the policy start and end date. For more information, see Controlling the number of customer contacts with contact policies.

4. **Optional:** To specify action suppressions, in the **Contact policy library** section, click **+ Add contact policy**.
5. **Optional:** Enter the policy name.
6. In the **Scope** section, select the type of outcome which is tracked by the policy. For example, if the suppression should activate after a specific number of negative responses, select **Track Rejects**.
7. Specify whether to track responses individually for each action or for all actions in the group.
   Note: If you track outcomes for all actions in the group, then all of these actions are paused once the suppression criteria are met.
8. Select the time period over to track the responses.
   By default, Pega Customer Decision Hub tracks responses over a period of 7 or 30 days. To add more tracking periods:
   a. In Dev Studio, click **Configure Decisioning Decisions Data Sources Interaction History Summaries** .
   b. Edit the Interaction History rule *Action Outcomes for the past 7 Days* or *Action Outcomes for the past 30 Days*.
   c. Save a copy of the rule into your data set and adjust the time period as required.
9. Click **Submit**. **Result:** The suppression configuration appears in the **Contact policy library** section.
10. Enter the number of responses required to fulfill the suppression criteria, for example, 30.
11. Select the channel for which you want to track the responses, for example, **All**.
12. Enter the number of days for which an action should be paused after the suppression criteria are met, for example, 5. **For example:** An example suppression is If there are 30 Rejects for All treatments, suppress the action for 5 days.
13. **Optional:** To add more suppression conditions, click **+ Add suppression**.
    Note: Suppression conditions defined under the same contact policy track the same outcomes over the same period of time. They can, however, monitor different channels and have different outcome limits for each channel.
14. **Optional:** To track different metrics over different periods of time, click **+ Add contact policy**.
    Note: For more information on adding additional tracking periods, see Adding more tracking time periods for contact policies.
15. Click **Save**.
16. To apply a contact policy to your Next-Best-Action Strategy, select it in the **Engagement policy** tab of Next-Best-Action Designer:
    a. In the Pega Customer Decision Hub portal, click **Next-Best-Action Designer Engagement policy** .
    b. Click the name of a group.
    c. Click **Edit**.
    d. In the **Contact policy** section, click the **Configure** icon.
    e. In the **Configure contact policy** window, select your new contact policy and click **Add**.
    f. Click **Apply**.
    g. Click **Save**.

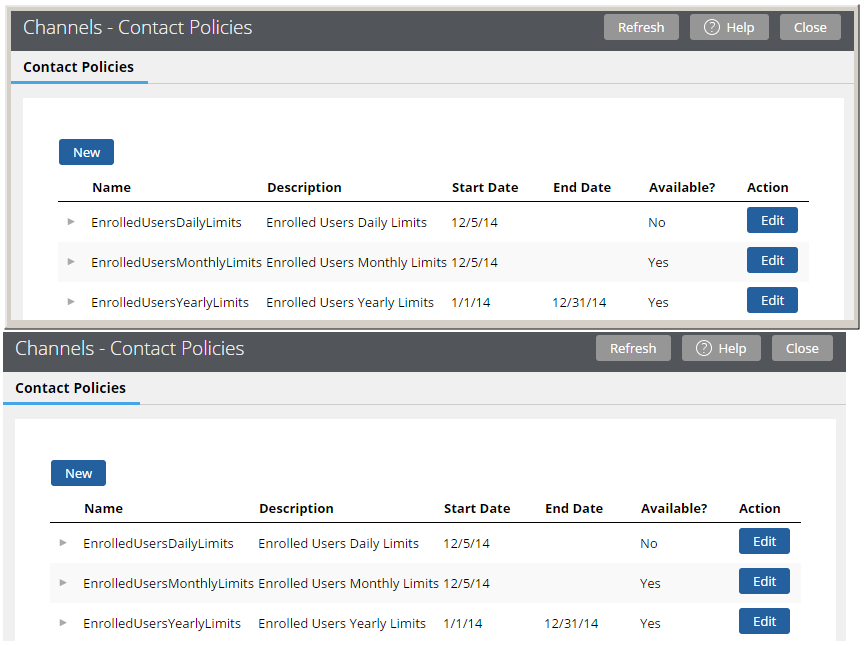
- Controlling the number of customer contacts with contact policies
- Creating a contact policy
- Adding more tracking time periods for contact policies
- Frequently asked questions about contact policies
- Avoiding overexposure of actions with volume constraints

Controlling the number of customer contacts with contact policies

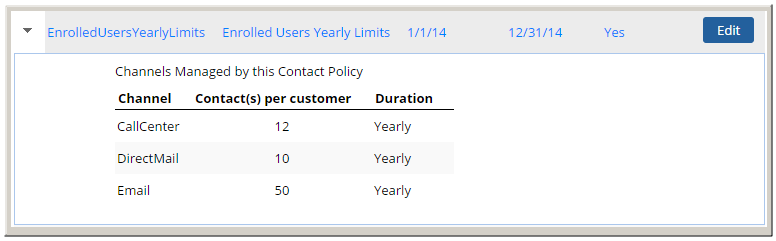
Contact Policies provide a mechanism for controlling and restricting how many times a customer is contacted in accordance with corporate and/or regulatory guidelines. A single policy can contain the contact limits for multiple channels of communication. Each contact limit is applicable over a specified time period (e.g. weekly, yearly, etc.). Furthermore, the contact policy, as a whole, can be disabled, enabled only for a specified date range, or always enabled.

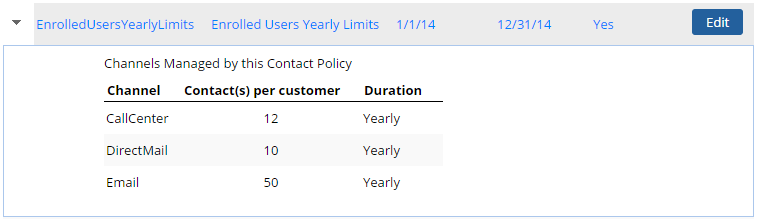
Pega Customer Decision Hub

Contact Policies are accessible in the Pega Customer Decision Hub via **Configuration Contact Policies** .



The **Contact Policies** landing page provides a quick overview of the various Contact Policies configured in the application. The landing page displays the start and end date of each policy along with its availability. Details about the limits specified for a particular policy are visible by expanding the row for the individual policy (via the arrow on the left side), as shown below:



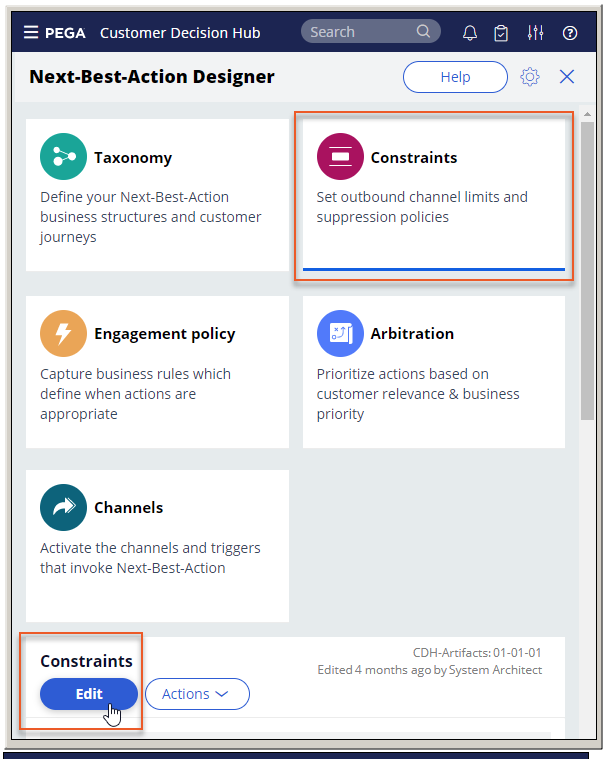


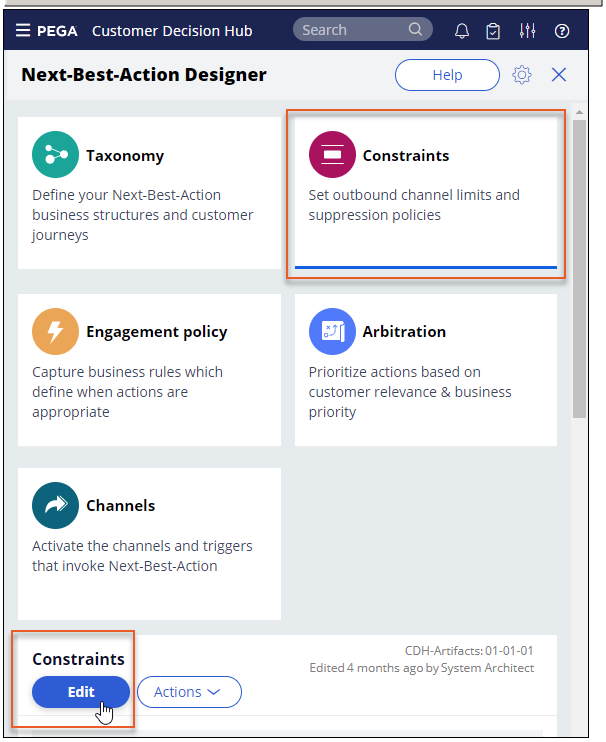
Creating a contact policy

Contact policies control how many contact attempts the customer receives through a specific channel over a set period of time. To define those suppression rules create a new contact policy.

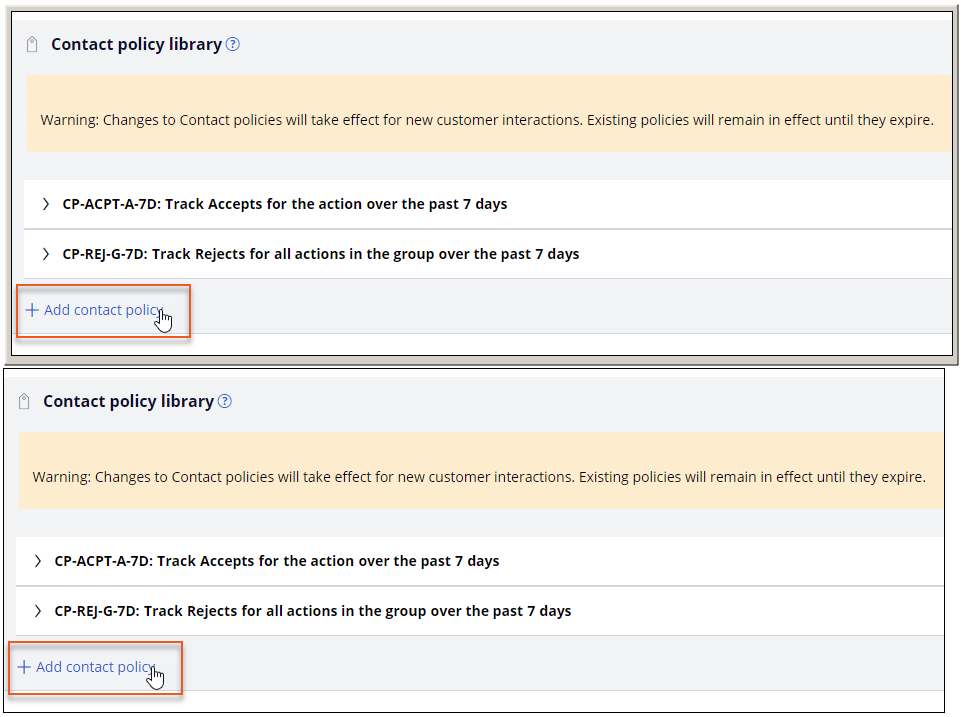
Pega Customer Decision Hub

1. Log in to your Pega Customer Decision Hub application.
2. Click **Next-Best-Action Designer Constraints Edit**

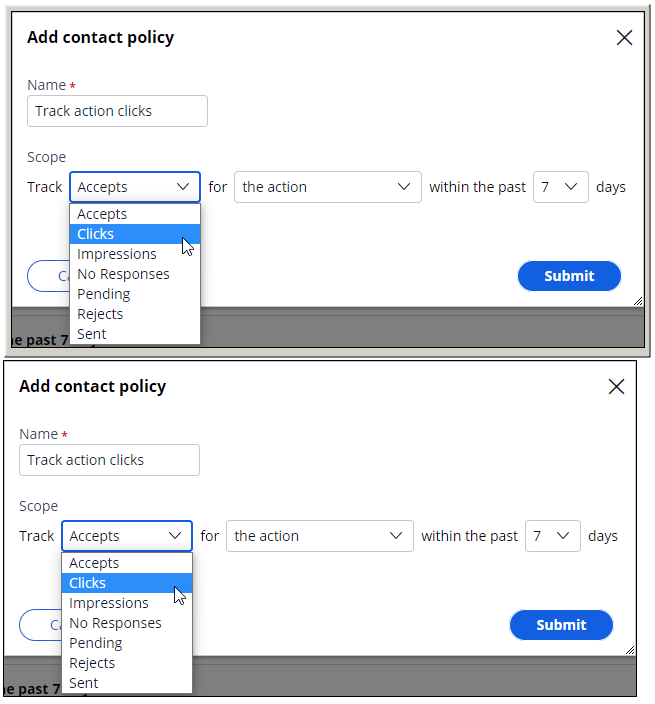




3. In the **Contact policy library** section, click **Add contact policy**.



4. In the **Add contact policy** window, enter the **Name** and **Scope** for the new contact policy.



Note: For more information about how to add more response outcomes, see Adding new response outcomes.

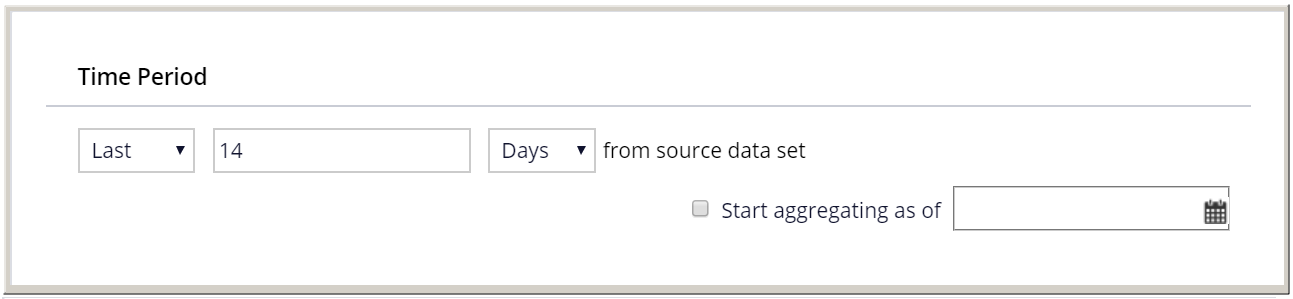
5. Confirm by clicking **Submit** button.

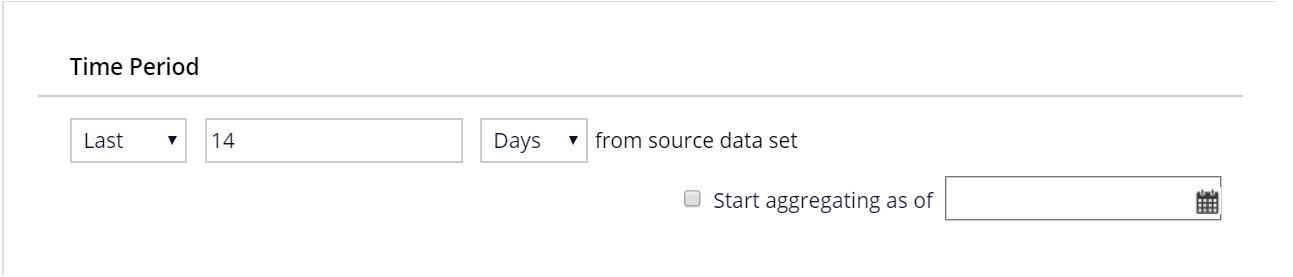
Adding more tracking time periods for contact policies

Contact policies track responses to actions over a specific period of time. By default, responses are tracked over a period of 7 or 30 days. If required, you can add more tracking periods by creating a new Interaction History time period, and then updating the part of the Next-Best-Action Strategy that references it.

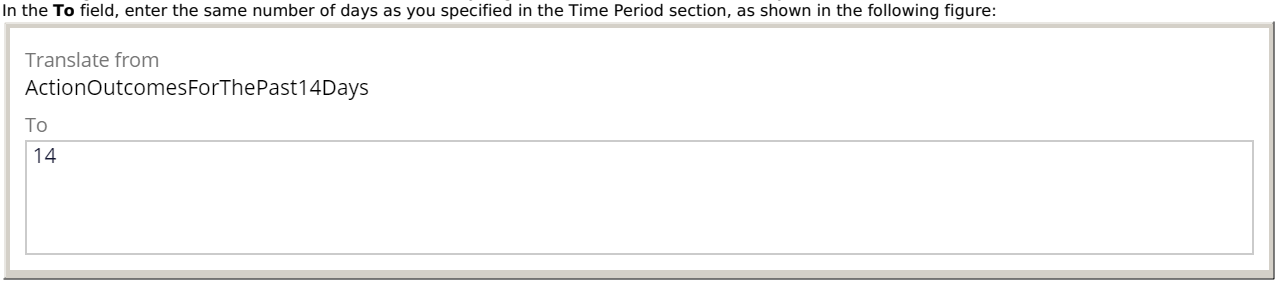
Pega Customer Decision Hub
Note: Contact policies determine when and for how long an action or a group of actions should no longer be shown to the customer. For example, if a customer views an ad for an action five times without responding, the add will no longer be shown to that customer over the next seven days. Suppressing or pausing an action prevents over-saturation by limiting the number of times that a customer is exposed to the same action.

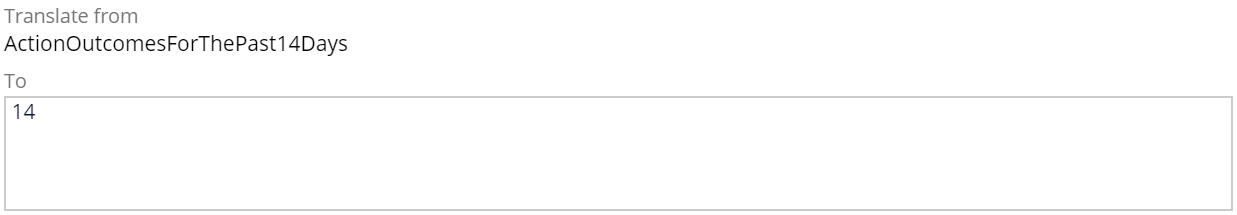
1. In Dev Studio, click **Configure Decisioning Decisions Data Sources Interaction History Summaries** .
2. Open the Interaction History rule *Action Outcomes for the past 7 Days* or *Action Outcomes for the past 30 Days* by clicking **Manage Edit** .
3. Click **Save as** to save a copy of the rule into your data set.
4. In the **Time Period** section, adjust the time period as required, as shown in the following figure:



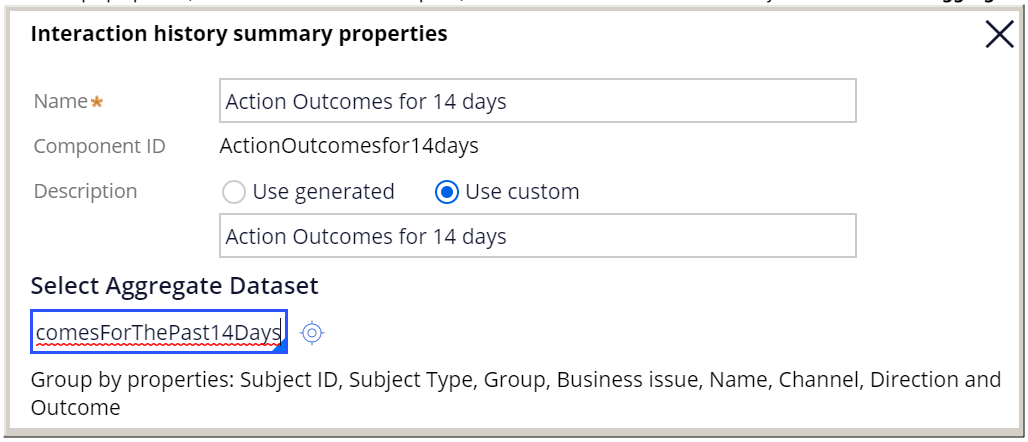


5. Click **Save**.
6. Click **Check in**.
7. Search for and open the *ActionOutcomesForThePast07Days* or *Action Outcomes for the past 30 Days* field value.
8. Click **Save Save as** to save a copy of the rule into your data set.
   You must give the new field value rule the same name as you gave to the new Interaction History rule.
9. In the **To** field, enter the same number of days as you specified in the Time Period section, as shown in the following figure:





10. Click **Save**.
11. Click **Check in**.
12. Search for and check out the *CheckOutcomeLimitsIHSummary* strategy.
    This strategy checks the contact policy threshold for specific channels, for example, web, email, or SMS. It is a sub-strategy of the *BehavioralLimits* strategy.
13. Add an IH Summary shape to the strategy canvas by copying one of the existing shapes.
14. In the shape properties, fill out the name and description, and select the new Interaction History rule in the **Select Aggregate Dataset** field, as in the following figure:



Interaction history summary properties

Name* Action Outcomes for 14 days

Component ID ActionOutcomesfor14days

Description ○ Use generated ◉ Use custom

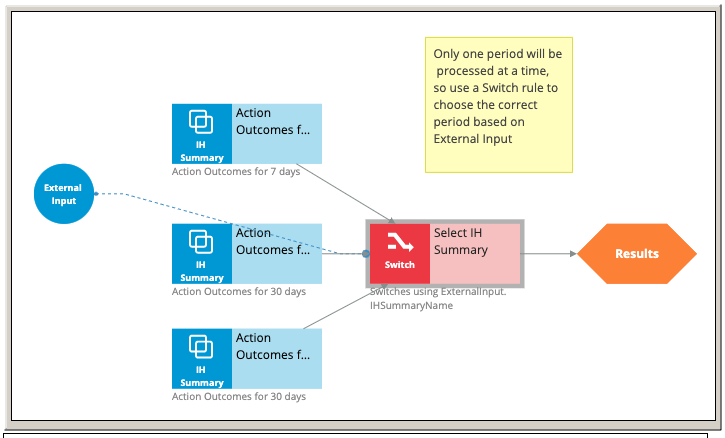
Action Outcomes for 14 days

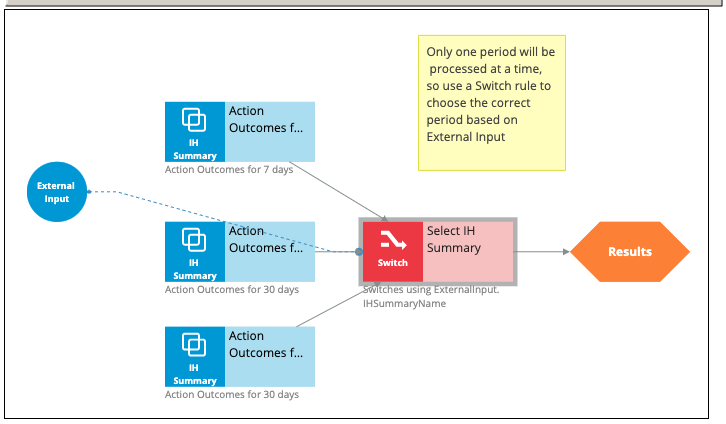
Select Aggregate Dataset

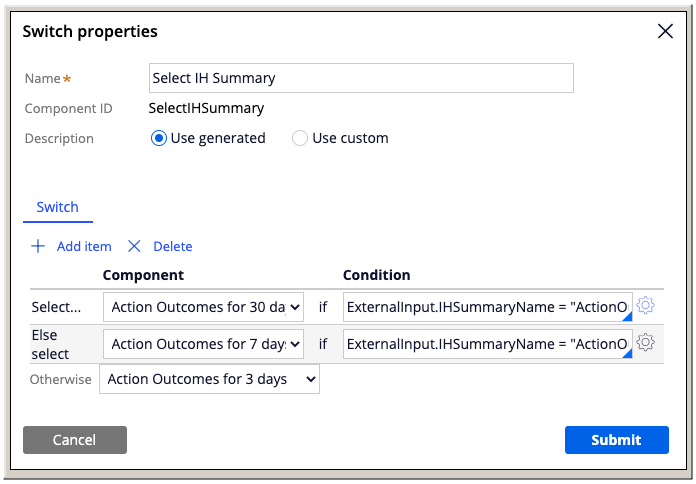
comesForThePast14Days

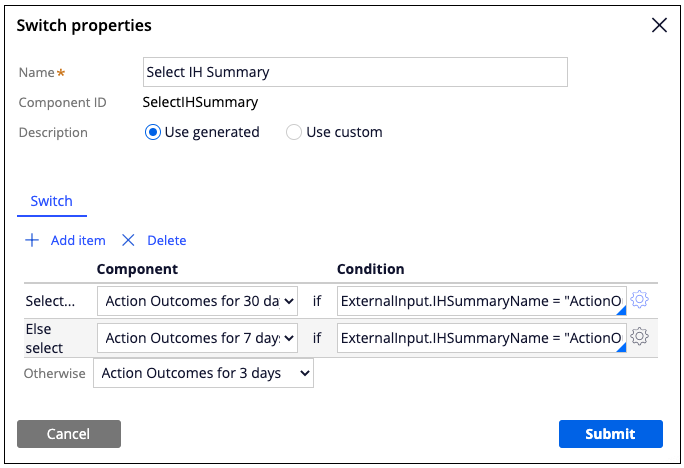
Group by properties: Subject ID, Subject Type, Group, Business issue, Name, Channel, Direction and Outcome

15. Modify the *Select IH Summary* switch rule component to reference your new IH Summary.
    a. Add item in switch component with the condition *ExternalInput.IHSummaryName =ActionOutcomesForThePastxxDays*.









b. If necessary, adjust and add conditions to the already existing component.

Frequently asked questions about contact policies

See the following topics for additional information, tips and tricks, and best practices.

Pega Customer Decision Hub

Can I create contact policies looking back on more than 30 days of interaction history?

Yes. For more information, see Adding more tracking time periods for contact policies.

If I create a contact policy on the group level, can I override it for individual actions, so that the contact policy does not apply to these actions?

This is currently not possible. If your contact policy does not apply to all actions in a group, do not create it on the group level.

Avoiding overexposure of actions with volume constraints

A Volume Constraint limits the number of actions that are delivered to customers.

Pega Customer Decision Hub
Note: New deployments of Pega Customer Decision Hub on Pega Cloud services are configured to use Decision Data Store for action state storage. Volume Constraints are disabled in this mode. If you want to enable the Volume Constraints feature, see Disabling the Decision Data Store as action state storage on Pega Cloud.

You can define the following constraint types:

- Per channel - Limits the number of actions delivered on a specific channel. For example, you can set a constraint that limits the number of emails sent per day.
- Per action - Limits the number of times a specific action is offered to customers. For example, you can set a constraint that limits the number of times that a specific Credit Card offer is presented to customers.
- Per action property - Limits the number of times a group of actions is offered to customers. The actions are grouped based on a specific property or properties. For example, you can limit the number of actions which belong to a specific business issue.

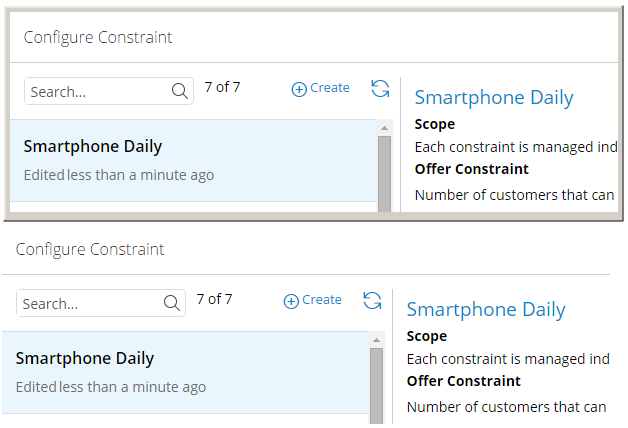
Channel constraints can be combined with action or action property constraints. For example, you can specify that no more than 1000 Credit Card offer emails can be sent in a given time period.

You can configure the constraint to require a minimum as well as a maximum number of actions. For example, you can specify that the campaign requires at least 50000 call center interactions.

To apply these constraints, the Volume Constraint rule should be associated with a Campaign or with an outbound run in the Next-Best-Action Designer. For Campaigns, when a Campaign runs, it first executes the Strategy to determine the potential set of actions that should be processed. Next, if a Volume Constraint has been associated with the Campaign, any enabled action and channel constraints are applied to the action set to determine the final list of actions which needs to be processed.

Note: If a Volume Constraint has been associated with a Campaign, it must have at least one constraint enabled. If no constraints are enabled on this Volume Constraint, the Campaign will not execute. At this point, the user can either enable a constraint and restart the specific run; or update the Campaign to no longer use Volume Constraint and re-submit the Campaign for execution.

Volume Constraints are primarily accessed from and created in the context of Campaigns. The Constraint setting of a Campaign references a Volume Constraint rule. When configuring this setting, users can use the **Configure Constraint** selector to review existing Volume Constraints and create a new one if needed.



Configure Constraint
Search...
7 of 7
Create
Smartphone Daily
Edited less than a minute ago
Smartphone Daily
Scope
Each constraint is managed ind
Offer Constraint
Number of customers that can

In addition to Campaigns, you can apply Volume Constraints in the Next-Best-Action Designer. For more information, see Setting constraints, contact policy limits, and controls.

To view all Volume Constraint rules configured in your system, in the Pega Customer Decision Hub portal, click **Intelligence Volume Constraints** .

- Creating a volume constraint
- Using volume constraints with action bundles
- Frequently asked questions about volume constraints

Creating a volume constraint

In situations where a customer is eligible for multiple actions, you may specify whether volume constraints should be applied to the actions as a group, or whether each action should be evaluated separately. Evaluating actions as a group allows Pega Customer Decision Hub to display all eligible actions, for example to a customer service representative, who should be able to see all the actions they can make to the customer.

Pega Customer Decision Hub
The two options available are:

- **Constrain each action individually** - Select this option to check each action or action bundle for the customer separately against applicable channel and action constraints. In this way, the highest priority action or actions for a customer may pass constraint checks, while lower priority actions may be rejected, for example due to volume constraints.
  Note: If the marketing Strategy pushes out multiple actions for a customer, these actions should be prioritized in the Strategy since the highest priority action (based on the pxPriority field) will be made to the customer.
- **Constrain as a group for the customer (output all actions)** - Select this option to use the highest priority action to apply constraints on the channels and actions, without removing the other actions for a customer. If the highest priority action is allowed to pass the constraint, all lower priority actions for this customer are also allowed without applying separate constraint checks. In this way, all eligible actions for a customer can be displayed.
  Note: You can automatically apply the setting **Constrain as a group for the customer (output all actions)** to all volume constraints in all the campaigns in your application. To do this, edit the Dynamic System Setting *MKTConstraintAsGroup* and set its value to `true`. If the value of the setting is set to `true`, the two customer volume constraint options are no longer displayed when you edit a volume constraint.

Using volume constraints with action bundles

You can use Volume Constraints together with action bundles. If the parent action of a bundle passes the Volume Constraint, all child actions for the bundle are also passed to the action processing stage. If the parent action does not pass the Volume Constraint, all child actions for the bundle are also rejected.

Pega Customer Decision Hub

Child actions are not processed as individual items during volume constraints. This applies to all bundle types. For more information about action bundles, see Grouping offerings into action bundles.

Frequently asked questions about volume constraints

See the following topics for additional information, tips and tricks, and best practices.

Pega Customer Decision Hub

Can I use volume constraints with action bundles?

Yes. If you grouped any of your actions into bundles, all volume constraints apply to the action which you marked as the bundle parent, that is, the main offering of the bundle. If the parent action passes the volume constraint, all child actions for the bundle also pass. If the parent action does not pass the volume constraint, all child actions for the bundle are also rejected. Child actions are not processed as individual items during volume constraints.

Configuring engagement policies for business groups

Edit each group and define eligibility, relevancy, and suitability conditions for actions associated with this group.

Pega Customer Decision Hub

Engagement policies specify the conditions under which an action or group of actions is eligible for a customer. For more information, see Understanding Next-Best-Action Designer engagement policies.

Note:

If you have configured a multi-level Context Dictionary, you will see a separate engagement policy section for each context. Each context section allows you to define conditions using that context level and above.

For example, if your Context Dictionary structure is `Accounts > Subscribers`, the **Subscribers** engagement policy section allows you to use criteria from both the Subscriber and Accounts entities. There is also an **All Actions** section, where only top-level criteria are available. The criteria specified in the **All Actions** section are applied to all actions in the group, regardless of the context of the action.

1. In the Pega Customer Decision Hub portal, click **Next-Best-Action Designer Engagement policy** .
2. On the **Engagement policy** tab, click the name of a group, or click **All groups** to define an engagement policy that applies across all groups in your business structure.
3. Click **Edit**.
4. Expand the **Eligibility**, **Applicability** or **Suitability** section.
5. In the first column, select a field, a strategy, or a when condition from the list.
   Note: To make a property, To make a property, when rule, or strategy available for use as an eligibility condition, define it as a Relevant Record for your primary context class, which by default is your Customer class, or for its the parent or child classes.
6. In the second column, select a comparator from the list. For strategies, the comparator is always **has results for**.
7. In the third column, select as needed:
   - For field conditions, enter a value in the field or select a field by clicking **Select values**. The system compares the value of this field to the value of the corresponding field or condition in the first column. For example, `Status work is equal to Done`.
   - For strategies, select the required component. The system determines whether there exist strategy results for that component.
   - When conditions do not have a third column, because their comparator can only be **true** or **false**.
8. To specify additional conditions and define the relationships between them, do the following steps:
   a. Click **Add a row** and add as many conditions as required.
   b. Use the drop-down lists between each pair of conditions to specify how these conditions relate to each other. You can group conditions with **AND** or **OR** operators.
   c. Use the drop-down list in the upper right corner to specify how condition groupings should be evaluated.
      - **Group ANDs** - Select this option if you want conditions linked with the AND operator to be evaluated as a group, and conditions linked with the OR operator to be evaluated individually. With this grouping, the grouping of condition 1 AND condition 2 OR condition 3 OR condition 4 is evaluated as `(1 AND 2) OR 3 OR 4`. That is, either conditions 1 and 2 must both be true, or either one of conditions 3 and 4 must be true.
      - **Group ORs** - Select this option if you want conditions linked with the OR operator to be evaluated as a group, and conditions linked with the AND operator to be evaluated individually. With this grouping, the grouping of condition 1 AND condition 2 OR condition 3 OR condition 4 is evaluated as `1 AND (2 OR 3 OR 4)`. That is, both condition 1 and one of conditions 2, 3, and 4 must be true.
      - **Use advanced logic** - Select this option if you want to define a mix of grouping by the AND and OR operator. Use the **Logic string** field to specify the condition grouping.

        If you change a condition that uses advanced logic back to Group ANDs or Group ORs, all the groupings that you previously defined are reset.
9. To save the condition for future reuse, click **Actions Save to library** .
   The **Save to library** option is not available if the conditions include either Strategy rules or When conditions with parameters. The option is also disabled for eligibility conditions which include default criteria created with the **Use advanced logic** option.
10. Click **Save**.
11. To test the behavior of each component of the engagement policy, you can run audience simulations and persona tests. For more information, see Simulating Next-Best-Action configuration changes.

What to do next: The **Actions** section displays all currently available actions from this group. If any actions have additional engagement policies, these policies are also shown in this section.**Tip:** To quickly create new actions for this group, you can click the **More** icon and then click **Create actions**.

Configuring engagement policies for actions

Define eligibility, applicability, and suitability conditions for your actions. Engagement policies allow you to specify the conditions under which an action or group of actions is eligible for a customer.

Pega Customer Decision Hub

You can define the following condition types as part of your engagement policies:

Eligibility
: Specify the conditions that the customer must fulfill in order to qualify for the action. For example, an action may only be available for customers in a specific geographic location.

Applicability
: Specify the conditions for the action to be relevant at this point in time. For example, a discount on a specific phone model may not be relevant for a customer who already owns a phone of this type.

Suitability
: Specify the conditions for the action to be appropriate for a customer. For example, a new credit card offer may not be appropriate for a customer who is likely to default on payments.

If you want to define eligibility conditions for multiple actions at the same time, edit the Business Group configuration on **Engagement policy** tab of the Next-Best-Action Designer. Any conditions defined at the Business Group level are displayed on top of the page, in the Group Eligibility section. You can toggle the **Apply** check box to allow or disallow them for this action. To view the Business Group configuration in the Next-Best-Action Designer, click the Business Group name in **Inherited from *<Business Group>***.

Tip: You can also create separate engagement policies for different customer contexts that you defined for your application. For example, to offer the account owner a family data plan that applies to the whole account, and also send out new phone offers to individual household members, you can define separate engagement policies for the Account and Subscriber contexts.
Design Patterns: You can configure an engagement policy to qualify the customers who have previously interacted with a specific action for a different action. For this use case, you need to create a strategy that filters users based on their interaction history, and then use this strategy to define a condition in your engagement policy. For instructions on how to configure the strategy, see Defining engagement policies to qualify customers for an action based on their interaction with another action.

1. **Optional:** To remove all previously added criteria, click **Actions Clear all criteria** .
2. On the **Engagement policy** tab, expand the **Eligibility**, **Applicability**, or **Suitability** section.
3. In the first column, select a field, a Strategy, or a When condition from the list.
   Note: To make a property available for use as an eligibility condition, define it as a Relevant Record for your primary context class, which by default is your Customer class. For more information, see Configuring properties for use in engagement policies and proposition filters.
4. In the second column, select a comparator from the list. For Strategies, the comparator is always **has results for**.
5. In the third column, select as needed:
   - For field conditions, enter a value in the field or select a field by clicking **Select values**. The system compares the value of this field to the value of the corresponding field or condition in the first column. For example, `Status work is equal to Done`.
   - For Strategies, select the required component. The system determines whether there exist Strategy results for that component.
   - When conditions do not have a third column, because their comparator can only be **true** or **false**.
6. **Optional:** Specify additional conditions and define the relationships between them.
   a. Click the **Plus** icon and add as many conditions as required.
   b. Use the drop-down lists between each pair of conditions to specify how these conditions relate to each other. You can group conditions with **AND** or **OR** operators.
   c. Use the drop-down list in the upper right corner to specify how condition groupings should be evaluated.
      - **Group ANDs** - Select this option if you want conditions linked with the **AND** operator to be evaluated as a group, and conditions linked with the **OR** operator to be evaluated individually. With this grouping, the grouping of condition 1 **AND** condition 2 **OR** condition 3 **OR** condition 4 is evaluated as (1 AND 2) OR 3 OR 4. That is, either conditions 1 and 2 must both be true, or either one of conditions 3 and 4 must be true.
      - **Group ORs** - Select this option if you want conditions linked with the **OR** operator to be evaluated as a group, and conditions linked with the **AND** operator to be evaluated individually. With this grouping, the grouping of condition 1 **AND** condition 2 **OR** condition 3 **OR** condition 4 is evaluated as 1 AND (2 OR 3 OR 4). That is, both condition 1 and one of conditions 2, 3, and 4 must be true.
      - **Use advanced logic** - Select this option if you want to define a mix of grouping by the **AND** and **OR** operator. Use the Logic string field to specify the condition grouping.

        If you change a condition that uses advanced logic back to **Group ANDs** or **Group ORs**, all the groupings that you previously defined are reset.
7. **Optional:** To save the condition for future reuse, click **Actions Save to library** .
   The conditions are saved to a new when rule in the top level class of the eligibility condition. The when rule is automatically registered as a relevant record for use in the condition builder.**Note:** The **Save to library** option is not available if the conditions include either Strategy rules or when conditions with parameters. The option is also disabled for eligibility conditions which include default criteria created with the **Use advanced logic** option.
8. **Optional:** In the **Contact policy** section, select one or more contact policies to associate with this action.
   A contact policy limits the number of times customers can be contacted on a particular communication channel over a certain period of time. For more information, see Controlling the number of customer contacts with contact policies.
9. **Optional:** Do steps 1 to 7 again to configure another type of engagement conditions.
   You can define any combination of Eligibility, Applicability, and Suitability conditions for an action.

Result: Conditions that are evaluated as a group are displayed on a single block of gray background, so you can identify them more easily.

- Configuring properties for use in engagement policies and proposition filters
- Frequently asked questions about eligibility, applicability, and suitability policies

Configuring properties for use in engagement policies and proposition filters

To make a property or When rule available for use in an engagement policy or proposition filter, define it as a Relevant Record for your primary context class, which by default is your Customer class.

Pega Customer Decision Hub

1. Log in to your application as a user with the *your application*:CDHAdmins access group.
2. Click **Configure Application Inventory Relevant Records** .
3. Specify your customer class and click **Add records**.
4. Add all properties which should be available for use as eligibility criteria.

Frequently asked questions about eligibility, applicability, and suitability policies

See the following topics for additional information, tips and tricks, and best practices.

Pega Customer Decision Hub

Can I define eligibility, suitability or applicability conditions on the action and treatment levels?

You can define engagement policies for individual actions. For more information, see Defining action engagement policies.

Engagement policies are not available at treatment level. Actions convey an outcome or value to the customer, whereas treatments are the way in which the customer receives the action (for example, the specific web ad for the action). If needed, you can create multiple treatments for the same channel, and then allow the Next-Best-Action adaptive models to discover which treatment works best for which customers, without the use of treatment-specific engagement policies.

Can I define eligibility, suitability or applicability conditions on the issue level?

In the current version of Pega Customer Decision Hub, this is not available.

I want to show a new offer to the customers who interacted with another offer. Can I do this with an engagement policy?

Yes, you can configure an engagement policy to qualify customers for a new action if they interacted with another action before, for example, if the customers clicked on a mortgage offer or ignored a mortgage offer a number of times. First, you need to create a strategy that uses the interaction history to filter customers who have previously engaged with the action. Then, add a condition to your engagement policy that uses the results of that strategy to identify the customers who qualify for the new action.

Design Patterns: For instructions on how to configure the strategy for this use case, see Defining engagement policies to qualify customers for an action based on their interaction with another action.

Prioritizing actions based on customer relevance and business priority

On the **Arbitration** tab of the Next-Best-Action Designer, you can define the AI controls used to decision on the Next-Best-Action for each customer. The customer data and insights help provide proactive contextual actions according to the customer's propensity, context, and other factors.

Pega Customer Decision Hub

The following controls can be used by Pega Customer Decision Hub to select the Next-Best-Action for a customer. Each control can be toggled on and off. If you toggle a control off, it is no longer used to arbitrate Next-Best-Actions.

Propensity
: Propensity is the likelihood of a customer responding positively to an action. The propensity can be calculated separately for every treatment. For example, if a customer is most likely to interact with a push notification, an action which has a push notification treatment is most suitable for that customer. It is also possible to apply propensity calculated to the action as a whole.

Context weighting
: Pega Customer Decision Hub uses context weighting to consider the situational context for each action. For example, if a customer contacts the call center to cancel their service package due to concerns about the quality of their mobile service, the Retention issue should take priority for that customer. In this example, the context weighting can be defined as:

  - **Keys** - **Intent**
  - **Value** - Cancellation
  - **Issue / Group** - **Retention / Mobile**
  - **Weighting (+/-)** - 100 %

Action value
: Action value assigns a financial value for each action and prioritizes high over low value. The base value is calculated with the **Calculate Business Value** strategy. The strategy uses a static **business value** property which you can enter on the Details tab of each action. The static values are used as input for the **CalculateValueExtension** strategy. By default, the **CalculateValueExtension** strategy does not have any calculation logic, and simply passes the static Value property directly to strategy results. To implement a custom logic for Value calculation, edit the **CalculateValueExtension** strategy.

Business levers > Action weighting
: For each individual action, you can specify a static **business weight (+/-)** property on the Details tab of the action. Actions with a higher weight are prioritized over those with a lower weight. If you do not want to consider action weighting when deciding on Next-Best-Action, set this toggle to off.

Business levers > Business purpose weighting
: You can specify a static **Weighting (+/-)** property for actions associated with a specific issue and group. Actions associated with higher-weight issues and groups are prioritized over those with a lower weight. If you do not want to consider business purpose weighting when deciding on Next-Best-Action, set this toggle to off.

- Applying the propensity
- Applying the context weighting
- Applying the action value
- Applying the business levers

Applying the propensity

Propensity is the likelihood of a customer responding positively to an action. To get the most accurate, granular results, configure Next-Best-Action to calculate propensity for each treatment. For example, if a customer is most likely to interact with a push notification, an action which has a push notification treatment is most suitable for that customer.

You can also calculate the propensity for each action, without taking treatments into account. This option is not recommended if you have treatments, but may be relevant if you did not create any treatments for your actions.

To ensure that only highly relevant actions and treatments are presented to customers, you can specify minimum propensity thresholds. Actions and treatments that do not meet these thresholds are not sent to customers.

Pega Customer Decision Hub
Before you begin: Define actions with corresponding treatments.

1. In the Pega Customer Decision Hub portal, click **Next-Best-Action Designer Arbitration** .
2. In the **Propensity** section, ensure that the **Apply propensity calculated for every treatment** option is selected.
3. Ensure that the toggle in the upper right corner of the **Propensity** section is set to on.
   If the toggle is inactive, Next-Best-Action no longer takes propensity into account when selecting the next best action for a customer.
4. **Optional:** In the **Predictions** section, select the prediction used to determine the propensity for actions and treatments.
   By default, Pega Customer Decision Hub uses the built-in *PredictActionPropensity* and *PredictTreatmentPropensity* predictions. You can instead use different predictions that you created in Prediction Studio. For more information, see Predicting outcomes.
5. **Optional:** In the **Threshold** section, specify one or more minimum propensity thresholds for the action or treatment.
   Actions and treatments that do not meet the propensity threshold are filtered out, so that customers receive only the most relevant messages.

Applying the context weighting

Context weighting allows Pega Customer Decision Hub to consider the situational context for each action. For example, if a customer contacts the bank to close their card account, the highest-priority action is ensuring that the customer is retained. Because of that, the Retention issue should take priority for that customer.

Pega Customer Decision Hub

1. In the Pega Customer Decision Hub portal, click **Next-Best-Action Designer Arbitration** .
2. Click **Edit**.
3. In the **Context weighting** section, click **Add weighting**.
4. From the **Keys** list, select the type of key which holds the value used to evaluate the context:

Type	Description
CurrentPage	Use the current page which the customer is viewing for context. For example, if a known customer is currently viewing a page about loans, knowing this context may raise the priority of an action which belongs to the Cross-sell/up-sell issue and Loans group.
PreviousPage	Use the last page which the customer viewed for context.
Intent	Use the customer's stated intent for context.
Event	Use a real-time event detected by Pega Marketing or Pega Customer Decision Hub. For example, if a customer's mobile data is nearly used up, knowing this context may raise the priority of an new data plan offer for that customer.
EventType	Use a real-time event detected by Pega Marketing or Pega Customer Decision Hub.

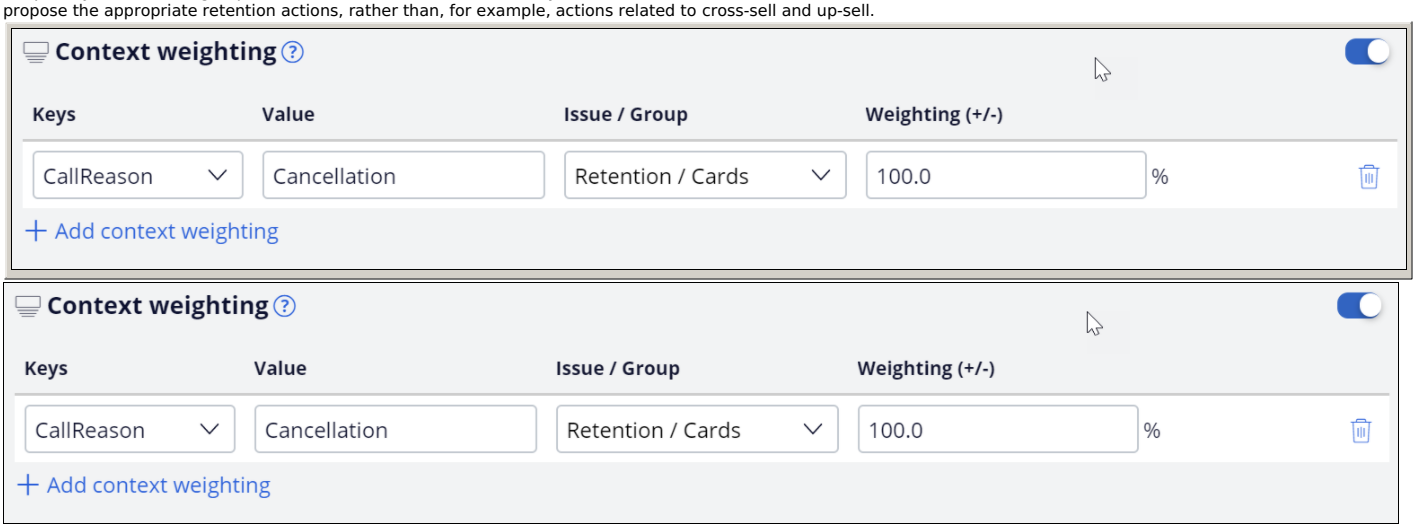
CallReason	Use the customer's stated call reason for context.

5. In the **Value** field, enter the value of the key, for example, Cancellation.
6. In the **Issue/Group** field, select the issue and group whose priority should be adjusted for the selected context.
7. In the **Weighting (+/-)** field, enter the value by which the priority should be adjusted.

   To reduce the priority, enter a negative value.

8. Ensure that the toggle in the upper right corner of the **Context weighting** section is set to **on**.
   If the toggle is inactive, the Next-Best-Action no longer takes context weighting into account when selecting the next best action for a customer.
9. Click **Save**.

Result: The priority of all actions in the specified issue and group is increased or decreased by the specified value when the selected context is detected. In the figure below, the priority of the Cards group within the Retention issue is raised by 100% if the customer calls the bank with the intention to cancel their card. Next-Best-Action can then propose the appropriate retention actions, rather than, for example, actions related to cross-sell and up-sell.

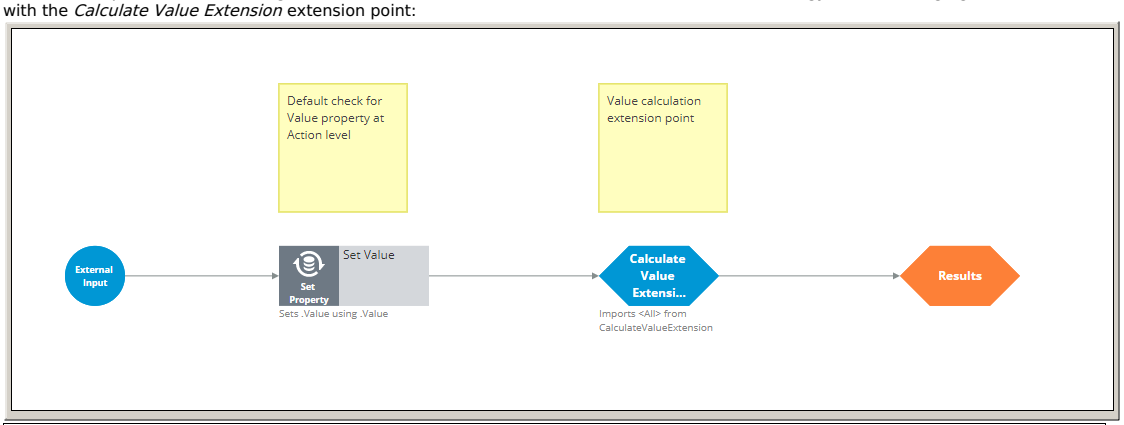


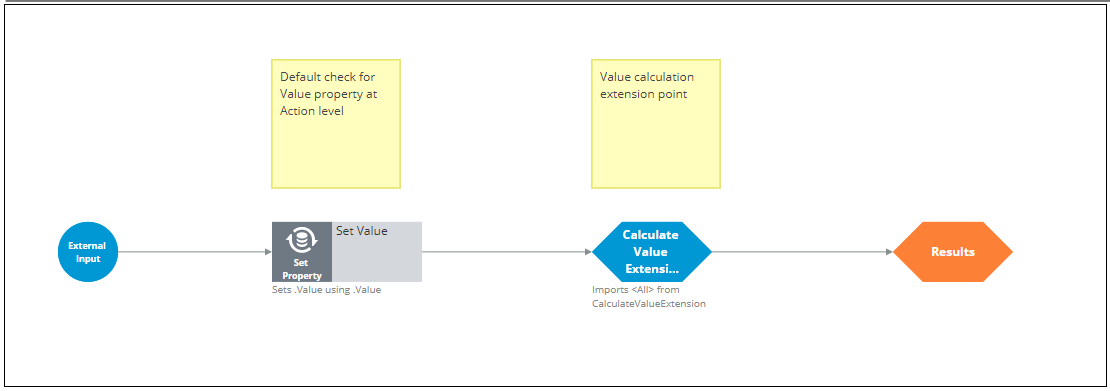
Applying the action value

Apply the action value to Next-Best-Action selection in order to assign a financial value for each action and prioritize high over low value.

Pega Customer Decision Hub
Before you begin: Create actions and specify their business values.

1. In the Pega Customer Decision Hub portal, click **Next-Best-Action Designer Arbitration** .
2. Click **Edit**.
3. Ensure that the toggle in the upper right corner of the **Action value** section is set to **on**.
   If the toggle is inactive, Next-Best-Action no longer takes action value into account when selecting the next best action for a customer.
4. **Optional:** Click **Base value calculation** to view or edit the *Calculate Business Value* strategy, which defines how action values are calculated.
   The strategy uses a static business value property which you can enter on the **Details** tab of each action. The static values are used as input for the *Calculate Value Extension* strategy. By default, the *Calculate Value Extension* strategy does not have any calculation logic, and simply passes the static *Value* property directly to strategy results. To implement a custom logic for *Value* calculation, edit the *Calculate Value Extension* strategy. The following figure shows the *Calculate Business Value* strategy with the *Calculate Value Extension* extension point:





Applying the business levers

For each individual action, you can specify a static business weight property on the **Details** tab of the action, and then use this property to prioritize actions with a higher weight over those with a lower weight. In addition to action-specific weighting, you can also specify a static *Weighting (+/-)* property for actions associated with a specific issue and group.

Pega Customer Decision Hub
Before you begin: Create actions and specify their business weights.

1. In the Pega Customer Decision Hub portal, click **Next-Best-Action Designer Arbitration** .
2. Click **Edit**.
3. Ensure that the toggle in the upper right corner of the **Business levers Action weighting** section is set to **on**.
   If the toggle is inactive, Next-Best-Action no longer takes action-specific weighting into account when selecting the next best action for a customer.
4. **Optional:** Ensure that the toggle in the upper right corner of the **Business levers Business purpose weighting** section is set to **on**, and then click **Add weighting**.
5. In the **Issue/Group** field, select the issue and group whose priority should be adjusted.
6. In the **Weighting (+/-)** field, enter the value by which the priority should be adjusted.

Result: The priority of actions is always increased or decreased by the value specified in the business weight property and in the **Business purpose weighting** section of the Next-Best-Action Designer. For more information about the underlying strategy that calculates the final priority, see Arbitration strategy.

Enabling channels and defining triggers

Select the channels which you use to communicate with your customers, as well as the triggers that cause Pega Customer Decision Hub to decision on Next-Best-Actions.

Pega Customer Decision Hub

- Understanding channels
- Enabling channels
- Adding additional channels to Next-Best-Action Designer
- Configuring additional channels for use with volume constraints
- Defining real-time container triggers
- Defining real-time event triggers
- Configuring a recurring schedule for Next-Best-Action
- Defining triggers for paid media
- Frequently asked questions about channels

Understanding channels

Channels are a means of communication that customers can use to interact with a company. These communications can be outbound (allowing you to send out messages to customers) or inbound (allowing customers to contact you or otherwise respond to your brand). Pega Customer Decision Hub includes support for the following channels:

Pega Customer Decision Hub

Channel	Direction	Additional comments
Call center	Inbound	
Email	Outbound	
Mobile	Inbound	
Other	N/A	Use the Other channel to manage communications that are not covered by any of the predefined channels.
Paid	N/A	Use the Paid channel to enable advertising on paid destinations, such as Facebook Ads, Google Ads, and others. For more information, see Extending next-best-action to digital advertising platforms with Paid Media Manager.
Push notification	Outbound	
Retail	Inbound	
SMS	Outbound	
Web	Inbound	

You can enable or disable any of the individual channels. If a channel is toggled off, its status is stored in a decision data rule (DDR) and the related part of the strategy framework is turned off. Setting the toggle to on reactivates the related part of the strategy framework.

Enabling channels

Use the on/off toggles to indicate which inbound and outbound channels are active and should be considered for Next-Best-Action.

Pega Customer Decision Hub

1. In the **Channels** tab of the Next-Best-Action Designer, in the **Channels** section, review the active toggles.
2. If you want to disable a channel for Next-Best-Action, set the related toggle to OFF.
   If a channel is toggled off, its status is stored in a Decision decision data rule (DDR) and the related part of the strategy framework is turned off. Setting the toggle to on reactivates the related part of the strategy framework.

Adding additional channels to Next-Best-Action Designer

In addition to the built-in channels provided by Pega Customer Decision Hub, you can implement additional channels by extending the *ChannelSettings* decision data rule. For example, if you send out physical product catalogs to your customers, you can create a Direct Mail channel for these types of treatments.

Pega Customer Decision Hub

1. In Dev Studio, click **Records Decision Decision data** .
2. Open the *ChannelSettings* decision data rule that resides in the *Data-Decision-Request-Customer* class.
3. Click **Check out**.
4. Click **New**.
5. Configure the channel details:
   - **Name**- For example, Direct Mail.
   - **Description**- An optional description of the channel, for example, Used for product catalogs related to the Household line.
   - **Channel icon**- An optional icon to visually identify the channel in Next-Best-Action Designer. Available icons include **pi-forward**, **pi-mail**, **pi-picture**, **pi-mobile-phone**, **pi-notification**, **pi-money**, **pi-headset**, and **pi-chat-typing**.
   - **ModelName**- Enter NBA.
   - **SettingName**- A unique identifier for the channel, for example, DirectMail.
   - **SettingValue**- Set to true to enable the new channel, or false to create the channel without enabling it.

     For more information about enabling channels, see Enabling channels.
6. Save and check in your changes.

Result: The new channel now appears in the **Channels** tab of Next-Best-Action Designer. If it is enabled, you can also use it when configuring contact policies in the

Constraints tab of Next-Best-Action Designer.

- Creating treatments for additional channels

Creating treatments for additional channels

After creating the new channel in Next-Best-Action Designer, you can configure treatments for that channel.

Pega Customer Decision Hub

1. In the Pega Customer Decision Hub portal, click **Content Treatments** .
2. Click **Create Other** .
3. In the **Channel** list, select the new channel that you created.
4. Fill out all other fields required to configure the treatment.

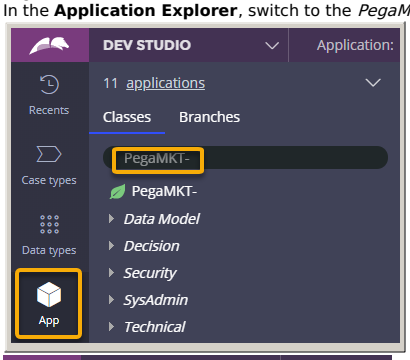
What to do next: For more information about treatments, see Defining treatments for actions.

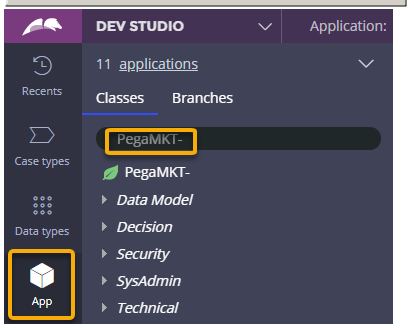
Configuring additional channels for use with volume constraints

Volume constraints limit the number of actions which are offered to customers, for example, to comply with corporate or regulatory guidelines. Volume constraints can apply per action, per action property, or per channel. If the default channels do not fulfill your requirements, you can create additional channels for use with volume constraints.

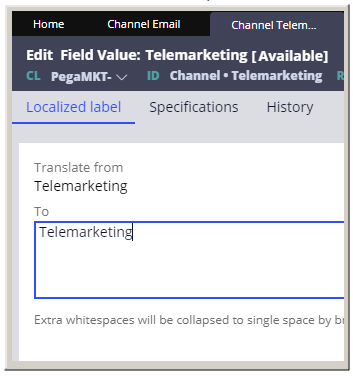
Pega Customer Decision Hub

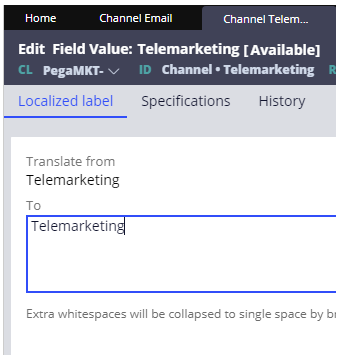
1. Log in to Dev Studio.
2. In the **Application Explorer**, switch to the *PegaMKT-* class, as in the following figure:



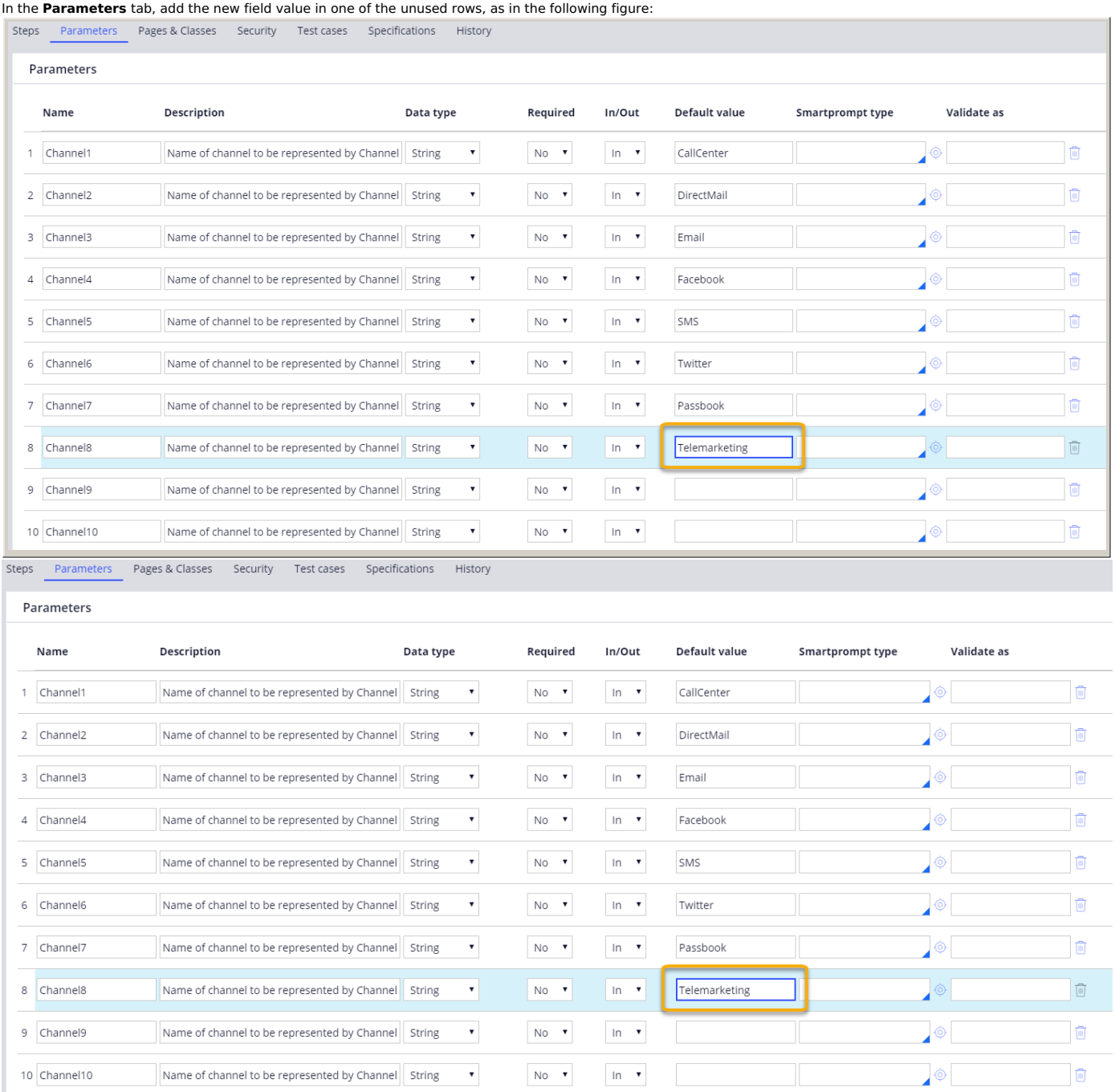


3. Click **Data Model Field Value** .
4. Open a field value, for example, **Channel Email**.
5. Click **Save as**.
6. Enter a label for the new channel, for example, Telemarketing.
7. In the **Add to ruleset** list, select the ruleset for your custom rules. By default, this is *<your implementation application>-Rules*.
8. Click **Create and open**.
9. Click **Check out**.
10. In the **Translate To** field, enter the name of the new channel, as in the following figure:





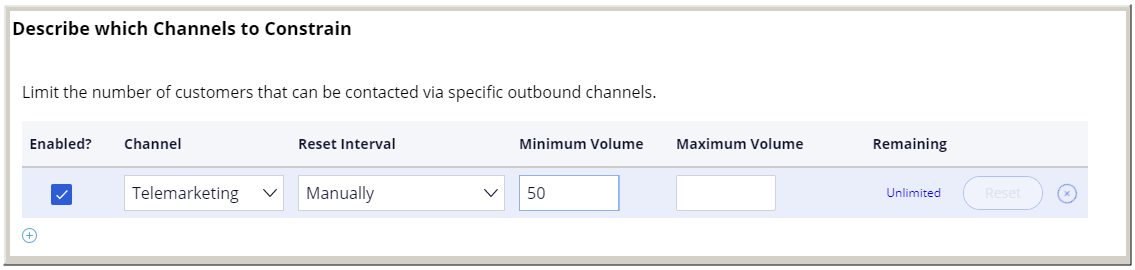
11. In the **History** tab, in the **Description** field, enter a new description for the channel, for example, Field value for the Telemarketing channel.
12. Save and check in your changes.
13. In the **Application Explorer**, switch to the *PegaMKT-System-Setting* class.
14. Click **Technical Activity** .
15. Open the **SetChannelPropertyMappings** activity.
16. Click **Private edit**.
17. In the **Parameters** tab, add the new field value in one of the unused rows, as in the following figure:



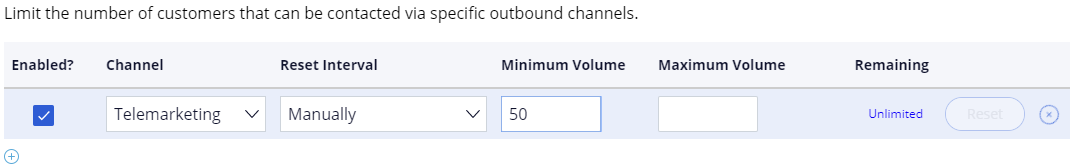
Note: The parameter value is case-sensitive.

18. Click **Save**.
19. Click **Actions Run** .

Result: You can now use the new channel in your volume constraints and contact policies, as in the following figure:



Describe which Channels to Constrain



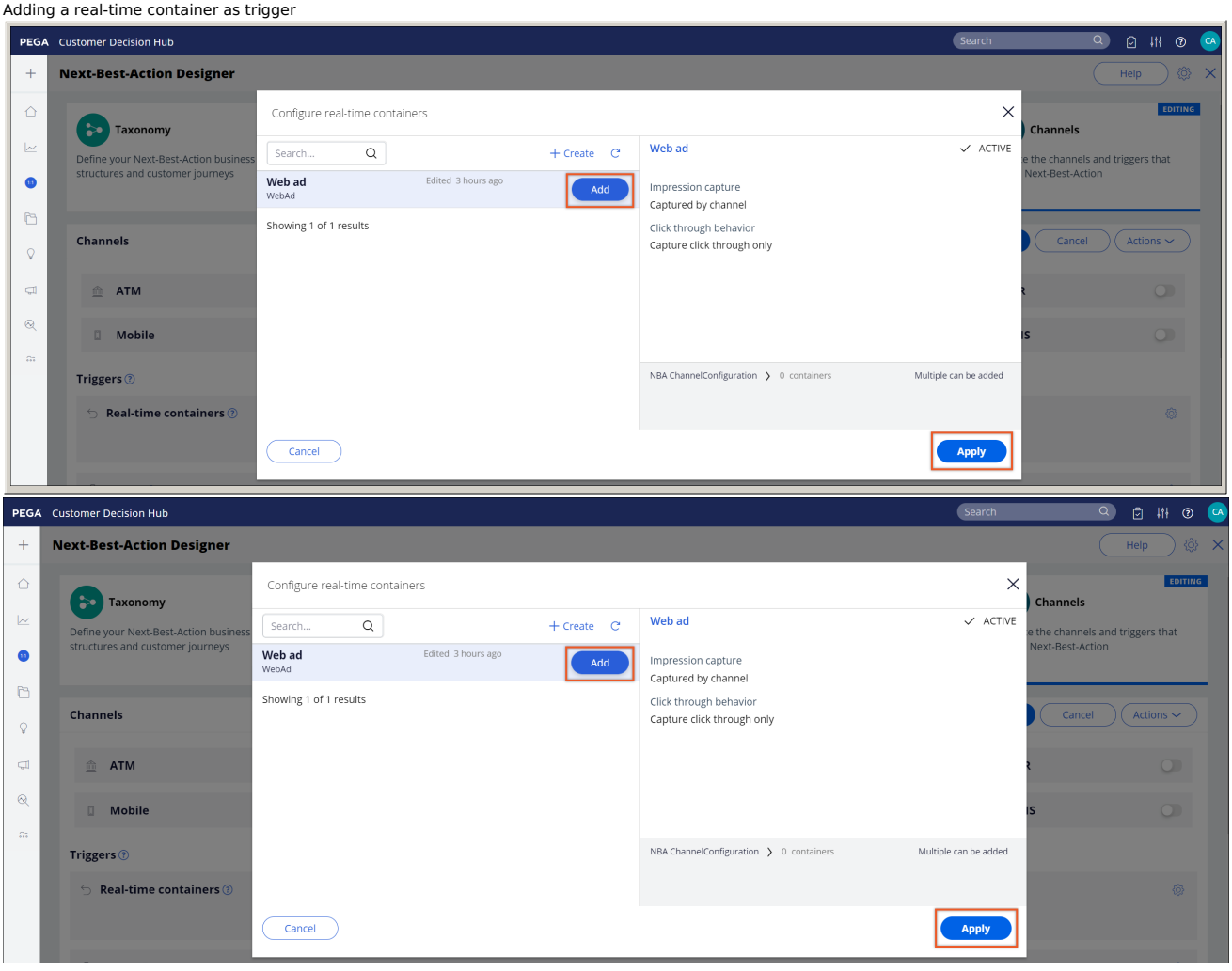
Defining real-time container triggers

Real-time containers manage content in real-time channels. Adding a real-time container as a trigger enables it to be called by an external channel. When a container is loaded, for example, when an individual views an ad for an action in a predefined placement on the web channel, it triggers the next best action. For more information, see Managing content in real-time channels with real-time containers.

Pega Customer Decision Hub

1. Define a real-time container. For more information, see Managing content in real-time channels with real-time containers.
2. In the **Channels** tab of the Next-Best-Action Designer, click **Edit**, and then in the **Triggers Real-time containers** section, click **Configure** icon.
3. Select the real-time container that you want to add as a trigger and click **Add Apply** .

Adding a real-time container as trigger



4. In the **Business structure level** list, associate the container with the business issue and group.
   Note: Before adding real-time containers as triggers, work with your system architect to make sure your the design time configuration of your access group is pointing to highest (and only) unlocked artifacts ruleset version.
5. Click **Submit**.
6. Click **Save**.

Defining real-time event triggers

Define real-time events to respond to real-time customer engagement opportunities triggered by an external event. For example, you can define a real-time event which triggers Next-Best-Action after an individual signs up for a loyalty card with your retail store. For more information, see Responding to real-time customer engagement with real-time event triggers.

Pega Customer Decision Hub

1. Define a real-time event. For more information, see Responding to real-time customer engagement with real-time event triggers.
2. In the **Channels** tab of the Next-Best-Action Designer, in the **Triggered by Events** section, click **Configure**.
3. Select the event that you created.
4. In the **Business structure level** list, select the business issue and group with which the event is associated.
5. Click **Submit**.
6. Click **Save**.

Configuring a recurring schedule for Next-Best-Action

Configure a recurring schedule for running Next-Best-Action using an audience that represents all customers and a daily schedule. The outbound schedule initiates any outbound actions, for example, campaigns or audience updates on external advertising platforms.

Pega Customer Decision Hub
Note: This schedule applies across all issues and groups in your business hierarchy, and uses the top level customer context. If you want to define a schedule specific to a different context, or a particular issue or group, see step 7.

1. In the **Channels** tab of the Next-Best-Action Designer, in the **Primary schedule > Recurrence** section, set up a recurring schedule, start date, and end date for outbound Next-Best-Action.
2. To select the starting population, in the **Outbound schedule Starting population** section, click **Configure**.
   Note: The starting population for Next-Best-Action is based on the top level customer context. For more information about configuring segments, see Defining customer audiences with customer segments. In the current version of Pega Customer Decision Hub, prospect segments cannot be used as starting populations for Next-Best-Action.
3. To apply constraints to actions and avoid overexposure, specify the number of times that an action can be offered to a group of customers, in the **Volume constraint** section, click **Configure**. For more information about volume constraints, see Avoiding overexposure of actions with volume constraints.
4. Click **Save**.
   Note: To review and adjust the outbound schedule, click **Actions Open outbound schedule** . This opens the underlying multi-channel campaign strategy. This action is restricted to administrator-type operators.
   Result: The outbound schedule configuration is created in Draft mode.
5. To launch the outbound schedule, click **Actions Run** .
   The Run action is only available after you configure the schedule.
6. To suspend an active outbound schedule, click **Actions Suspend** .
7. **Optional:** In addition to the primary schedule defined for your main customer context, you can define secondary schedules to run your marketing operations for different customer groups or contexts.
   a. In the **Channels** tab of the Next-Best-Action Designer, click **Actions Add schedule** .
   b. In the new **Additional schedule** section, in the **Issue / Group** list, select the issue and group or groups to which the schedule applies.
      Note: As a best practice, you should launch all outbound actions at the top level of your Next-Best-Action Designer hierarchy. However, if required, you can select specific issues and groups to define additional outbound schedules that run your marketing operations at different times for different levels of your business hierarchy.
   c. In the **Context** list, select the customer context that you want to target. **For example:** For example, when targeting the customers at the account level, you might use a context like Household or Account, depending on the settings that you configured in the Context Dictionary.
      Note: Choosing context other than top level customer context will result a warning, as using contexts below primary will impact engagement policies and might result in action failures.
   d. Select the starting population and optionally also the volume constraint, as described in steps 2 and 3.

Result: After an outbound run is launched, you can click the **Run details** link to view details about the campaign data flow. For more information about run details and scheduled runs, see Viewing Run Details and Run schedule.

Defining triggers for paid media

For consistent results across owned and paid channels, you can use Next-Best-Action to drive customer engagement on external media platforms such as Facebook Ads and Google Ads. Using Next-Best-Action with paid media ensures that each individual in an audience is targeted with the optimal offers at the right priority and maximum bid amount.

Pega Customer Decision Hub
For more information, see Triggering paid updates through Next-Best-Action.

Frequently asked questions about channels

See the following topics for additional information, tips and tricks, and best practices.

Pega Customer Decision Hub

Can I add or remove channels?

You can edit the decision data rule to add or remove channels if required. For an example, see Configuring additional channels for use with volume constraints.

You can also use the Other channel type for any channel which is not covered by Pega Customer Decision Hub out of the box.

How many Other channels can I add?

There is currently no limit to the number of channels that you can add.

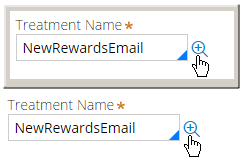
Defining treatments for actions

A treatment represents how the action is presented and experienced by a customer in a specific communication channel. You can define treatments for a number of built-in and custom channels, such as email, web, and others.

Pega Customer Decision Hub

Once created, treatments are associated to a specific action on the **Treatments** tab of the action rule. Treatments added to the **Treatments** tab of the action will be prioritized by the next-best-action strategy, which uses adaptive models to determine the best channel and treatment. This is done by continuously testing the treatment success rate, by randomly increasing the propensity of certain treatments to allow the model to explore and learn about other situations that could yield positive results. If you use Next-Best-Action Designer, you do not need to manually test the effectiveness of your treatments. For more information, see Model learning for new actions.

The **Flow** tab of the action should utilize the **Use associated treatment** option on the appropriate shape in order to take advantage of the best treatment based on the strategy framework. To create a new treatment, the user simply needs to type in the desired name and click the magnifying glass to launch the **Create** form for the treatment.

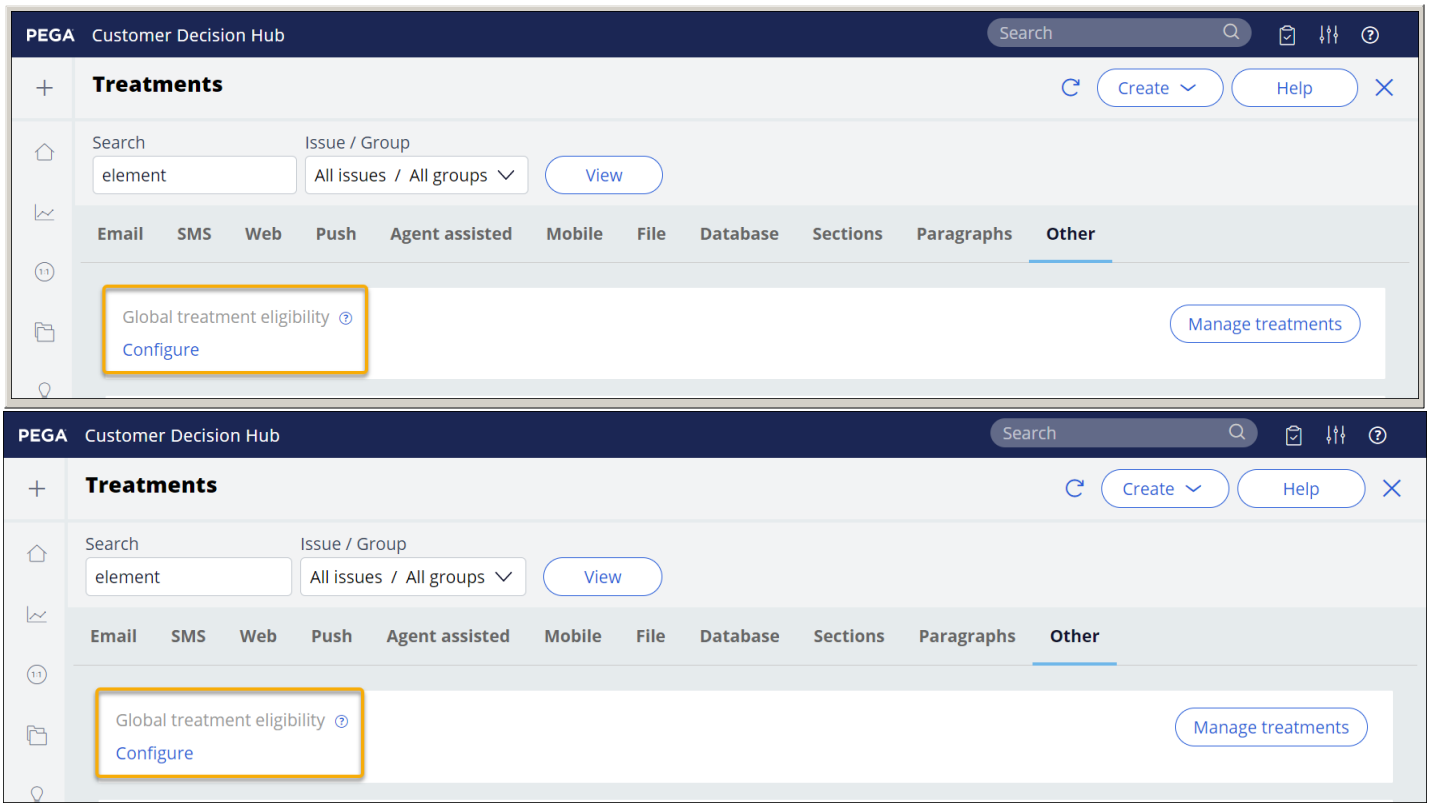


Treatments can be defined at the top level or can be categorized into issues and groups. The system automatically selects the issue and group in the Create form using the

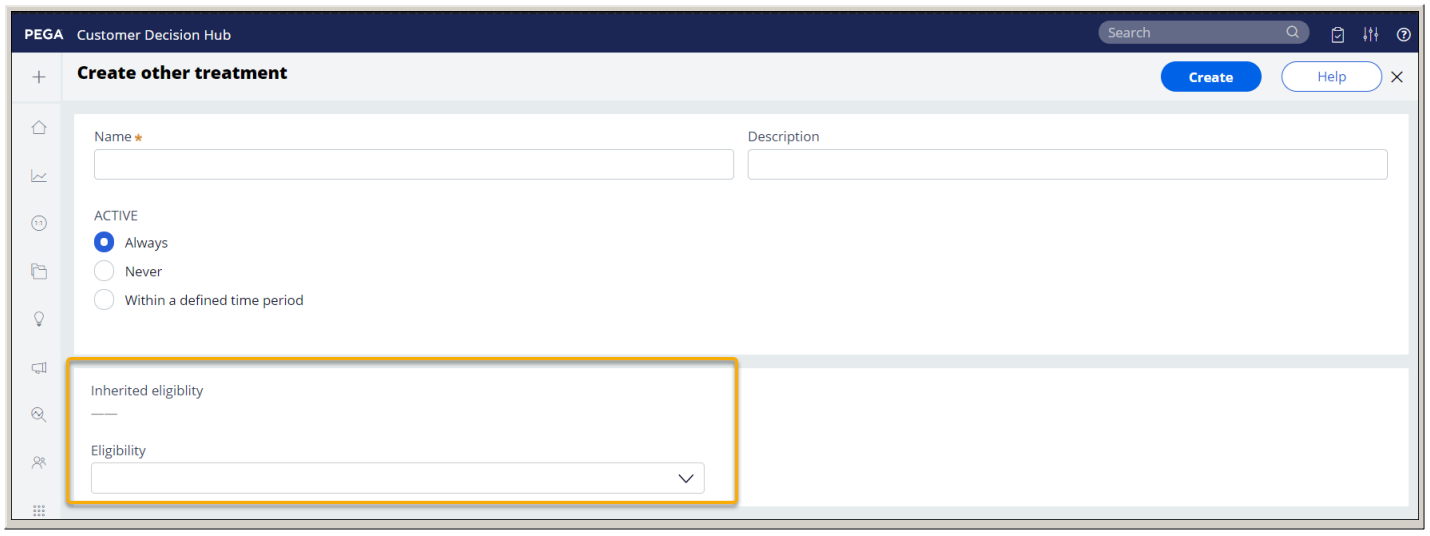
corresponding values from the action rule. Users can change these selections to store the new treatment at a different hierarchical level.

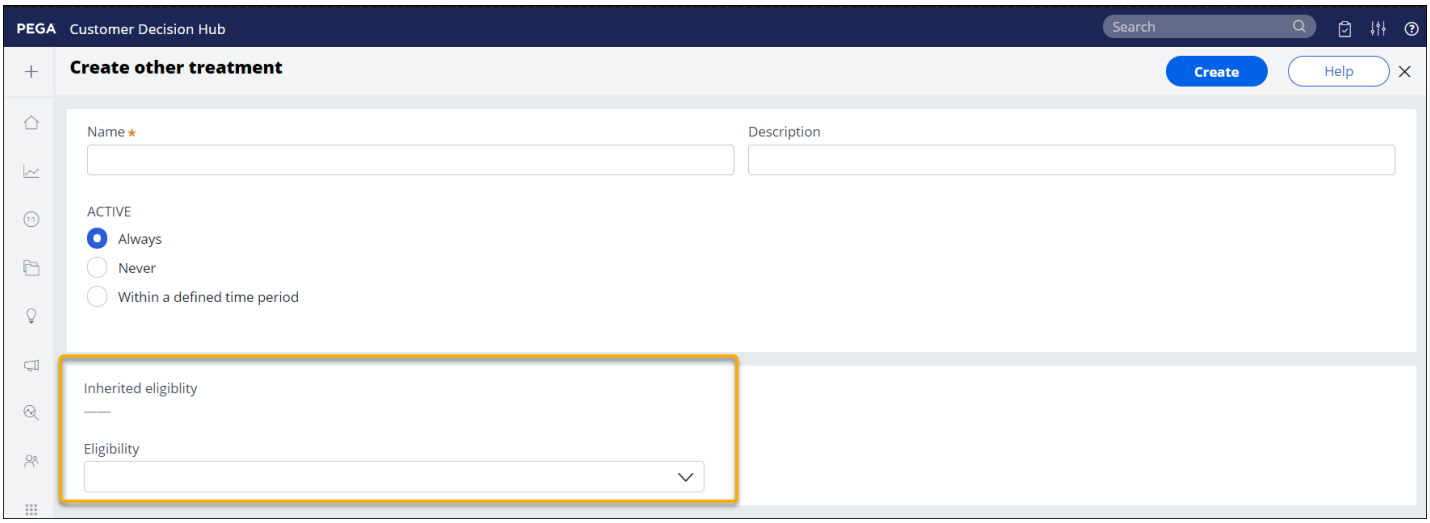
Eligibility conditions for treatments

You can specify the conditions under which a particular treatment or treatment type should be made available to a customer. You can apply both global (inherited) and individual eligibility conditions. Global treatment eligibilities are applied to all treatments of a specific type, as in the following figure:



Individual eligibility conditions are applied to a specific treatment, in addition to any global eligibility conditions that you specified.





Creating eligibility conditions

Eligibility conditions are defined as When rules associated with specific customer contexts. Work with your implementation specialist to define the required eligibility conditions for your treatments. For more information, see Creating treatment eligibility criteria.**Tip:** Do not use treatment eligibility rules for targeting treatments to customers. Treatment targeting is better handled by the Arbitration process. Treatment eligibility is intended for eligibility rules such as Treatment is in a customer's preferred language, customer has opted into this channel, customer has a downloaded the mobile app, and so on.

- Configuring email treatments
- Configuring SMS treatments
- Configuring push notification treatments
- Configuring web treatments
- Configuring mobile treatments
- Configuring agent-assisted treatments
- Configuring other channel treatments
- Sections
- Adding custom attributes to treatment form

Configuring email treatments

The **Subject** field allows the user to specify the subject for the email. The email subject can be static text, a property reference, or an expression which combines both static text and property references.

Pega Customer Decision Hub Pega Customer Decision Hub Pega Customer Decision Hub Pega Customer Decision Hub Pega Customer Decision Hub Pega Customer Decision Hub

The user can also utilize the **Key Code** field to specify a code for the Email Treatment. This code can be used by downstream functionality, such as the Send Email shape in the action canvas.

When the **Track Email Open Rate** checkbox is enabled, the system automatically captures email opens (impressions) in Interaction History. This setting is enabled by default in a new Email Treatment.

The Email Treatment Editor enables the user to design high quality treatments to send to customers. This editor supports two modes of editing:

- Design – This is the default editing mode. In this mode, users can enter free-form content and use a wide array of editing tools to format this content. To apply formatting to content; select the content first and then, use the formatting icons to update the content's format.
- Source – This mode is for advanced users and provides direct access to the source HTML for this treatment. In this mode, the formatting icons are disabled.

Users can switch between the two modes above by toggling the **Source** button:



The editor also includes an array of buttons to insert content (such as links, images, properties, etc.) into the treatment:



Two of the buttons in this array are essential in designing effective customer-focused treatments. These are further outlined below.

Extending the email treatment form with custom properties

If the default list of fields available on the email treatment form does not contain a property which you require, edit the *EmailTreatments* Decision Data rule used by the treatments.

For more information, see Managing Decision Data rules in Pega Customer Decision Hub. Any new properties you add to the form are automatically available for all previously created treatments.

Embedding Relevant Data

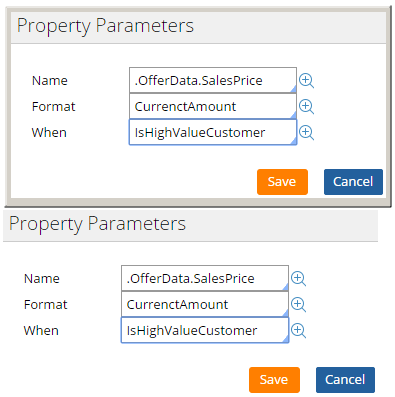
Relevant data can be embedded into the treatment via the Insert Property button:




Examples of this data include customer information, action details, etc.

Upon clicking this button, the **Property Parameters** dialog can be configured to embed the necessary data field. In addition to specifying the name of the field to be embedded, users can also specify:

- **Format** – Reference to a Format rule that determines how this field should be formatted.
- **When** – Reference to a When rule that determines whether this field should be displayed or not.



Note: When an SMS message contains embedded fields, the character count is an approximation of the message length. Each embedded field counts as 15 characters towards the character count for the message.

Embedding a Response Link

Response links can be embedded into an Email Treatment via the Insert Button button:



Clicking this button opens the **Button Parameters** dialog. This dialog presents the user with a variety of options for configuring the response mechanism for the treatment. The type of response link is determined by the value selected in the **Function** drop-down. The following options are available:

- **Accept action** - This option embeds a link for the customer to accept the action. The user must select whether to direct the customer to a response screen or to an external URL.
- **Decline action** - This option embeds a link for the customer to decline the action. The user must select whether to direct the customer to a response screen or to an external URL.
- **Open internal page** – This option embeds a link to launch an internal page. The user must specify the name of the response screen (internal page). **Note:** This page cannot be customized with customer or action data, because internal pages do not support referencing this type of data. Use the **Open external URL** or **Microsite** options instead.
- **Open external URL** – This option embeds a link to launch an external URL. The user must specify the external URL to open.
- **Microsite** – This option embeds a link to launch a microsite. The user must select the microsite from a list of all microsites configured in the system.

Tracking Click Rate

For each of the above functions, the user can enable the **Track click rate** option. The system automatically tracks clicks (in Interaction History) for any response links that are generated with this option enabled. This setting is enabled by default.

Redirecting to a Response Screen

This option is available for the Accept Offer, Decline Offer, and Open Internal Page links. The user can configure a response screen to be displayed when the customer clicks on a response link in the action email. This response screen is an internal page in the application and is typically configured using a Section rule.

- For Accept Offer and Decline Offer, the response screen value is defaulted to the value configured by the application administrator. In these cases, a typical response screen should represent a confirmation for the action taken by the customer (accept or decline) and any other information that needs to be presented.
- For Open Internal Page, the response screen can be considered as a page to provide more information on the action. This page can contain appropriately configured action buttons (such as accept and decline) that the customer can use to respond to the action.**Note:** This page cannot be customized with customer or action data, because internal pages do not support referencing this type of data. Use the **Open external URL** or **Microsite** options instead.

If the value for the response screen is not set, the out-of-the-box default response screen (DefaultEmailResponse) will be used. Both customer and action data are available to be referenced when building and configuring a custom response screen.

Note: Response screens must be created in the PegaMKT-Work-Offer class.

Redirecting to an External URL

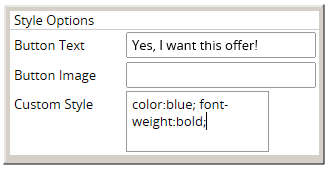
This option is available for the Accept Offer, Decline Offer, and Open External URL links. The user can further choose whether relevant Offer Data should be included while re-directing to the external URL. If selected, these values are passed as URL parameters to the external URL being launched. The following parameters are furnished:

- C – Customer ID
- R – Customer response
- I – Issue of the action
- G – Group of the action
- O – Name of the action
- T – Name of the Email Treatment
- W – ID of the Campaign Runs that initiated the action
- RA – Rank of the action in the Campaign Run (batch output) table
- OU – Name of the Campaign Run (batch output) class containing the action instance

Adding Style Options

The following style options can be specified for the response link:

- **Button text** – The text that is displayed to the user as the link
- **Button image URL** – Reference to an image that is displayed alongside the link
- **Custom style** – CSS formatting style that should be applied to the link



Style Options
Button Text
Yes, I want this offer!
Button Image
Custom Style
color:blue; font-weight:bold;

Previewing Included Sections

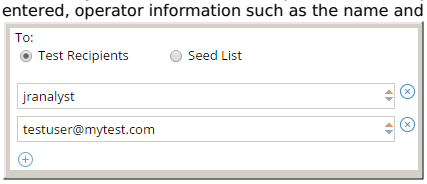
The **Show the preview of included section** checkbox (at the bottom of the Email Treatment editor) enables users to preview sections inline in the editor.

Testing the Email Treatment

The **Test Message** tab on the Email Treatment provides an easy way to test the look and feel of the Email Treatment when it is actually delivered.

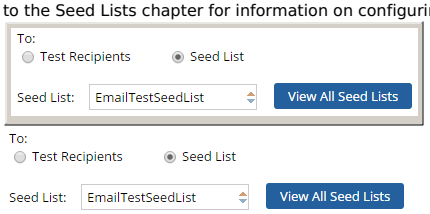
The following options are available for configuring the test message:

- **To** – Recipients of the test email. Select either Test Recipients or Seed List.
  - **Test Recipients** - Enter list of recipients. Each entry can either be the Operator ID of an existing user in the system or a valid email address. If the operator ID is entered, operator information such as the name and email is populated as the customer data and the test email is sent to the operator's email address.



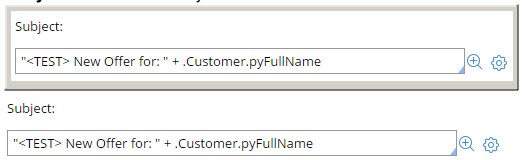
To:
Test Recipients
Seed List
jranalyst
testuser@mytest.com

  - **Seed List** - Select a predefined Seed List as the intended recipients of the test email. The data for each seed is populated as the customer data and is visible (if referenced) in the test email. In this case, the **Email 1** column in the Seed List (.pyEmail1 property) specifies the email address to which the test email is sent. Refer to the Seed Lists chapter for information on configuring a Seed List.



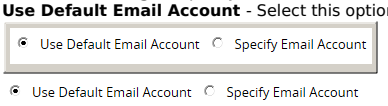
To:
Test Recipients
Seed List
Seed List:
EmailTestSeedList
View All Seed Lists

- **Subject** – Enter the subject for test email.



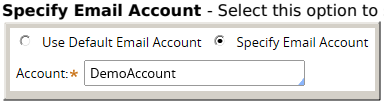
Subject:
"<TEST> New Offer for: " + .Customer.pyFullName

- **Test Account Settings** – Specify Email Account settings for the test email.
  - **Use Default Email Account** - Select this option to use the Default outbound Email Account.



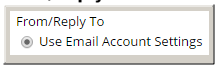
Use Default Email Account
Specify Email Account

  - **Specify Email Account** - Select this option to specify the outbound Email Account to use for delivering the test email. Specify the account name in the **Account** field.



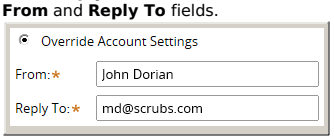
Use Default Email Account
Specify Email Account
Account:*
DemoAccount

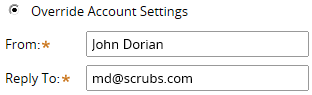
  - **From/Reply To**: **Use Email Account Settings** - Select this option to use the sender's name and reply-to address that is configured on the Email Account being used.



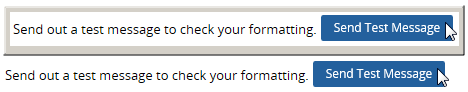
From/Reply To
Use Email Account Settings

  - **From/Reply To**: **Use Operator Settings** - Select this option to use the name and email address of the current operator for these values.
  - **From/Reply To**: **Override Account Settings** - Select this option to specify the sender's name and reply-to address for the test email. Specify these values in the **From** and **Reply To** fields.

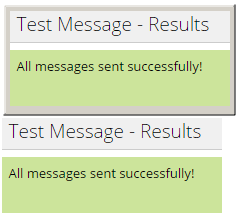




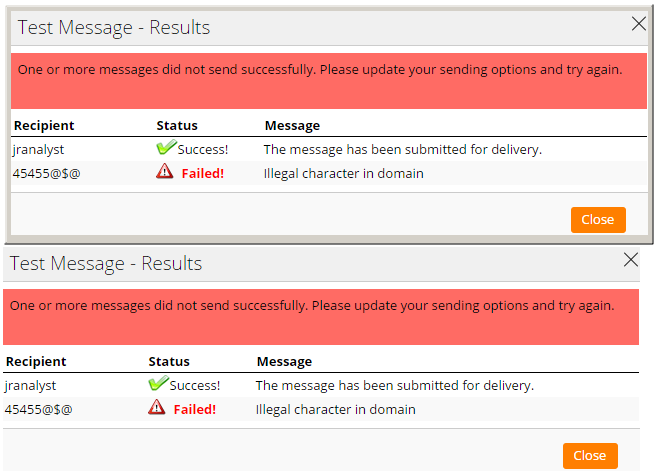
After the test message has been configured, it can be sent via the **Send Test Message** button.



Upon clicking this button, the Email Treatment is saved and the test email is sent to the specified recipients. The status of the test is displayed.



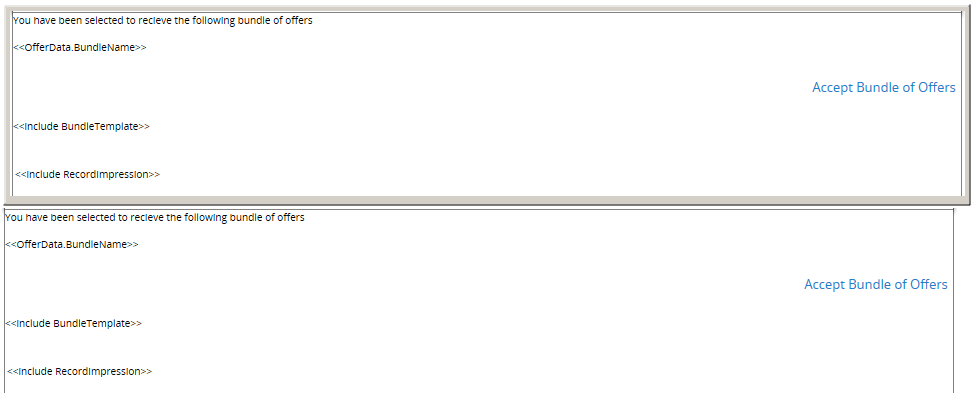
Any errors in the test delivery are displayed in case of a failure. These can be corrected and the test email can be resubmitted for delivery.



Configuring email treatments for actions which are grouped together as a bundle

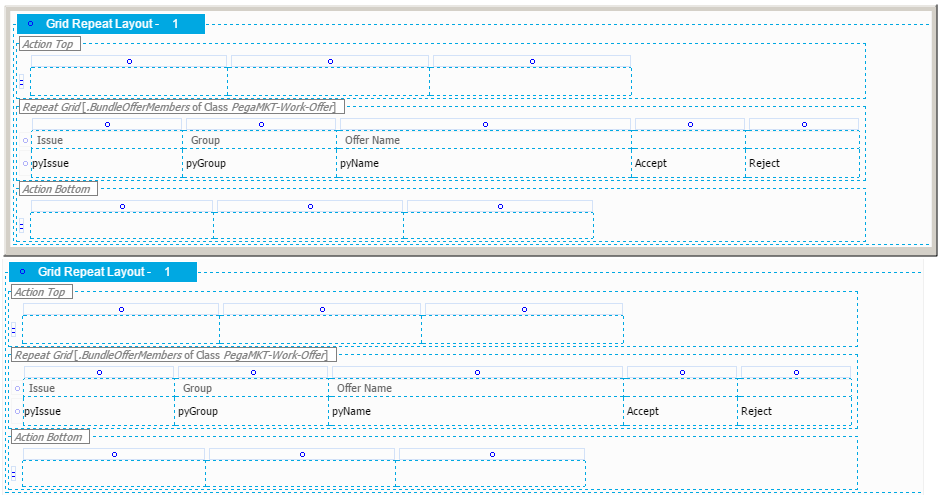
When designing Email Treatments for bundles, the user needs to consider the type of the bundle. A typical Email Treatment for an action bundle includes an overall description of the bundle and details for each of its members. Additionally, based on the bundle type, response links can be provided at the bundle and/or the member level.

The top level BundleTemplate Email Treatment is provided as an out-of-the-box sample.



This treatment includes the following salient pieces of a bundle Email Treatment:

- Bundle name - References the .OfferData.BundleName property
- Response link for bundle – Added using the Insert Button functionality
- Impression capture - References the RecordImpression section. With bundles, it is typical to include the impression capture section at the parent level.
- Member details - References the BundleTemplate section. This section uses the BundleOfferMembers page list property to display the member actions in a grid with accept and reject response links for each member.



The BundleOfferMembers property is populated at runtime to contain details of the various bundle members. This can be used in dynamic containers such as the repeating grid. Users can also make static references to member actions in this page list. For example, to display the name of the first action in the bundle, reference the .BundleOfferMembers(1).OfferData.pyName property.

Another common usage pattern is to update the grid in the BundleTemplate (or equivalent) section at the top level (PegaMKT-Work-Offer) to include a sub-section for displaying relevant action information, say MemberOfferDetails. Users can then create a top-level version of this section to serve as a base for all member actions. To customize the details for a particular issue or group, the user simply needs to override the details section in the appropriate issue or group class.

Identifying customers who open or respond to email treatments

Identity matching occurs when users open and respond to marketing emails sent via the application.

The following associations are captured when a customer opens a marketing email (handled by the IMPR service):

- CustomerID to UniqueID
- CustomerID to Email Address
- CustomerID to IP Address
- UniqueID to Email Address
- UniqueID to IP Address

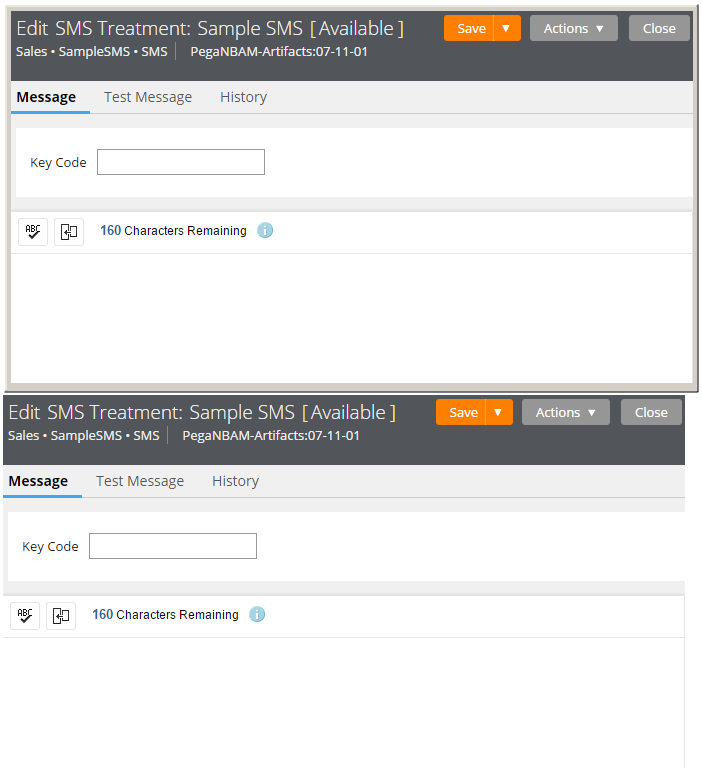
The following associations are captured when a user clicks on a response link in a marketing email (handled by the PORE service):

- CustomerID to UniqueID
- CustomerID to IP Address
- UniqueID to IP Address

Configuring SMS treatments

A newly created SMS Treatment starts out with an input field for the key code and a blank canvas to compose the SMS message, as shown below.

Pega Customer Decision Hub Pega Customer Decision Hub Pega Customer Decision Hub Pega Customer Decision Hub



The SMS Treatment rule includes the following tabs:

Tab Name	Purpose
Message	Create the contents of the Treatment. Refer to Using the SMS Treatment Editor.
Test Message	Test the content and format of this Treatment by sending it. Refer to Testing the SMS Treatment.
History	Specify the Description and Usage information for this Treatment.

Extending the SMS treatment form with custom properties

If the default list of fields available on the SMS treatment form does not contain a property which you require, edit the *SMSTreatments* Decision Data rule used by the treatments.

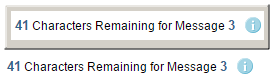
For more information, see Managing Decision Data rules in Pega Customer Decision Hub. Any new properties you add to the form are automatically available for all previously created treatments.

Using the SMS Treatment Editor

The SMS Treatment editor enables the user to specify the contents of the SMS message. Users can simply type the content of the message into the editor.

The SMS editor provides the following features:

- Character count - As a user types the message content, the editor displays the number of characters in the message. Once the content goes past the length of one SMS message (160 characters), the editor also displays the message number along with an informational note to the user about SMS message lengths.

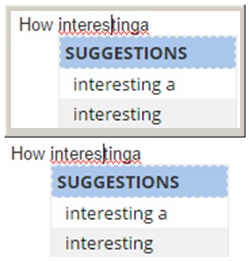


41 Characters Remaining for Message 3

- Spell check – Users can use perform a spell check via the Check Spelling icon:



Spelling errors are underlined in red. Clicking an incorrectly spelled word provides a list of possible corrections. Selecting one of these corrections replaces the incorrect word with the selected suggestion.



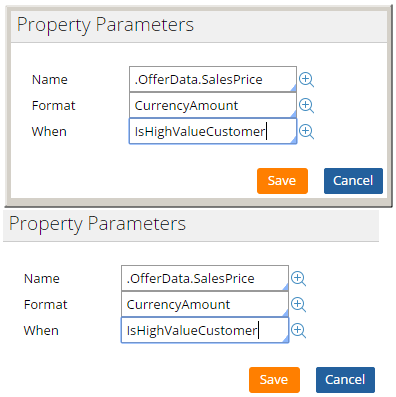
How interestinga
SUGGESTIONS
interesting a
interesting

Embedding Relevant Data

Relevant data (such as customer information, action details, etc.) can be embedded into an SMS Treatment via the Insert Property icon.

Upon clicking this icon, the **Property Parameters** dialog can be configured to embed the necessary data field. In addition to specifying the name of the field to be embedded, users can also specify:

- **Format** – Reference to a Format rule that determines how this field should be formatted.
- **When** – Reference to a When rule that determines whether this field should be displayed or not.



Property Parameters
Name
.OfferData.SalesPrice
Format
CurrencyAmount
When
IsHighValueCustomer
Save
Cancel

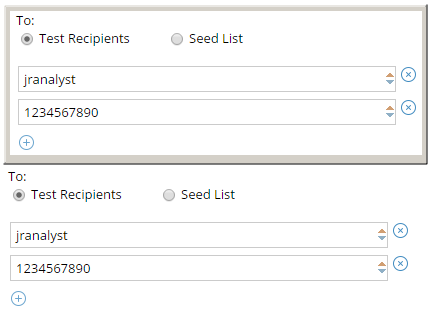
When an SMS message contains embedded fields, the character count is an approximation of the message length. Each embedded field counts as 15 characters towards the character count for the message.

Testing the SMS Treatment

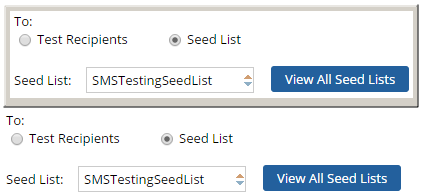
The **Test Message** tab on the SMS Treatment provides an easy way to test the content of the SMS Treatment when it is actually delivered.

The following options are available for configuring the test message:

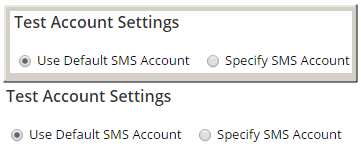
- **To** – Recipients of the test message. Select either Test Recipients or Seed List.
    - **Test Recipients** - Enter list of recipients. Each entry can either be the Operator ID of an existing user in the system or a valid mobile phone number. If the operator ID is entered, operator information such as the name and phone number is populated as the customer data and the test message is sent to the operator's phone number.



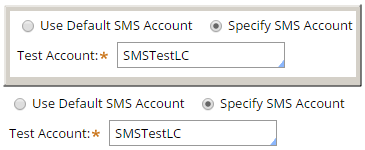
- **Seed List** - Select a predefined Seed List as the intended recipients of the test SMS message. The data for each seed is populated as the customer data and is visible (if referenced) in the message. In this case, the **Mobile Phone** column in the Seed List (.pyMobilePhone property) specifies the phone number to which the test SMS message is sent. Refer to the Seed Lists chapter for information on configuring a Seed List.



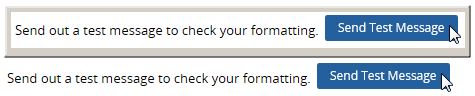
- **Test Account Settings** – Specify SMS Account settings for sending the test SMS message.
  - **Use Default SMS Account** - Select this option to use the Default outbound SMS Account.



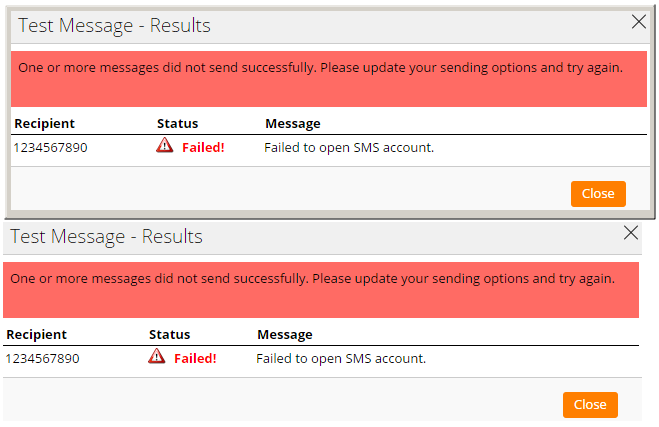
  - **Specify SMS Account** - Select this option to specify the outbound SMS Account to use for delivering the test message. Specify the account name in the **Test Account** field.



After the test message has been configured, it can be sent via the **Send Test Message** button.



Upon clicking this button, the SMS Treatment is saved and the test SMS message is sent to the specified recipients. The status of the test is displayed. Any errors in the test delivery are displayed in case of a failure. These can be corrected and the test message resubmitted for delivery.



Configuring push notification treatments

A push notification treatment represents the message which you want to communicate to your customer through a notification message in an app that you own. It is typically displayed on the lock screen of a mobile device where the app is installed. Opening a push notification brings the user to the app that sent the message.

Pega Customer Decision Hub

1. **Optional:** If the default list of fields available on the other treatment form does not contain a property which you require, edit the *PushTreatments* Decision Data rule used by the treatments.
   For more information, see Managing Decision Data rules in Pega Customer Decision Hub. Any new properties you add to the form are automatically available for all previously created treatments.
2. On the Treatments landing page, click **Create Push** .

3. Enter a unique name and a description for the treatment.
4. Specify when the treatment is active.
    - **Always**
    - **Never**
    - **Within a defined time period**
5. In the **Eligibility** section, select the eligibility condition that you want to apply to this treatment.
   For more information, see Eligibility conditions for treatments.
6. In the **Push Message** field, provide the text that should be displayed in the push notification.
   Tip: Push Notification shape in the action flow should be configured to reference .OfferData.PushMessage to display the Push Message.
7. Click **Create**.

Result: The treatment you created is now available for use in actions.

Configuring web treatments

A Web Treatment represents the visual message you want to communicate to your customer on the web channel. It typically consists of an image or HTML fragment and text which is rendered in a specific location on a web page. If a user clicks on this image or HTML, Pega tracks the click event and redirects the user to a specified web page. This redirection is specified through the click-through URL.

Pega Customer Decision Hub
Web Treatments are placed on web pages and contain a link which takes a Customer to a target page when clicked. To create a Web Treatment:

1. **Optional:** If the default list of fields available on the other treatment form does not contain a property which you require, edit the *WebTreatments* Decision Data rule used by the treatments.
   For more information, see Managing Decision Data rules in Pega Customer Decision Hub. Any new properties you add to the form are automatically available for all previously created treatments.
2. On the **Treatments** landing page, click **Create Create web** .
3. Enter a unique name and a description for the Treatment.
4. Specify when the Treatment is active.
   The following options are available:
    - **Always**
    - **Never**
    - **Within a defined time period**
5. In the **Eligibility** section, select the eligibility condition that you want to apply to this treatment.
   For more information, see Eligibility conditions for treatments.
6. In the **Content URL** field, provide a link to an image or HTML fragment to display for this Treatment. For example, this can be the URL of an image hosted in a content repository.
   After you populate the **Content URL** field, a preview of the content is displayed on the bottom of the page. For image content, the **Dimensions** field shows the image dimensions is pixels.
7. In the **Click-through URL** field, provide a link to a web page or a file that provides more details on the action for which you created the Treatment. Clicking this link will re-direct to the URL specified in this field.
8. In the **Format** field, specify whether the Treatment content is a plain image or an HTML fragment.
9. In the **Placement type** field, select the intended location and style of the treatment design. For example, the Treatment can appear as a large central banner, as a rotating strip of images, or as a footer.
   The following options are available:
    - **Carousel**
    - **Feature band**
    - **Footer bar**
    - **Hero**
    - **Page takeover**
    - **Tile**
10. In the **Language** field, select the language of the treatment.
    Specifying the treatment language allows Pega Customer Decision Hub to pick the right treatment based on the Customer's preferred language.
11. Click **Create**.

Result: The Treatment you created is now available for use in actions.**What to do next:** If you want to use adaptive analytics for Next-Best-Treatment selection, save the *Web_Click_Through_Rate* adaptive model into your *Artifacts* ruleset and add the relevant Customer property predictors. The predictors allow the adaptive model to learn and produce the desired results.

Configuring mobile treatments

A mobile treatment represents the visual message you want to communicate to your customer on the mobile channel. It typically consists of an image or HTML fragment and text which is rendered in a specific location on a web page when viewed on mobile devices. If a user clicks on this image or HTML, Pega Customer Decision Hub tracks the click event and redirects the user to a specified web page. This redirection is specified through the click-through URL.

Pega Customer Decision Hub

1. **Optional:** If the default list of fields available on the other treatment form does not contain a property which you require, edit the *MobileTreatments* Decision Data rule used by the treatments.
   For more information, see Managing Decision Data rules in Pega Customer Decision Hub. Any new properties you add to the form are automatically available for all previously created treatments.
2. On the Treatments landing page, click **Create Mobile** .
3. Enter a unique name and a description for the treatment.
4. Specify when the treatment is active.
    - **Always**
    - **Never**
    - **Within a defined time period**
5. In the **Eligibility** section, select the eligibility condition that you want to apply to this treatment.
   For more information, see Eligibility conditions for treatments.
6. In the **Content URL** field, provide a link to an image or HTML fragment to display for this treatment. For example, this can be the URL of an image hosted in a content repository. **Result:** After you populate the **Content URL** field, a preview of the content is displayed on the bottom of the page. For image content, the **Dimensions** field shows the image dimensions in pixels.
7. In the **Click-through URL** field, provide a link to a web page or a file that provides more details on the action for which you created the treatment. Clicking this link will re-direct to the URL specified in this field.
8. In the **Format** field, specify whether the treatment content is a plain image or an HTML fragment.
9. In the **Placement type** field, select the intended location and style of the treatment design. For example, the treatment can appear as a large central banner, as a rotating strip of images, or as a footer.
    - **Carousel**
    - **Feature band**
    - **Footer bar**
    - **Hero**
    - **Page takeover**
    - **Tile**
10. In the **Language** field, specify the language of the treatment.
11. Click **Create**.

Result: The Treatment you created is now available for use in actions.**What to do next:** If you want to use adaptive analytics for Next-Best-Treatment selection, save the *Web_Click_Through_Rate* adaptive model into your *Artifacts* ruleset and add the relevant Customer property predictors. The predictors allow the adaptive model to learn and produce the desired results.

Configuring agent-assisted treatments

An agent-assisted treatment represents the message which you want a call center agent or retail assistant to communicate to your customer during an interaction such as a call center call or an in-person visit to a brick and mortar retail store.

Pega Customer Decision Hub

1. **Optional:** If the default list of fields available on the other treatment form does not contain a property which you require, edit the *AssistTreatments* Decision Data rule used by the treatments.
   For more information, see Managing Decision Data rules in Pega Customer Decision Hub. Any new properties you add to the form are automatically available for all previously created treatments.
2. On the Treatments landing page, click **Create Agent assisted** .
3. Enter a unique name and a description for the treatment.
4. Specify when the treatment is active.
   - **Always**
   - **Never**
   - **Within a defined time period**
5. In the **Eligibility** section, select the eligibility condition that you want to apply to this treatment.
   For more information, see Eligibility conditions for treatments.
6. **Optional:** In the **Relevant description** field, enter a short description.
   If you leave the field blank, the **Relevant description** field will be populated with the description of the action with which you associate the treatment.
7. In the **Content URL** field, provide a link to an image or HTML fragment to display for this treatment. For example, this can be the URL of an image hosted in a content repository. **Result:** After you populate the **Content URL** field, a preview of the content is displayed on the bottom of the page. For image content, the **Dimensions** field shows the image dimensions in pixels.
8. In the **Format** field, specify whether the treatment content is a plain image or an HTML fragment.
9. **Optional:** In the **Benefits** field, enter a description of the benefits associated with the action.
   If you leave the field blank, the **Benefits** field will be populated with the benefits specified for the action with which you associate the treatment.
10. In the **Eligibility description** field, enter a description of the cicumstances under which the action is relevant for a customer.
    If you leave the field blank, the **Eligibility description** field will be populated with the eligibility description specified for the action with which you associate the treatment.
11. In the **Treatment type** field, specify the agent-assisted channel to which the treatment applies.
12. In the **Language** field, specify the language of the treatment.
13. Click **Create**.

Result: The Treatment you created is now available for use in actions.

Configuring other channel treatments

You can create treatments for channels other than the ones which Pega Customer Decision Hub provides out of the box. By default, the Other Treatment rule form includes the same fields as the Web and Mobile forms, with the addition of a **Treatment type** field for easier classification.

Pega Customer Decision Hub

1. **Optional:** If the default list of fields available on the other treatment form does not contain a property which you require, edit the Decision Data rule form used by the treatments.
   a. To create a new property, in the Pega Customer Decision Hub portal, click **Next-Best-Action Designer Taxonomy Properties** .
   b. Click **Add Add property** , and then configure the required property.
      For more information about configuring properties, see Properties.
   c. To add the new property to the treatment form, in the Pega Customer Decision Hub portal, click **Intelligence Decision Data** .
   d. Find and open the *OtherTreatments* Decision Data rule form, and then click the **Form** tab.
   e. In the Form fields section, review the list of properties which are available as fields on the treatment form.
      - To add a field, click the **Add field** button.
      - To remove a field, click the **Remove property** icon.
   f. Open the section *CustomOtherTreatmentElement* and save it into your implementation layer.
   g. Add the same fields that you added on the **Form** tab to the new section.
   h. Open the section *CustomOtherTreatmentElement* and save it into your implementation layer.
   i. Add the same fields that you added on the **Form** tab to the new section.

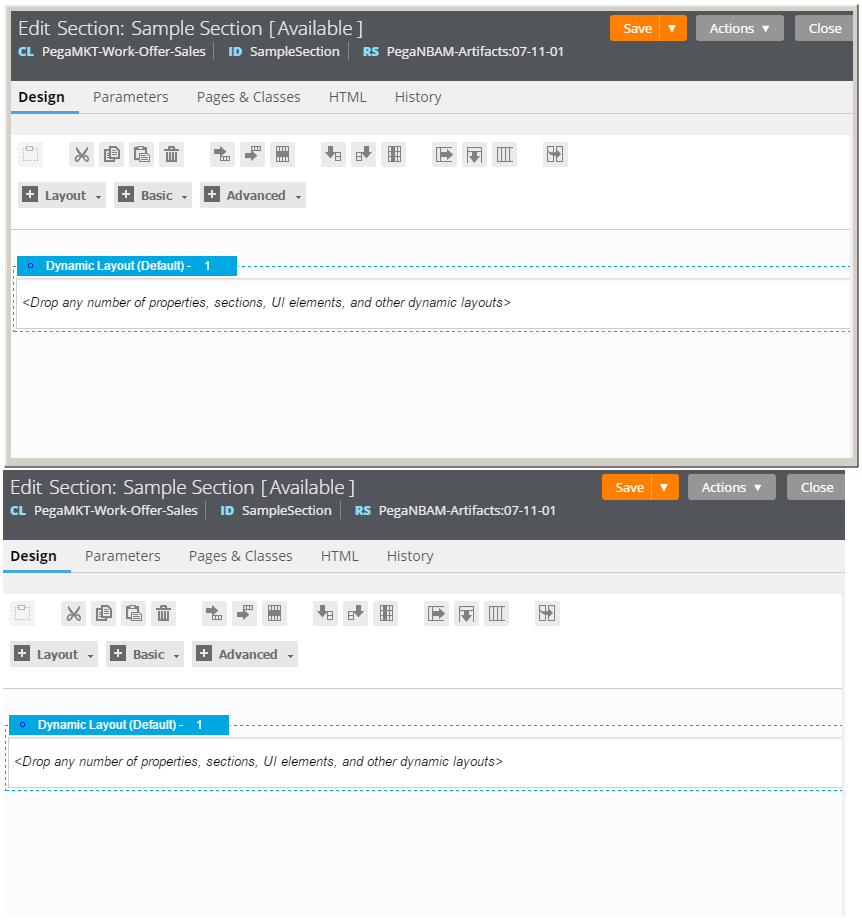
   Any new properties you add to the form are automatically available for all previously created treatments.
2. On the Treatments landing page, click **Create Other** .
3. Enter a unique name and a description for the treatment.
4. Specify when the treatment is active.
   - **Always**
   - **Never**
   - **Within a defined time period**
5. In the **Content URL** field, provide a link to an image or HTML fragment to display for this treatment. For example, this can be the URL of an image hosted in a content repository. **Result:** After you populate the **Content URL** field, a preview of the content is displayed on the bottom of the page. For image content, the **Dimensions** field shows the image dimensions in pixels.
6. In the **Click-through URL** field, provide a link to a web page or a file that provides more details on the action for which you created the treatment. Clicking this link will re-direct to the URL specified in this field.
7. In the **Treatment type** field, type in or select the type of treatment, for example, IVR.
   The **Treatment type** property corresponds to the channel for which the treatment is intended.
8. In the **Format** field, specify whether the treatment content is a plain image or an HTML fragment.
9. In the **Language** field, specify the language of the treatment.
10. Click **Create**.

Result: The Treatment you created is now available for use in actions.

Sections

A newly created Section starts out as shown below.

Pega Customer Decision Hub



For more information about creating Sections, refer to the Pega Platform documentation.

Note: Unlike other artifacts, a Section rule does not show Issue and Group drop-downs in the Create form. However, these are still correctly set using the values from the action rule. The **Apply To** setting of the Section rule is automatically set to correspond to the issues and group class under PegaMKT-Work-Offer.

Adding custom attributes to treatment form

You can add custom attributes to treatment form to describe additional characteristics of the treatment.

Note: Custom attributes cannot be added to Internal Email and Internal SMS treatments.
Pega Customer Decision Hub

1. Log in to Pega Customer Decision Hub portal.
2. Add new custom property to the Next-Best-Action Designer Taxonomy.
3. Update the Treatment decision data rule (DDR):
   a. Click **Content Treatments** .
   b. Click **Manage treatments**.
   c. On the treatment rule form, click the **Form** tab.
   d. In the **Form fields** section, review the list of properties which are available as fields on the treatment form.
      - To add a field, click the **Add field** button.
      - To remove a field, click the **Remove property** icon.**Tip:** Any new properties that you add to the form are automatically available for all previously created treatments.
4. Switch to Dev Studio.
5. Add the property to the Treatment section:
   a. Search and open the custom extension section corresponding to the channel of the treatment to which you want to add the custom properties to. The name of custom extension section is of the format *Custom<Channel>TreatmentElement*, for example *CustomWebTreatmentElement*.
   b. Save the section into your implementation ruleset by clicking **Save as**.
   c. On the **Design** tab of the new section, add the property that you created.
6. Update the *<Web/Mobile/Assisted/Other>TreatmentStrategy* or *<Email/SMS/Push>TreatmentStrategy*:
   a. Search for and open the *<Web/Mobile/Assisted/Other>TreatmentStrategy*, *<Email/SMS/Push>TreatmentStrategy*.
   b. Save the strategy into your implementation ruleset.
   c. In the **Top Level SR class** field, select the strategy result.
   d. On the **Strategy** tab, open the **<Channel> Treatment Library** component.
   e. In the **Decision data > Import** section, clear the **Decision Data** field, and then re-select the <Channel>Treatments decision data rule.

      This updates the value of the **Results in class** field to your top-level SR class, for example, *<Application>-SR class*.

   f. On the **Mapping** tab of the component, click **Remap properties**, and then click **Submit**.
   g. Open the **Join Treatment Library** component.
   h. On the **Properties mapping** tab of the component, click **Add item** to add the property that you created, and then click **Submit**
   Tip: If the property is not visible, switch to another component, and then switch back to the **Join Treatment Library** component.
7. Configure the custom property in the real-time container response. Follow step 2 onwards as described in Setting the custom property in a real-time container response.

Defining customer audiences with customer segments

Use customer segments to identify a group of customer entities to use as audience in always-on outbound campaigns.

Pega Customer Decision Hub

The following types of segments are supported:

- Creating List Segments
- Creating a Criteria Segment
- Managing Segments

- Defining an exclusion list of customers who opted out of receiving notifications
- Managing prospects

Creating List Segments

You can use a List Segment to quickly convert a list of customers into a Segment that Pega Customer Decision Hub can use as the audience in Campaign and scheduled Next-Best-Action execution.

Pega Customer Decision Hub
Advanced users can use the customer list to update existing customer data. You can also manually edit the list of customers included in the Segment.

1. Click **Audiences Segments Create** .
2. Enter a **Description** for the new Segment.
3. From the **Type** list, select **List**.
4. Click **Create and Open**.
5. Add customers to your Segment. There are two ways of adding customers to List Segments:
   - Importing Customers
   - Manually Adding Customers to a List Segment
6. Populate the segment.
   - If you have **Check in / out** enabled, click **Check in**.
   - Otherwise, click **Save**.

   Pega Customer Decision Hub automatically starts to populate the Segment with the current customer list. The total number of customers in this Segment is displayed in the **Population count** field.
7. After you create a Segment, you can view it in the Segments landing page. For more information, see Managing Segments.

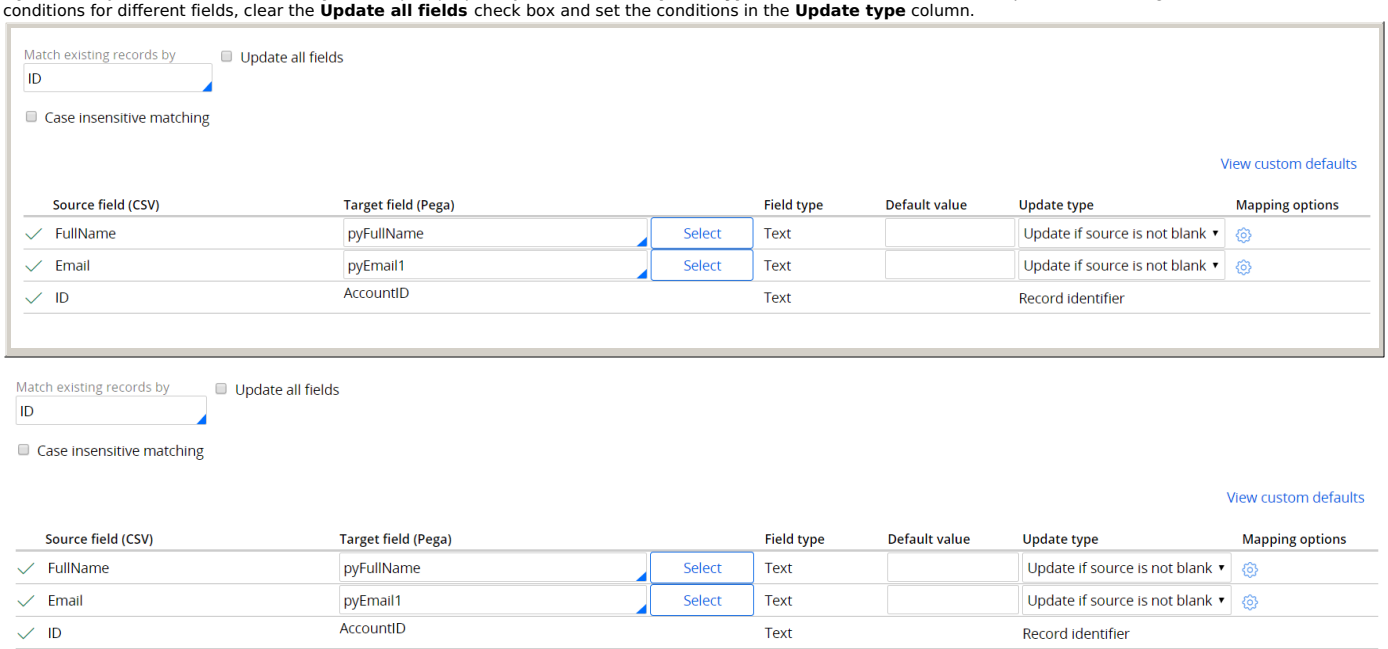
- Importing Customers
- Manually Adding Customers to a List Segment

Importing Customers

You can use a comma-separate values (CSV) file to import customer data into your application. By adding or updating customer records in bulk, you can quickly update the Segment. You can import customer data at any time.

Pega Customer Decision Hub

1. On the **Customers list** tab of the Segment, click **Import**.
   Note: The import process uses the Data Management functionality of Pega Platform. For details on this process, click the help icon in the wizard to launch the Pega Platform help topic.
2. Click **Choose file** , and then upload the CSV file with your customer data.
3. In the **Purpose** list, select one of the following options:
   - **Add or update** - Creates customer records for entries in the import file that do not exist in the customer table, updates existing customer records, and adds all valid customers to the List Segment.
   - **Add only** - Creates customer records for entries in the import file that do not exist in the customer table, ignores existing customers, and adds all valid customers to the List Segment.
   - **Populate Segment Only** - Adds existing customers from the import file to the List Segment, but does not modify the customer table.
4. Click **Next**.
5. In the **Match existing records by** field, select the data field from your import file which will be used to match existing customer records. For example, select CustomerID. The field used to match records must be unique.
6. For each **Source field** in the import file, specify the **Target field** mapping. If you previously saved a field mapping template, you can select it from the **Template type** list.
7. **Optional:** If you selected the **Add or update** import purpose, you can use the **Update type** list to control which fields are updated for existing customers. To set different conditions for different fields, clear the **Update all fields** check box and set the conditions in the **Update type** column.



8. Click **Next**.
9. **Optional:** To save a field mapping template, select the **Save import settings as a template** check box.
10. Click **Start validation**.
    Tip: If the validation reports any invalid rows, click the **See more** link to download a CSV file containing the records which failed to import. The last column in this file contains a description of the error that occurred. Resolve the errors before you proceed.
11. Click **Continue import**.
    When importing large files, the import progress can take some time. You can close the import window and continue with other activities. The system displays the status of the import at the top of the customer grid. Once the import completes, the system displays the number of instances that were added or updated by the import. If there were errors during the import, the number of erroneous records is also shown.
12. Click **Finish**.

Manually Adding Customers to a List Segment

You can manually add existing customer records to your Segment.

Pega Customer Decision Hub

1. On the **Customers list** tab of the List Segment, click **Add**.
2. Enter a part of the customer name, email, or ID into search box, and then click the search icon.
3. Select one or more customers to add to the List Segment, and then click **Add**.
4. **Optional:** Repeat the above steps to add more customer records to the Segment.
5. Click **Done**.

Creating a Criteria Segment

With Criteria Segments, you can define one or more conditions that must be true before a customer can be part of audience.

Pega Customer Decision Hub

Historically, the assembly of criteria to describe a Segment has been an area in which the marketer had to rely on intuition, prior complex analysis, or guess work. Pega Customer Decision Hub solves this issue through the use of intelligent segmentation. Intelligent segmentation uses statistical analysis to discover predictors within the customer data that contribute positively towards a targeted outcome, for example, the purchase of a product. For more information about intelligent segmentation, see Intelligent Segmentation (deprecated).

1. Click **Audiences Segments** .
2. Click **Create**.
3. Enter a **Description** for the new Segment.
4. In the **Context** section, select a customer context.

   By default, the primary context, or customer entity, is Customer, mapped to the PegaMKT-Data-Customer class. You can use the Context Dictionary to review or update the default context, configure a different primary context, or add an additional customer context. For example, you can configure Account or Subscriber as additional customer entities. For more information about configuring additional customer entities, see "Setting the context for your customer class" in the *Pega Customer Decision Hub Implementation Guide* on the Pega Customer Decision Hub product page.

5. In the **Type** section, select **Criteria**.
6. Optional: If you have an Analysis Project which you want to use in this Segment, select it from the **Analysis Project** drop-down list. For more information, see Configuring Analysis Projects (deprecated).
7. Click **Create and Open**.
8. Add criteria groups to your Segment. For more information, see Designing Criteria Segments.
9. Click **Save**. The Criteria Segment is saved and checked in.
10. Click **Run** to populate the Segment with the current set of customers.
    You can click **Stop** to stop the Segment execution, or continue with other work while the Segment populates. After Pega Customer Decision Hub finishes populating the Segment, **Population count** field displays the number of customers in this Segment out of the total number of customers in your application.**Note:** If your Pega Customer Decision Hub application is installed on a PostgreSQL database, the total number of customers is shown as an approximation.
11. **Optional:** To automatically refresh your Segment, configure a schedule. For more information, see Optional: Configuring Criteria Segment Runs.

What to do next: Once a Segment is created, you can view it in the Segments landing page. For more information, see Managing Segments.

- Designing Criteria Segments
- Optional: Configuring Criteria Segment Runs
- Intelligent Segmentation (deprecated)

Designing Criteria Segments

This section describes the default method of designing Criteria Segments. If required, you can also build a Criteria Segment by using a legacy non-visual rule form.

For more information, see Designing Criteria Segments by using a rule form (deprecated).

Pega Customer Decision Hub
To design a Criteria Segment, add one or more criteria groups. A group is a container for multiple criteria that can be combined together to represent a desired logical grouping of the population. These groups are evaluated from top to bottom and can be organized into sets.

1. In the **Design**tab of a Criteria Segment, click **Add group**.
2. Click **Add criteria**.
3. Provide values in the fields on the following tabs:
   - **Analysis project** - Lists all the predictors available in the Analysis Project. This is only available when you associate an Analysis Project with the Segment and run the Analysis Project at least once.
     - **Recommendations** - Lists the best performing predictors from the modeled analysis for the selected Analysis Project. That is, it shows the predictors which are the most effective at predicting the required outcome.
     - **Modeled fields** - Lists all the predictors available in the Analysis Project that have been assembled during the execution of the modeled analysis. By default, the predictors are sorted by the **Performance** column. The predictors which are the most effective at predicting the required outcome are displayed on top of the list.
     - **Static fields** - Lists the set of static attributes from the Analysis Project which have been processed as part of static analysis execution. The **Group** column displays the number of groupings for the predictor. The **Count** column displays the number of items across all groupings for the predictor.

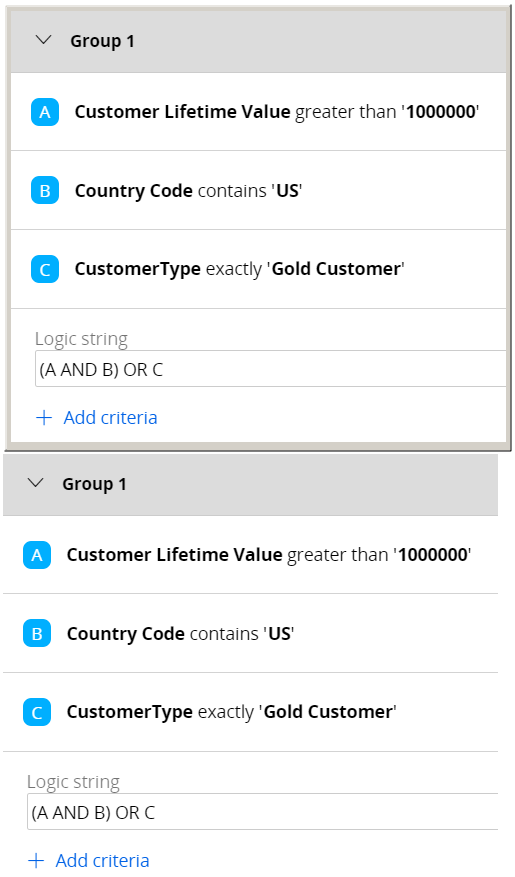
     For more information about Analysis Projects, see Configuring Analysis Projects (deprecated).

   - **Data fields** - This tab allows you to select the customer data fields which you want to use as criteria in your Segment. Use the **Search** box to look for specific data fields, or scroll through the list of available fields. The following types of data fields are available in this tab:
     - **Customer** - data fields for customer properties, for example **CustomerType** or **Customer Lifetime Value**.
     - **Customer History** - data fields for customer history, for example the **Outcome** of an interaction with the customer.
     - You can also configure other associated data entities which should be available in this tab, for example **Customer Address** . To configure additional data entities, click **Settings** in the App Studio navigation bar, then click **Context Dictionary**.
   - **History fields** - This tab allows you to select the criteria related to treatment and action responses. For example, you can use the **Received action via channel during a time period** to look for customers who were sent an email with an action within the last two weeks.
   - **Segments** - This tab allows you to select other Segments as criteria within this Segment. For example, you can select which existing customer Segment should be excluded from this Segment by selecting the existing Segment as a criterion and then clicking **Exclude**. Selecting a Segment criterion automatically includes or excludes all customers in that Segment.
4. Click any criteria to add them to the criteria group.

   For example, if you want the Segment to include only customers above a specific Customer Lifetime Value, add the Customer Lifetime Value criterion and then enter a **Greater than** value. Depending on the criteria type, you may need to specify additional settings. For more information, see Criteria Types.

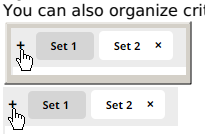
   You can add up to 26 criteria to a criteria group. By default, the criteria are applied with an AND logic. For example, you can look for customers with Customer Lifetime Value greater than 100 000 AND located in the United States.

5. If you want to change the relationship between some criteria to OR, edit the value in the **Logic string** field. For example, you can configure the criteria group to look for customers who match both criterion A AND criterion B, OR criterion C.



6. **Optional:** To remove a criterion from the logic string and the Segment, click the **Delete** icon .
7. Click **Add group** to add another criteria group, if needed. You can add multiple criteria groups and combine them by using the following actions:
   - **Start with** – Selects customers in the top group to be combined with the actions of the other criteria groups.
   - **Intersect** – Includes customers that meet the group's criteria in the Segment population. This action is available for groups other than the first.
   - **Exclude** – Removes customers that meet the group's criteria. This action is available for all groups.
   - **Merge** – Adds customers matching the criteria to the Segment. These customers do not have to meet the criteria that is specified in the preceding groups. The Merge action removes the duplicate customers that would be added to the Segment. The Merge action is available for groups other than the top group.

   The WATERFALL COUNT field reflects the flow of customers through Segment criteria groups.
8. **Optional:** To rename a criteria group, double-click on the group name.
9. **Optional:** To remove a criteria group, click **Actions Remove** .
10. You can also organize criteria groups into separate sets for easier viewing. Click the **+** icon to add a new criteria set.



11. **Optional:** To rename a set, double-click on the set name.
12. **Optional:** To remove a set, click **X** by the set name.

- Criteria Types
- Designing Criteria Segments by using a rule form (deprecated)

Criteria Types

The configuration options for a criterion depend on the criteria type. The following criteria types are supported:

Pega Customer Decision Hub

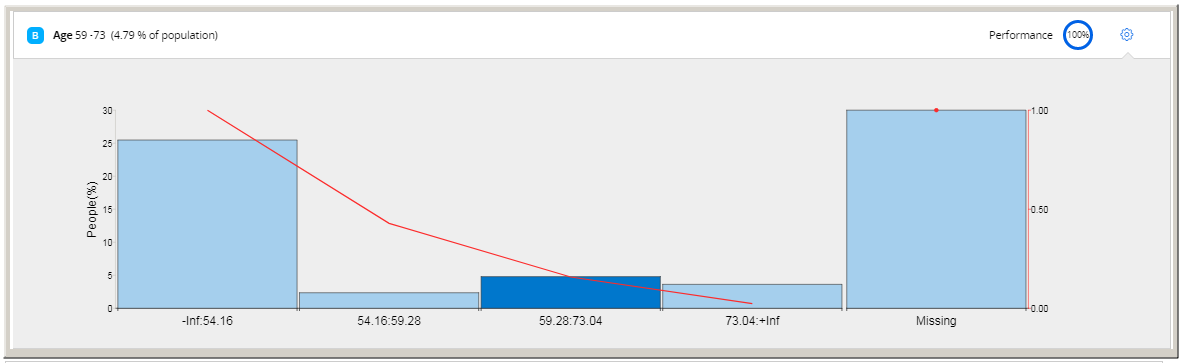
- **Analysis project Modeled fields**
  - Histograms
  - Categoricals
  - Long Categoricals
- **Analysis project Static fields**
  - Categoricals
  - Long Categoricals
- **Data fields**
  - Strings
  - Numbers
  - Dates and Times

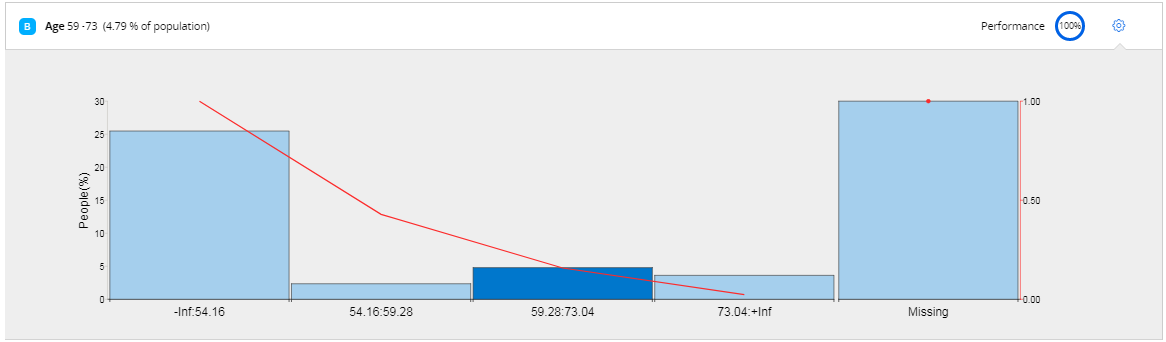
- Histograms
- Categoricals
- Long Categoricals
- Strings
- Numbers
- Dates and Times
- Aggregating Data
- Selecting Criteria Values from a List

Histograms

A histogram displays a numerical property which produced meaningful analysis information during modeled analysis execution.

Pega Customer Decision Hub





The following information is available in a histogram:

- **Performance** - The predictive power of this criterion.
- The grouping of the analyzed data, displayed as individual columns of the histogram. Ranges for the groupings are displayed as labels on the horizontal axis.
- The percentage of the overall population within the group, represented by the height of each column. Percentage values are displayed as labels on the left vertical axis.
- The propensity of individual groups, represented by the red performance line. Propensity is the ratio of the number of positive responses to the total number of responses. The propensity values are displayed as labels on the right vertical axis.

Click on a column to select or deselect groupings. You can select multiple groupings for the criterion. The system selects the column with the highest propensity by default.

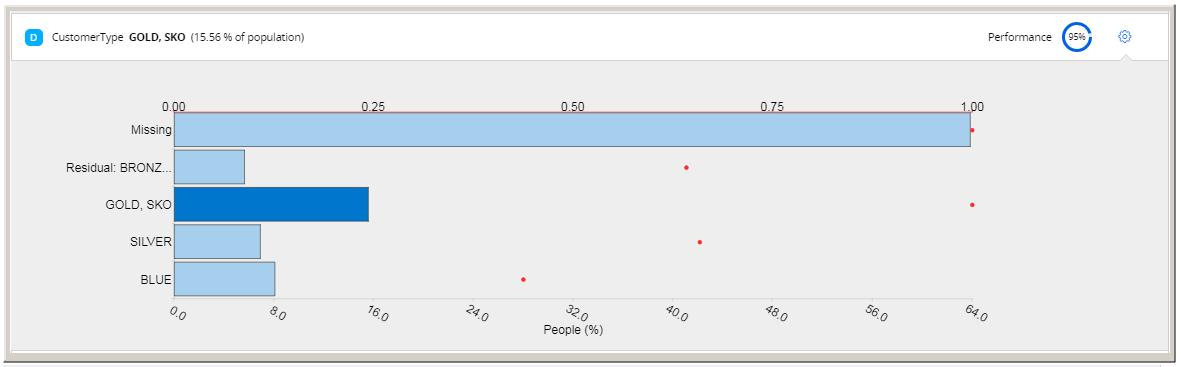
Pega Customer Decision Hub uses the range values associated with each group to compose queries against the customer source. It treats the left edge of a range as **greater than or equal to** (>=) and the right edge as **less than** (<) in its queries.

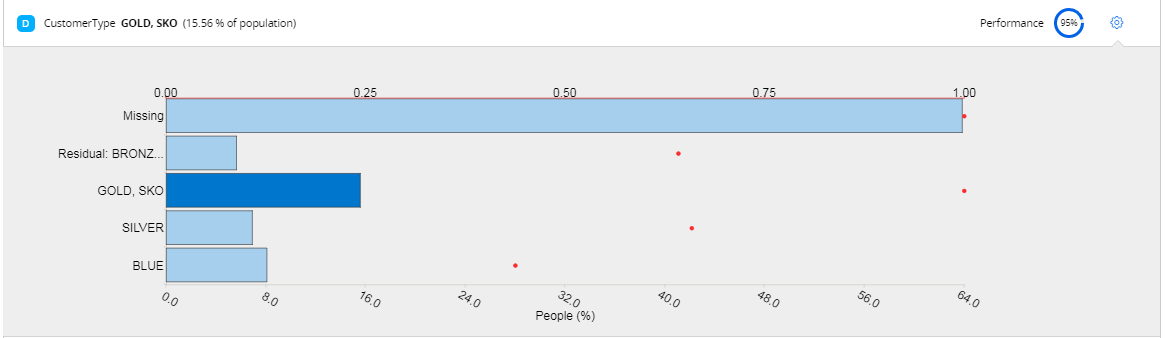
Categoricals

The categorical criteria editor is used for the following criteria types when they have no more than eight groupings:

Pega Customer Decision Hub

- Symbolic modeled analysis criteria
- Numeric and symbolic static analysis criteria





The following information is available:

- **Performance** - The predictive power of this criterion.
- The percentage of the population within the group, represented by the length of each bar. Percentage values are displayed as labels on the bottom horizontal axis.
- The propensity of individual groups, represented by the red dots. Propensity is the ratio of the number of positive responses to the total number of responses. The propensity values are mapped to the top horizontal axis. This information is only available for modeled analysis criteria.
- **Range** - Symbols in this group. To see this information, move your mouse over a bar or a propensity dot. The information is displayed as a tool tip.
- **People (%)** - Percentage of population (based on Sample data) in this group. To see this information, move your mouse over a column. The information is displayed as a tool tip.
- The predictive power of this group. To see this information, move your mouse over a propensity dot.

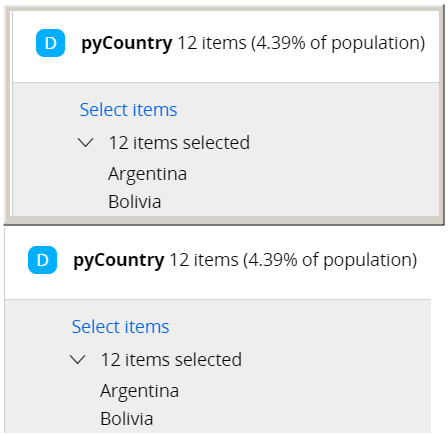
Click on a bar to select or deselect groupings. You can select multiple groupings, including **Missing**. For modeled analysis criteria, the system selects the bar with the highest propensity by default.

Pega Customer Decision Hub uses the symbols associated with each group to compose suitable queries for this criterion. When multiple symbols exist in a group, the system will query for all present symbols. For the **Missing** group, the system translates the query to search for where the criterion value is null.

Long Categoricals

If the number of elements for a categorical criterion is greater than eight, it becomes impractical to visualize this criterion as a bar chart. Instead, the criterion is displayed as a long categorical.

Pega Customer Decision Hub



Click **Select items** to open a pop-up window with a list of groupings for the criterion. You can select one or more groupings.

The following information is available:

- **Name** - Name of the grouping
- **Count** - Number of items in the grouping
- **Percent** - Percentage of items (based on Sample data) in the grouping.

Strings

The string criteria editor is used for text criteria available on the **Data fields** tab.

Pega Customer Decision Hub

The following operators are available for string criteria:

- **Contains**
- **Starts with**
- **Ends with**
- **ExactlyNote:** You can specify more than one value for the **Equal to** operator. Use a comma to separate the values, for example, `Ready,Waiting`. If you specify multiple values for the operator, customers who match at least one of the values are added to the Segment.
- **Is null**
- **Count** - For more information, see Aggregating Data
- **In list** - For more information, see Selecting Criteria Values from a List

The following additional options are available:

- **Exclude** – Negate the current condition
- **Include missing values** – Include results that do not have a value specified for the selected criterion
- **Ignore case** – Perform a case-insensitive search

Numbers

The number criteria editor is used for numerical criteria available on the **Data fields** tab.

Pega Customer Decision Hub

The following operators are available for number criteria:

- **Between**
- **Greater than**
- **Greater than or equal to**
- **Less than**
- **Less than or equal to**
- **Equal toNote:** You can specify more than one value for the **Equal to** operator. Use a comma to separate the values, for example, `2000, 100`. If you specify multiple values for the operator, customers who match at least one of the values are added to the Segment.
- **Missing value**
- **Count** - For more information, see Aggregating Data
- **In list** - For more information, see Selecting Criteria Values from a List

The following additional options are available:

- **Exclude** – Negate the current condition

- **Include missing values** – Include results that do not have a value specified for the selected criterion

Dates and Times

The date and time criteria editor is used for date and time criteria available on the **Data fields** tab. Use it to configure criteria for a range of dates and times or perform standard comparison tests against date and time values.

Pega Customer Decision Hub

The following options are available for date and time criteria:

- **Date** - Configure only dates for this criterion
- **Date & Time** - Configure both dates and times for this criterion.
- **Time** - Configure only times for this criterion.
- **Missing value** - Search for instances where this criterion value is null.
- **Relative date** - Search for instances of this criterion value using date operators relative to the current date. The following operators are available for relative date comparison:
  - **More than** - Use this option to search for instances where this criterion is strictly more than X days before today (i.e. X + 1 days before or earlier).
  - **At least** - Use this option to search for instances where this criterion is at least X days before today.
  - **In exactly** - Use this option to search for instances where this criterion is exactly X days after today. To search for days before today, specify a negative value for **Days**. For example, use -X days with this option to search for instances where this criterion is exactly X days before today.
  - **In more than** - Use this option to search for instances where this criterion is at least X days after today.
  - **In at least** - Use this option to search for instances where this criterion is strictly more than X days after today (i.e. X + 1 days after and later).
  - **In last** - Use this option to search for instances where this criterion is within the last X days.
  - **In next** - Use this option to search for instances where this criterion is within the next X days.
- **Occurrences** - For more information, see Aggregating Data.

The following additional options are available:

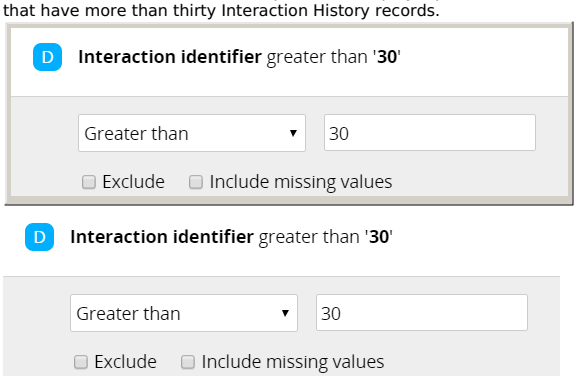
- **Exclude** – Negate the current condition
- **Include missing values** – Include results that do not have a value specified for the selected criterion

Aggregating Data

For string, number, and date and time criteria fields that belong to a repeating structure (such as a list of purchases), you can select the **Occurrences** search operator to aggregate the criteria values.

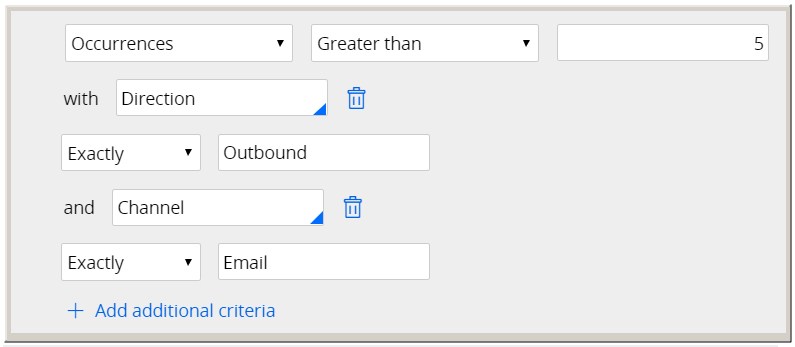
Pega Customer Decision Hub

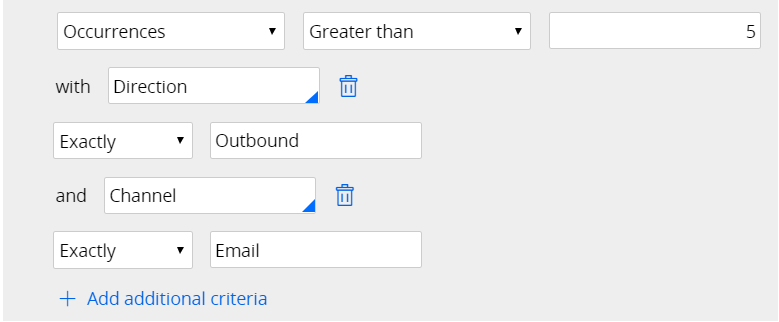
1. Select the **Occurrences** search operator to display input fields for the comparison operator and the comparison value. The following example criteria targets all customers that have more than thirty Interaction History records.



2. Click **Add additional criteria** to specify additional criteria for the aggregation. Each additional criterion must be on the same parent entity as the original criteria being aggregated.
3. After selecting the additional criteria, fields relevant to the criteria type are displayed. Use these fields to configure the additional criteria.

   The additional criteria are combined using AND logic. The following example targets all customers that have had more than five outbound email communications.





Note: Aggregation using the Occurrences operator is only available for first-level entities, that is, those entities that are directly associated with the customer class and not nested under other associated entities.

Selecting Criteria Values from a List

For string and number criteria available on the **Data fields** tab, you can select the **In list** search operator to view, search for, and select filter values for the criterion from a list of possible values.

Pega Customer Decision Hub

The items grid in the criteria editor lists the values that are currently selected for the criterion.

Note: For best performance, optimize the properties which you want to use as Segment criteria. For more information, see the *Pega Customer Decision Hub Implementation Guide* on the Pega Customer Decision Hub product page.

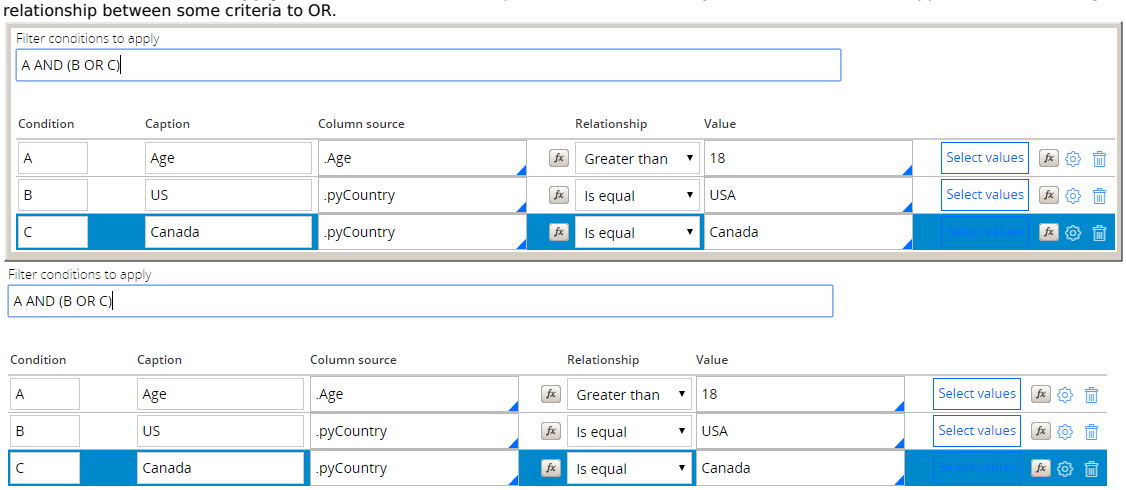
1. Select the **In list** search operator, and then click **Select items** to open the **Select items** dialog box.
2. Pick one or more values from a list of existing values for the criterion.
3. To find specific values, enter the search text in the **Search** field, and then click the search icon.
4. To add or remove individual items, select or clear the check box next to an item. You can select or clear multiple items.
5. To add or remove all items, click **Select all** or **Deselect all**.
6. To exclude customers from the results who have any of the selected values as their value for the criterion, select **Exclude**.
7. To include customers that do not have a value specified for the selected criterion, select **Include missing values**.
8. Click **Apply**.

Designing Criteria Segments by using a rule form (deprecated)

In this approach, you provide a series of criteria by identifying the appropriate column, the relationship type, and the value for which to search. These criteria are then combined in the **Filter conditions** field to allow the user to describe the logic for applying the various criteria.

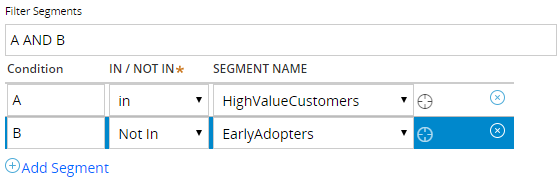
Pega Customer Decision Hub
Note: This approach to designing Criteria Segments is deprecated. If possible, design all Segments using the standard approach. For more information, see Designing Criteria Segments.

1. On the **Design** tab of a Criteria Segment, click the rule form icon .
2. In the rule form, click **Add filter**.
3. In the **Column source** field, enter or select the property which you want to use as a criterion.
4. In the **Relationship** list, select the operation that evaluates the property and value.
5. In the **Value** field, enter the required value of the property.
6. Use the gear icon to add filter options, such as **Ignore case** and **Use null if empty**.
7. In the **Filter conditions to apply** field, define the relationship between the criteria. By default, the criteria are applied with an AND logic. If needed, you can change the relationship between some criteria to OR.



8. **Optional:** Select other Segments as criteria within this Segment. Selecting a Segment criterion automatically includes or excludes all customers in that Segment.
   a. In the **Segments** section, click **Add Segment**.
   b. Select the Segment from the **SEGMENT NAME** list.
   c. In the **IN / NOT IN** list, select whether the new Segment evaluates customers who are in or not in the existing Segment.
   d. In the **Filter Segments** field, define the relationship between the Segments.

   Segments



9. Click **Save**.

Optional: Configuring Criteria Segment Runs

In the **Options and Schedule** tab of the Segment, you can configure various options for the Segment and schedule its execution. All settings in this section are optional and can be configured according to your needs.

Pega Customer Decision Hub

1. In the **Data Options** section, configure how data in the Segment is refreshed at run time. The following options are available:
   - **Enable Transactional Refresh** - Select this option for Segments that are used for exclusion lists or transactional notifications to customers. Transactional refresh is more resource-heavy than the default mode, but it ensures correct Segment population in the case of concurrent refreshes.
   - **Refreshable Segment** - Segments which refer to this Segment as part of their criteria can automatically refresh this Segment when they are run.
   - **Refresh Child Segments** - The Segment can automatically refresh data in its child segments, so long as they are marked as refreshable.

   If the refreshable option is not enabled, then the Segment's contents will remain untouched and used as is in the criteria evaluation of the parent, and any child Segment beneath the non-refreshable Segments are also not refreshed.

   Note: Top-level Segments in a dependency chain are always refreshed. Normally, Segments lower down in a dependency chain should also be marked as refreshable, as this ensures that any parent Segment is set up in the most accurate way. However, as a best practice, disable the refreshable option for all Segments that are too costly (in

terms of time and processing resources) to refresh every time a parent Segment is refreshed.
2. In the **Sampled Segments Options** section, select a sample of criteria results. For more information, see Control Groups.
3. In the **Control Group Options** section, designate Segment results as a control group. For more information, see Control Groups.
4. In the **Scheduling Options** section, select the **Enable Schedule** check box to control when the segment runs.
   Caution: You must check in the Segment. Segment scheduling or the removal of any prior schedule only takes place after the Segment is checked in.
5. In the **Delivery Options** panel, configure the Pega Customer Decision Hub to send email notifications for scheduled runs and refreshes. You can customize the email by using the provided fields. The email recipient must be an operator in the system.

- Control Groups

Control Groups

In an experiment, a Control Group is a group, separated from the rest of the experiment, which is isolated from the variable being tested. In customer engagement, Control Groups can be used to assess the performance of customer engagement initiatives.

Pega Customer Decision Hub

Pega Customer Decision Hub provides support for defining and populating Control Groups. In addition, the Strategy rule also provides support for assessing whether a customer is in a Control Group. This allows the user to treat Control Group members differently. For example, Control Groups members might not receive a particular action.

Pega Customer Decision Hub uses Interaction History (IH) to maintain Control Group memberships. This allows Control Groups to be used in both batch and real-time strategies.

Intelligent Segmentation (deprecated)

Intelligent segmentation assembles recommended criteria from statistical analysis of various input sources and ranks them according to how well they do at predicting certain outcomes. This helps the marketer to identify groups of customers, form Segments, and use them without the need for guess work.

Pega Customer Decision Hub
Note: This functionality is considered deprecated and may be removed in future versions of the product.

Intelligent segmentation is supported by Analysis Projects, which create a set of predictors based on samples of the main customer data. Intelligent segmentation also uses sampling strategies to optimize the processing involved in discovering these predictors.

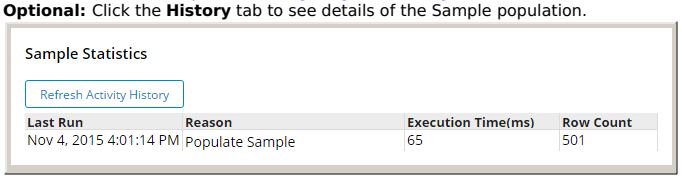
- Configuring a Sample (deprecated)
- Configuring Analysis Projects (deprecated)

Configuring a Sample (deprecated)

Use a Sample to define a sample of the overall customer population. An Analysis Project takes a Sample and performs a range of statistical analysis activities to determine which data fields have predictive power or influence when measured against a goal, for example the purchase of a product or service.

Pega Customer Decision Hub
Note: Starting in Pega Customer Decision Hub version 8.5, this functionality is considered deprecated and may be removed in future versions of the product.

1. Click **Campaign Segmentation Support** .
   Note: If you do not see the **Campaign** menu in the Pega Customer Decision Hub, enable the Marketing Campaign Edition.
2. In the **Sample populations** tab, click **Create**.
3. Enter the Sample name.
4. **Optional:** Enter the Issue and Group to classify the Sample.
5. Click **Create and open**.
6. On the **Design** tab, in the **About This Sample Population** section, configure the following information according to your requirements:
   a. **Sample Source** - Describes the location of the sample data.
   b. **Anticipated Sample Size** - Describes the required size in terms of an absolute number (**Fixed**) or a percentage (for example, 10%).
   c. **Sample Strategy** - Describes various sampling strategies. For **Ascending** and **Descending**, use the **Ordered On** field to identify the field in the data set on which to order.
7. In the **Sample Fields** section, clear any check boxes corresponding to fields which are not useful in analysis. As a general rule, fields that are system-related (for example partition key) or unique for individual customers (for example their customer id or their full name) typically have no value from an analysis perspective.
8. Click **Save**.
9. Run the Sample.
   - To run the Sample manually:
     1. Click **Run**. The **Run Sample** pop-up window is launched.
     2. Click **Confirm** to populate the Sample. The **Sample size** field is updated to reflect the number of rows in the Sample.
     3. Click **Preview** to preview the contents of the Sample and its associated customer data.
   - To run the Sample automatically, schedule the sample execution in the **Schedule** tab. Sample scheduling works in the same way as Segment scheduling. For more information, see Optional: Configuring Criteria Segment Runs.
10. **Optional:** Click the **History** tab to see details of the Sample population.



Sample Statistics
Refresh Activity History
Last Run
Reason
Execution Time(ms)
Row Count
Nov 4, 2015 4:01:14 PM
Populate Sample
65
501

Configuring Analysis Projects (deprecated)

Analysis Projects analyze Sample data to assemble a set of predictors that have influence on the nominated outcome, or to conduct distribution analysis on a data set.

Pega Customer Decision Hub
Note: Starting in Pega Customer Decision Hub version 8.5, this functionality is considered deprecated and may be removed in future versions of the product.

1. Click **Campaign Segmentation Support Analysis Projects** .
   Note: If you do not see the **Campaign** menu in the Pega Customer Decision Hub, enable the Marketing Campaign Edition.
2. Click **Create**.
3. On the **Create Analysis Project** form, enter the following information:
   - Analysis Project name
   - Sample – Select a Sample from the list.
4. Click **Create and open**.
5. Configure the analysis type for the Analysis Project. The following analysis types are supported:
   - Configuring Static Analysis provides a convenient way to understand what the groups of data look like for a specific property and how many occurrences exist within each group.
   - Configuring Modeled Analysis provides a statistical assessment of the significance of the individual properties and their influence or power in predicting the targeted outcome.
6. Click **Save**.

7. Run the Analysis Project.
   - To run the Analysis Project manually, click the **Results** tab, and then click **Run**.
     - For Static Analysis, the **Run** button is in the **Static Results** section. If the **Run** button is inactive, check out the Analysis Project.
     - For Modeled Analysis, the **Run** button is in the **Modeled Results** section. If the **Run** button is inactive, check out the Analysis Project.
   - To run the Analysis Project automatically, schedule the Analysis Project execution in the **Schedule** tab. Analysis Project scheduling works in the same way as Segment scheduling. For more information, see Optional: Configuring Criteria Segment Runs.
8. Click the **Results** tab to review the results of manual and scheduled analysis runs.
9. **Optional:** Click the **History** tab to see the analysis execution statistics.

- Configuring Static Analysis
- Configuring Modeled Analysis
- Associating Analysis Projects with Segments

Configuring Static Analysis

You can configure the Analysis Project to use static Data Distribution analysis. This provides a convenient way to understand what the groups of data look like for a specific property and how many occurrences exist within each group.

Pega Customer Decision Hub
Static Analysis is useful as a first pass to describe how the data in our data source is organized and may give some insight as to where to focus any criteria construction in Segments.

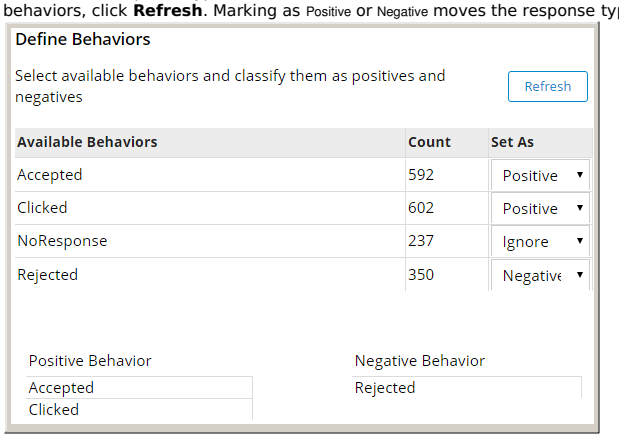
1. Click the **Static Analysis** tab of an Analysis Project.
2. Review the available properties which have been retrieved from the supporting sample.
3. Describe which grouping type is required for each property.
   - **Distinct Count** - This property will be processed during the execution phase and a distinct count of the grouping will be performed. Apply this grouping to properties which have a limited set of values and where knowing how many customers are in each group would be very valuable.
   - **No Grouping** - This property will be ignored. However, it may still be used as part of the modeled analysis. Apply this grouping type to properties which are likely to be a continuous value with a large number of groups and which are less useful from a grouping and distribution analysis perspective.
4. If the Sample fields changed since this Analysis Project was created, click **Refresh**.

Configuring Modeled Analysis

You can configure your Analysis Project to use modeled analysis. This approach uses statistical analysis to identify properties that have a high influence on the targeted outcome, completely removing the need for guess work or intuition when assembling criteria in a Segment.

Pega Customer Decision Hub
A key aspect of this analysis approach is the use of a closed loop feedback mechanism. Sample data is used to train or drive the creation of modeled analysis predictors. The sample data brings with it details of previous interactions and responses to actions made. While we are creating modeled analysis predictors that have predictive power over various outcomes, they are based on positive and negative responses to previous actions.

1. Click the **Modeled Analysis** tab of an Analysis Project.
2. In the **Select action** section, you can train the analysis on specific instances of responses to previously delivered actions. You can train the analysis using responses to previous actions for the same product or product group. Alternatively, for a new product offering, you can use similar products or product groups to assemble likely predictors that will help illustrate the type of customers who are likely to buy the new product.
   a. Select the **Issue**, **Group**, or **Action** on which you want to focus.
   b. Leave any of these values blank to bring through all entries for the parent category. For example, if you select the Sales Issue and the Handsets Group, and leave the action field blank, the analysis will consider the responses to all actions that belong under Sales or Handsets.
3. In the **Interaction Details** section, you can train the analysis on specific interaction details, in a similar way as with responses to actions. Select the **Direction**, **Treatment**, and **Channel** on which you want to focus. Use the **Offered After** field to restrict the analysis only to responses to those actions that were offered after a specified date and time.
4. In the **Select Predictors** section, remove or add properties from the base Sample which will be used as predictors for the analysis. Predictors which have no predictive value in modeled analysis should be removed from this section.
5. In the **Define Behaviors** section, select positive and negative outcomes for the modeled analysis based on the set of responses received so far.
   Review each response type in the **Available Behaviors** section and mark it as a positive, negative, or ignored behavior for the analysis. To reload the list of available behaviors, click **Refresh**. Marking as Positive or Negative moves the response type into the corresponding response category (**Positive Behavior** or **Negative Behavior**).



Define Behaviors

Select available behaviors and classify them as positives and negatives

Refresh

Available Behaviors	Count	Set As
Accepted	592	Positive
Clicked	602	Positive
NoResponse	237	Ignore
Rejected	350	Negative

Positive Behavior	Negative Behavior
Accepted	Rejected
Clicked	

Associating Analysis Projects with Segments

You can associate a Segment with an Analysis Project so that you can reuse static and modeled analysis predictors during criteria assembly. For more information about associating an Analysis Project with a Segment at Segment creation, see Creating a Criteria Segment.

Pega Customer Decision Hub
You can also associate Analysis Projects with existing Criteria Segments.

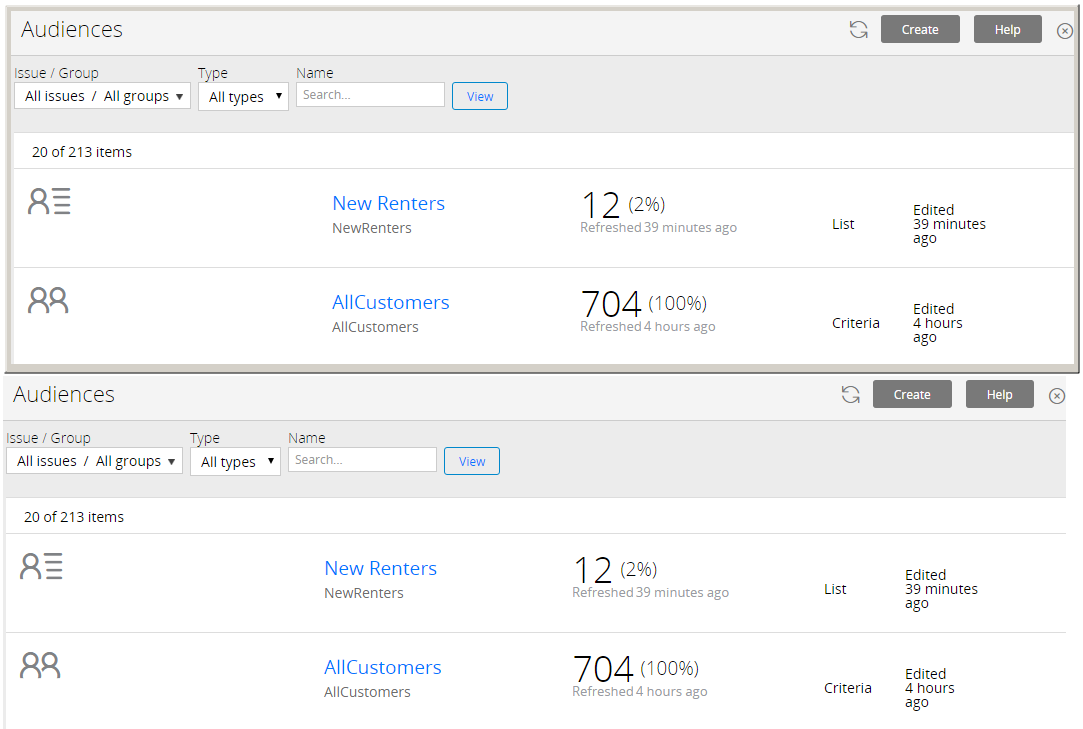
1. Click **Audiences Segments** .
2. Open the Criteria Segment which you want to associate with an Analysis Project.
3. Click **Check out**.
4. On the **Design** tab, click **More**.
5. In the **Analysis Project** list, select an Analysis Project which is relevant to your Segment. If the Criteria Segment was already associated with an Analysis Project, selecting a new Analysis Project will clear any existing criteria for this Segment.
6. Click **OK** to proceed or **Cancel** to cancel the change.
7. Optional: To view or modify the Analysis Project configuration, click the **Open** icon next to the **Analysis Project** list.
8. Click **Save**.

Managing Segments

You can view and manage List and Criteria Segments by using the Segments landing page.

Pega Customer Decision Hub

1. In the Pega Customer Decision Hub portal, click **Audiences Segments** .
2. For each Segment, review the following information:
   - Image associated with the Segment
   - Description
   - Rule name of the underlying Segment rule
   - Number of customer entities in the Segment**Note:** If your Pega Customer Decision Hub application is installed on a PostgreSQL database, this value shows the approximate customer count.
   - Percentage of the total population represented by the Segment
   - When the Segment population was last refreshed
   - Segment type
   - When the Segment rule was last updated



Tip: The landing page lists the twenty most recently updated Segments in the system. To load the next twenty results, click the **View more results** link at the end of the list.
3. To search for Segments by Issue / Group, Type, and Name, use the filters at the top of the page.
4. Click the link in the Segment description to open the Segment.

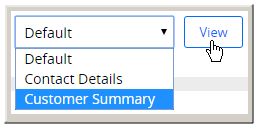
- Exporting Customers in a Segment
- Removing Customers from a List Segment
- Repopulating a Segment
- Reviewing Segment History Information

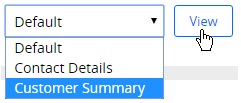
Exporting Customers in a Segment

After you populate a Segment, you can export a list of its unique customers.

Pega Customer Decision Hub

1. Use the customer grid to view the customers in the Segment.
   - For Criteria Segments, click the **Show customers** link in the Segment header panel. This launches a pop-up grid listing the customers in the Segment.
   - For List Segments, the same customer grid is available directly on the **Customer list** tab of the Segment.
2. The customer grid displays the built-in Default customer view. If your implementation team defined other views for the customer grid, you can switch to them by selecting the view in the list and clicking **View**.





Tip: To define a custom data view, create a Report Definition in the Customer- class that has a custom field named **DataManagement** and value Views.

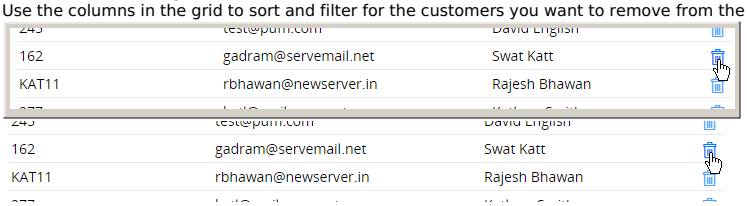
3. Export the data view to a CSV file.
   - For Criteria Segments, click **Export** in the pop-up window which displays the customer grid.
   - For List Segments, ensure that the Segment is checked in, and then click **Export** on the **Customer list** tab of the Segment.

Removing Customers from a List Segment

You can manually remove one or more customer record from a List Segment.

Pega Customer Decision Hub

1. Check out the List Segment.
2. Use the columns in the grid to sort and filter for the customers you want to remove from the Segment.



3. Click the **Delete** icon in the customer's row in the customer grid.
   Note: For Criteria Segments, this option is not available. To change which customers are included in a Criteria Segment, adjust the selection criteria instead. For more information, see Designing Criteria Segments.

Repopulating a Segment

After you make changes, such as adding customers or adjusting criteria, to a Segment, repopulate it to apply your changes.

Pega Customer Decision Hub

1. Check out the Segment and edit it as needed.
2. Click **Save**.
3. Check in the Segment.
   - For List Segments, checking in the Segment repopulates it automatically.
   - For Criteria Segments, click **Run** after checking in the Segment. Running a Criteria Segment repopulates it.

Reviewing Segment History Information

The **History** tab provides a historical view of the Segment, its updates and its population.

Pega Customer Decision Hub

1. Click **Audiences Segments** , and then click the name of a Segment.
2. Click the **History** tab.
3. For Criteria Segments, review statistics about the current segment run in the Current run section, and about previous Segment runs in the Segment Statistics section.
4. For List Segments, review details about previous uploads of files with customer data records in the Import History section.

Defining an exclusion list of customers who opted out of receiving notifications

Use exclusion lists to prevent Pega Customer Decision Hub from initiating outbound actions for a specific set of customer entities. For example, you can set up an exclusion list for customers who opted out of receiving notifications, customers who unsubscribed from your newsletter, or customers whose email addresses are invalid.

Pega Customer Decision Hub

If you specify an exclusion list of customers, Pega Customer Decision Hub checks it during action initiation and wait expiration for outbound channels. Actions for customers on the exclusion list are not initiated or continued.

Exclusion lists are not applied to actions that are initiated through real-time containers and inbound channels.

Managing prospects

In marketing terms, a prospect is a potential customer, which you want to target. Pega Customer Decision Hub provides support for importing prospect lists, creating prospect segments. If campaigns are enabled in your application, you can also use them to target prospects.

Pega Customer Decision Hub

Configuring the prospect class

Application administrators need to configure the class for storing prospects in the system. Refer to the *Pega Customer Decision Hub Implementation Guide* on the Pega Customer Decision Hub product page for detailed instructions.

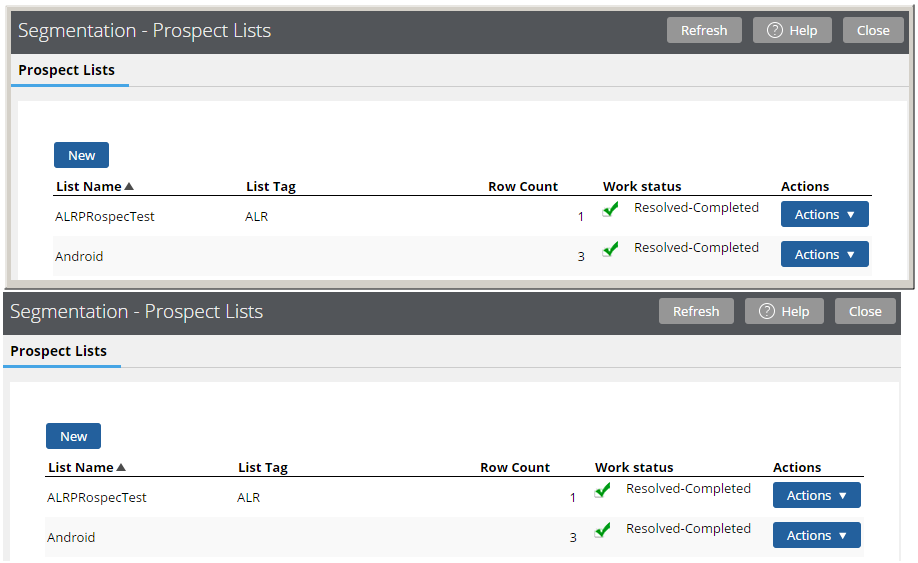
- Importing prospect lists
- Creating prospect segments
- Using campaigns to target potential customers

Importing prospect lists

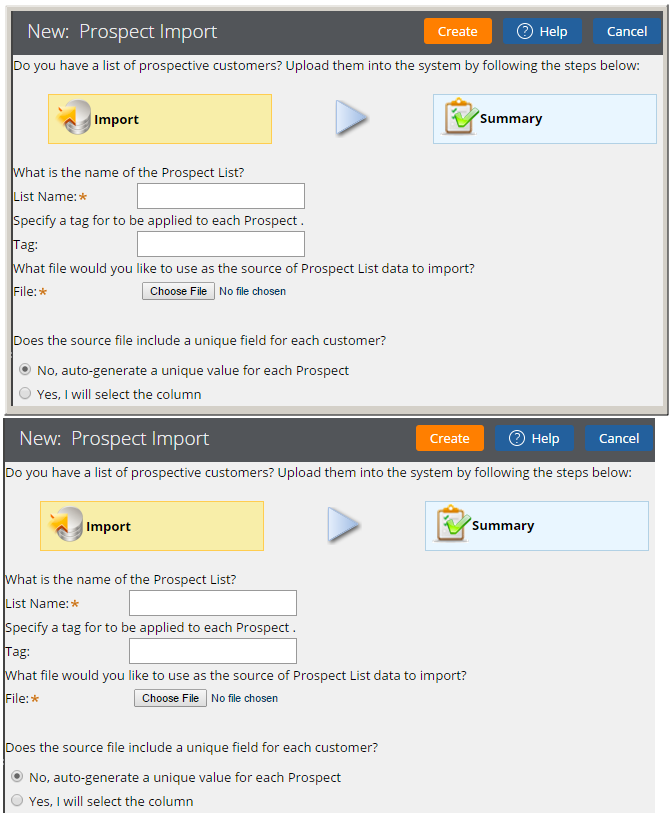
Prospect lists can be imported into the application via the Prospect Lists landing page. This landing page is accessible in the Pega Customer Decision Hub portal via: **Configuration Segmentation Prospect Lists** .

Pega Customer Decision Hub

This page lists the prospect lists that have already been imported into the system, along with their tags, row counts, and status. The **Open** action (in the **Actions** menu) can be used to view the import work object.



To import a new prospect list, Use the **New** button on the landing page. Clicking this button will initiate a new Prospect Import work object, as shown below:



Specify a unique name for the prospect list. Users can also specify a tag to apply to each prospect imported from the list. Tags can be used to group prospects imported from different prospect lists. Select the location of the prospect list xlsx file.

Next, select between the following options for determining the unique id of the prospect:

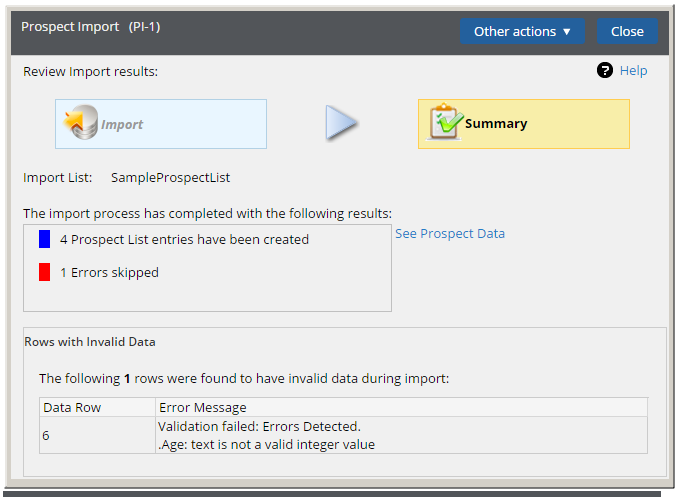
- **No, auto-generate a unique value for each Prospect**

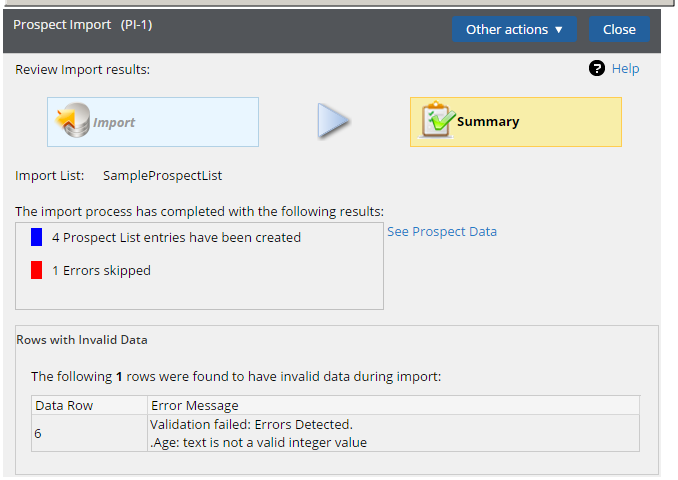
  Select this option to have the system generate the unique id for the prospect. The id will be a concatenation of the list name and a uniquely generated string.

- **Yes, I will select the column**

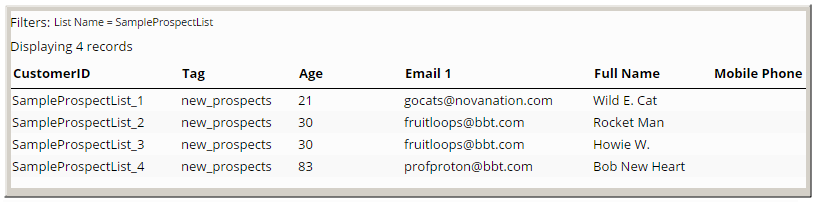
  Select this option to denote one of the fields in the prospect list spreadsheet as the unique field. Select this unique field from the list of available fields.

After configuring the necessary fields, click the **Create** button to proceed to the summary stage. Use the **Refresh** action from the **Other actions** menu to periodically check the status of the prospect import. This information is also available on the Prospect Lists landing page. The number of prospect entries created and errors detected are displayed along with details of the invalid rows.





Use the **See Prospect Data** link to view a report containing information about the prospect data that was imported. A sample report is shown below:



Filters: List Name = SampleProspectList

Displaying 4 records

CustomerID	Tag	Age	Email 1	Full Name	Mobile Phone
SampleProspectList_1	new_prospects	21	gocats@novanation.com	Wild E. Cat	
SampleProspectList_2	new_prospects	30	fruitloops@bbt.com	Rocket Man	
SampleProspectList_3	new_prospects	30	fruitloops@bbt.com	Howie W.	
SampleProspectList_4	new_prospects	83	profproton@bbt.com	Bob New Heart	

Filters: List Name = SampleProspectList

Displaying 4 records

CustomerID	Tag	Age	Email 1	Full Name	Mobile Phone
SampleProspectList_1	new_prospects	21	gocats@novanation.com	Wild E. Cat	
SampleProspectList_2	new_prospects	30	fruitloops@bbt.com	Rocket Man	
SampleProspectList_3	new_prospects	30	fruitloops@bbt.com	Howie W.	
SampleProspectList_4	new_prospects	83	profproton@bbt.com	Bob New Heart	

- Understanding the Prospect Import wizard

Understanding the Prospect Import wizard

Use the **New** button on the Prospect Lists tab of the Prospect Lists landing page to launch the Import Prospects wizard.

Pega Customer Decision Hub

The import process has the following two stages:

- Import
- Summary

Import

In this stage, provide the following information for the prospect list to import:

Field	Description
List Name	Enter a unique name for the prospect list.
Tag	Optional. Specify a tag to apply to each prospect in the list. You can use tags to group prospects imported from different prospect lists.
File	Use the **Browse...** button to select the prospect list ".xlsx" file.**Note:** The columns in the import spreadsheet must correlate with the properties on the prospect class. Use the prospect import template file provided by your application administrator as a template for your prospect lists.
	Choose from the following options for determining the unique id of the prospect:

Field	Description
(no label)	• **No, auto-generate a unique value for each Prospect** - Select this option to have the system generate the unique ID for the prospect. The ID will be a concatenation of the prospect list name and a uniquely generated string. • **Yes, I will select the column** - Select this option to denote one of the fields in the prospect list spreadsheet as the unique field. Additionally, select the **Unique column** from the list of available fields.

After configuring the necessary fields, click the **Create** button to initiate the import process and proceed to the summary stage.

Summary

In this stage, you can review the results of the import process. If the prospect list is large, use the **Refresh** action from the **Other actions** menu to refresh the summary results.

The system displays the following summary information:

- Number of prospect entries imported
- Number of erroneous entries skipped

In case of errors, the **Rows with Invalid Data** section displays the following information:

Field	Description
Data Row	Row number in the import spreadsheet of the erroneous record
Error Message	Error message explaining why the record could not be imported

Use the **See Prospect Data** link to view a report containing information about the prospect data that was imported.

Once in the summary stage, you can close the import wizard (if desired). To return to the wizard, use the **Open** action from the **Actions** menu of the prospect list entry on the Prospect Lists landing page. You can also double-click the prospect list entry to open the wizard.

Creating prospect segments

After you import a prospect list, Pega Customer Decision Hub automatically creates a segment for the prospects. You can also manually create new prospect segments. Before you can use these segments in campaigns, you must first run the segment to populate it.

Pega Customer Decision Hub

1. Click **Audiences Segments** .
2. Click **Create**.
3. Enter a **Description** for the new Segment.
4. In the **Context** section, select **Prospect**.
   Note: If you want to create a segment for existing customers instead, see Defining customer audiences with customer segments.
5. In the **Type** section, select **Criteria**.
   Note: The List segment type, available for regular segments, is not supported for prospect-based segments. However, after you import a prospect list as described in Importing prospect lists, you can use it as a criterion by selecting it in the **List Name** field.
6. Click **Create and Open**.
7. Add criteria groups to your segment.
   If you specified a prospect list, it is automatically included as a criterion. You can configure further criteria, as described in Designing Criteria Segments in the "Segments" chapter, with the following notes specific to prospect segments:
   - Analytical criteria (both modeled and static) are not available for prospect-based segments.
   - The data fields that are available as criteria are those that are defined on the Prospect class.
8. Click **Save**. The prospect segment is saved and checked in.
9. Click **Run** to populate the segment with the current set of customers.
   You can click **Stop** to stop the segment execution, or continue with other work while the segment populates. After Pega Customer Decision Hub finishes populating the segment, the total number of customers in this segment is displayed in the **Population count** field.
10. **Optional:** To automatically refresh your segment, configure a schedule. For more information, see Optional: Configuring Criteria Segment Runs in the "Segments" chapter.
11. **Optional:** To use your segments for prospecting on external ad platforms, configure the paid synchronization settings in the **Paid sync** tab. For more information, see Enabling segment updates on paid destinations in the "Paid Media Manager" chapter.

Using campaigns to target potential customers

To target prospects with your marketing efforts, create a multi-channel campaign. Self-optimizing and outbound campaigns do not support targeting prospect audiences.

Pega Customer Decision Hub

For general information about creating multi-channel campagins, see Multi-Channel Campaigns. The following special considerations apply when creating a campaign that targets prospects:

- You must select **Prospect** as the campaign context.
- When selecting the campaign audience, only prospect-based segments are available. However, the marketing strategy does not need to be specific to the Prospect class, so you can target prospects with strategies defined for your main Customer context.
- If the campaign actions include email or SMS treatments, you must ensure that the required email or mobile phone contact information is available in your prospects' profiles.

Seed Lists

A Seed List is a collection of artificial entities (also called “seeds”) that can be used to test, monitor, and validate various aspects of the processing infrastructure. For example, such a list can be used to test the timeliness and reliability of processes being handled by a third party.

Pega Customer Decision Hub

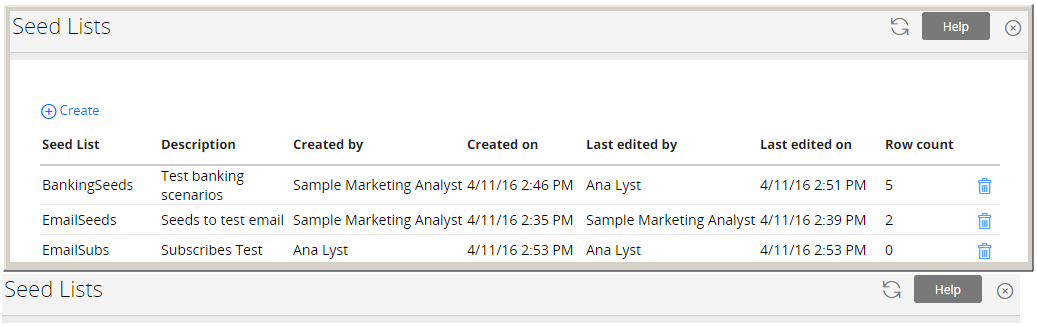
Each seed represents a make-believe customer and all applicable customer attributes can be specified for a seed. Multiple lists, each containing their own seed data, can be created and managed within the application. These lists can be used to test the following capabilities:

- Campaign execution
- Action execution
- Personalized email delivery
- Personalized SMS delivery

Each of these modes of testing is explained in the chapter pertaining to the entity being tested (e.g. campaigns, actions, etc.).

Seed Lists are listed under the Seed Lists landing page, which is accessible in the Pega Customer Decision Hub portal via: **Campaigns Seed Lists** .**Note:** If you do not see the **Campaign** menu in the Pega Customer Decision Hub, enable the Marketing Campaign Edition.

If you do not enable the Marketing Campaign Edition, you can access the Seed List landing page from the Action's **Test** tab, **Test Message** tab in Email treatment and **Test Message** tab in SMS treatment.



Seed Lists
Help
Create
Seed List
Description
Created by
Created on
Last edited by
Last edited on
Row count
BankingSeeds
Test banking scenarios
Sample Marketing Analyst
4/11/16 2:46 PM
Ana Lyst
4/11/16 2:51 PM
5
EmailSeeds
Seeds to test email
Sample Marketing Analyst
4/11/16 2:35 PM
Sample Marketing Analyst
4/11/16 2:39 PM
2
EmailSubs
Subscribes Test
Ana Lyst
4/11/16 2:53 PM
Ana Lyst
4/11/16 2:53 PM
0

The following actions are available on the Seed Lists landing page:

- **Create** – Create a new Seed List. See Creating a new Seed List.
- Refresh icon – Reload the Seed List grid
- **Help** – View context-sensitive help
- Close icon – Close the landing page

- Seed List Management
- Updating a seed list

Seed List Management

This section explains the various actions that can be performed on a Seed List.

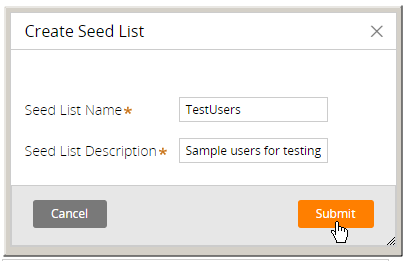
Pega Customer Decision Hub

- Creating a new seed list
- Modifying Seeds in a Seed List
- Deleting a Seed List

Creating a new seed list

To create a new Seed List, click the **Create** link on the landing page. Enter a unique name for the Seed List and a helpful description of its contents.

Pega Customer Decision Hub



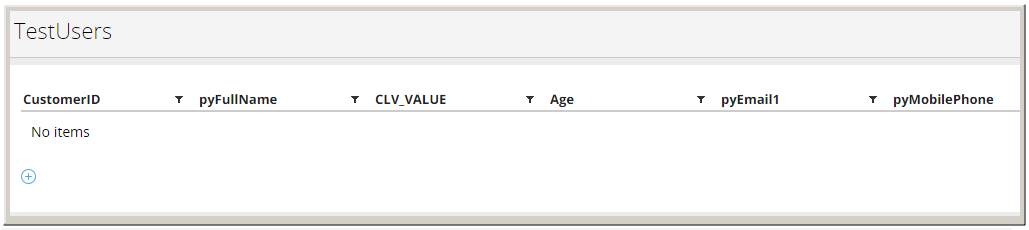
Create Seed List
Seed List Name*
TestUsers
Seed List Description*
Sample users for testing
Cancel
Submit

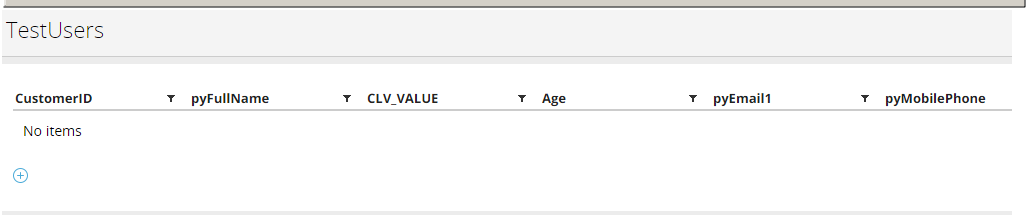
Click **Submit** to create the Seed List. At this point, the backing structure of the Seed List is created. The landing page grid should refresh and display the newly created Seed List. The row count column for the new Seed List will initially be 0.

Modifying Seeds in a Seed List

Once a Seed List has been created, new seeds can be added to it and existing seeds can be modified or deleted. To do so, double click a Seed List row in the grid. This will open the Seed List editor, as show below:

Pega Customer Decision Hub





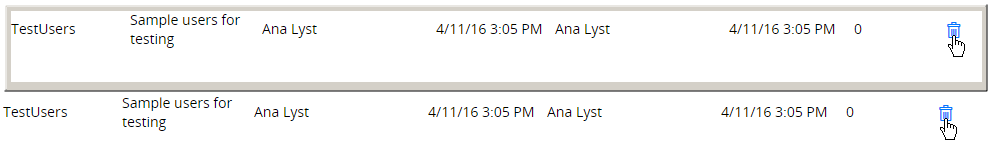
This editor displays all entries (seeds) as rows in the Seed List. The columns of the Seed List are made up of the columns from the customer table.

To..	Do...
Add a new seed	Click the add icon. Fill in all necessary fields and click anywhere outside the input row to save the record. To cancel your changes and discard the new record, click the delete icon at the right end of the row.
Modify an existing seed	Click anywhere in the desired row. The row should toggle to editable mode. After making the desired changes, click anywhere outside the row to save your changes.
Delete an existing seed	Click the delete icon at the right end of the row.

Deleting a Seed List

Existing Seed Lists can be deleted by using the row-level delete icon in the grid on the Seed Lists landing page.

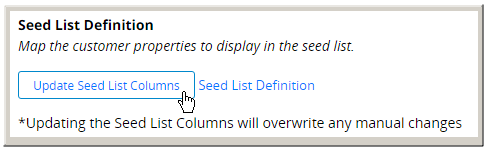
Pega Customer Decision Hub

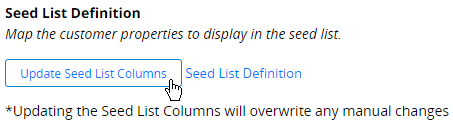


Updating a seed list

All Seed Lists share a common structure (set of columns) and this structure is derived from the columns present on the current customer class, which is in use in the application. In time, the structure of the customer class might deviate from what was initially set up for use with Seed Lists. In a situation like this, administrators can synchronize the Seed List structure to the customer class structure via the **Update Seed List Columns** button on the Manage Data Relationships tab of the Application Settings landing page.

Pega Customer Decision Hub





Upon clicking this button, the backing Seed List structure will be synchronized with the customer class.

This task should typically be performed by the administrator and will impact all existing Seed Lists. This task should be performed in the following scenarios:

- Initially after a new install
- Whenever the customer class changes
- When attributes are added to (or removed from) the customer class and these changes need to be reflected in the seed data

The administrator can also use the **Seed List Definition** link to view the current Seed List structure. Clicking this link opens the SeedListPopulationRD Report Definition, which represents the structure utilized by the Seed List editor. If desired, administrators can directly modify this Report Definition to update the columns visible in the Seed List editor. However, any manual updates to this Report Definition will be overwritten when the Update Seed List Columns button is used on the landing page.

Predicting customer behavior

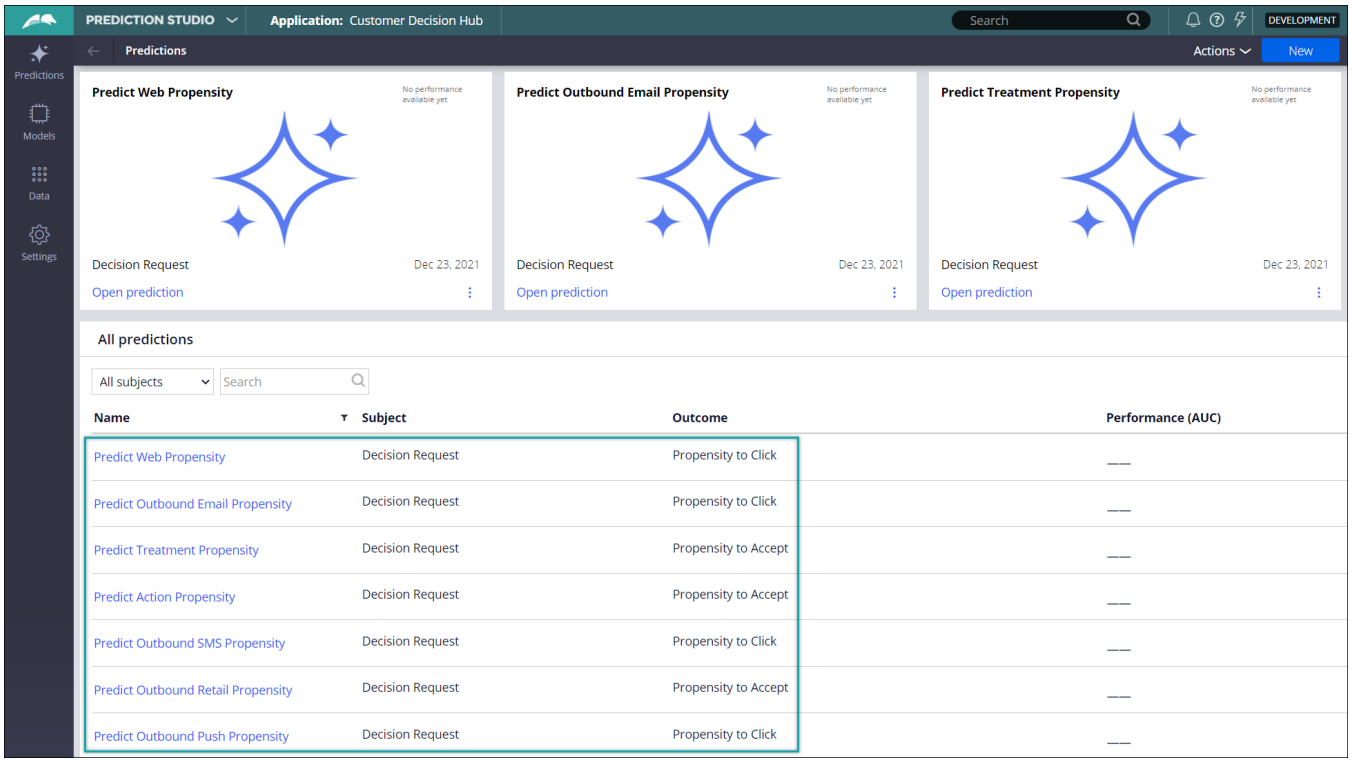
Address your customers' needs better by predicting customer behavior with predictions. For example, you can determine the probability of a customer accepting an offer or the likelihood of a customer discontinuing a subscription (churn).

Pega Customer Decision Hub

Default predictions

The most common predictions are built into Pega Customer Decision Hub as part of the Next-Best-Action Designer strategy framework, for example, Predict Action Propensity and Predict Treatment Propensity. Predictions are strategies driven by adaptive and predictive models. Predictions are used in decision strategies to predict outcomes based on customer characteristics and historical customer behavior, such as offer acceptance or web banner clicks.

Standard predictions built into Pega Customer Decision Hub



Custom predictions

To extend your next-best-action strategy framework, you can create your own predictions. Prediction Studio provides a range of prediction templates with preconfigured outcomes for customer engagement. For more information, see Creating predictions for customer engagement.

MLOps

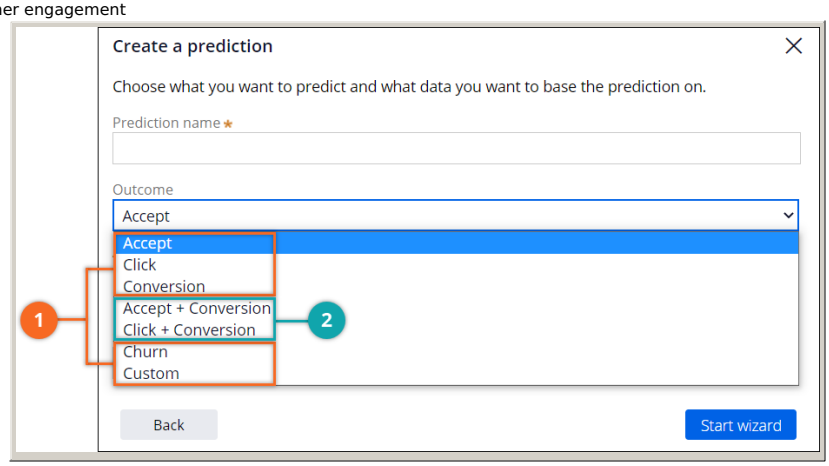
Replacing a model in a prediction is possible at any time in the prediction's life cycle, through Machine Learning Operations (MLOps). As a data scientist, you can add and approve the new model in a non-production environment, and start a deployment process to migrate the new model to production.

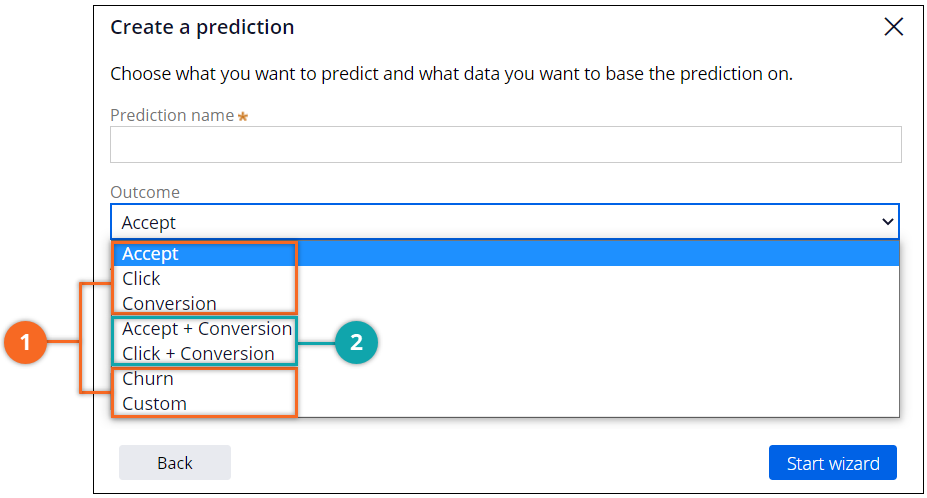
Creating predictions for customer engagement

Optimize customer engagement by creating predictions that predict customer behavior, such as web clicks, conversions, or the probability of customer discontinuing a service (churn). By adding predictions to your next-best-action strategy framework, you can determine the next best actions for your customers.

As part of the next-best-action strategy framework, Pega Customer Decision Hub includes several default predictions that cover the most common uses cases. If you want to extend your next-best-action strategy framework, you can create additional predictions in Prediction Studio, and add them to extension strategies. Prediction Studio provides several templates with preconfigured outcomes to help you create predictions for customer engagement:

Prediction templates for customer engagement





Depending on your use case, you can predict one or two outcomes:

1. Single-stage prediction
   A single-stage prediction predicts a single outcome, for example, how likely a customer is to click a banner or accept an offer, to convert, or to stop using your service (churn).
2. Multistage predictions
   A multistage prediction predicts two outcomes that occur in a sequence, for example, a web banner click followed by a conversion. A multistage prediction requires access to conversion data and long-term feedback. This type of prediction can be useful when groups of customers who click or accept an offer and those who subsequently convert have distinct characteristics.

Choosing between single-stage and multistage predictions

To decide whether a single-stage or a multistage prediction is the better choice for you, your team should answer the following questions regarding your customers, data, and storage:

Role	Question	Guidance
Data scientist	Are the customers who click and the customers who convert similar or different?	Conduct a data science analysis to compare the profiles of both groups: • If the profiles are similar (except for volumes), you can predict conversions with a single-stage prediction. Clicks are often a good proxy of conversions, so you can use a click model. Alternatively, if you have access to conversion data, you can use a conversion model. • If the profiles are significantly different, consider using a two-stage prediction to assess which customers out of those who are likely to click an ad are also likely to convert.
System architect	Do you have access to conversion data?	You might have to retrieve conversion data from another system, for example, a fulfillment system. This might pose technical challenges, such as a longer waiting period.
	Can you retain impression data until customers convert?	In systems that process large volumes of data, it might not be feasible to keep the impression data long enough for customers to convert.

To create a prediction, you answer several questions about what you want to predict. Based on your answers, Prediction Studio creates a self-learning adaptive model that is the basis of the prediction. You can replace this model with a different one, for example, a predictive model that you developed in a third-party machine learning platform. You can then include the prediction in your decision strategy.

For example, you can create a prediction that calculates the probability of customer churn, and then add the prediction to a next-best-action strategy. The next-best-action strategy arbitrates several actions to retain the customer, and then selects the one that the customer is most likely to accept.

Pega Customer Decision Hub

1. In the navigation pane of Prediction Studio, click **Predictions**.
2. In the header of the **Predictions** work area, click **New**.
3. In the **Create a prediction** window, select **Customer Decision Hub**, and then click **Next**.
4. In the **Prediction name** field, enter a name for the prediction.
5. Specify what you want to predict. **For example:**

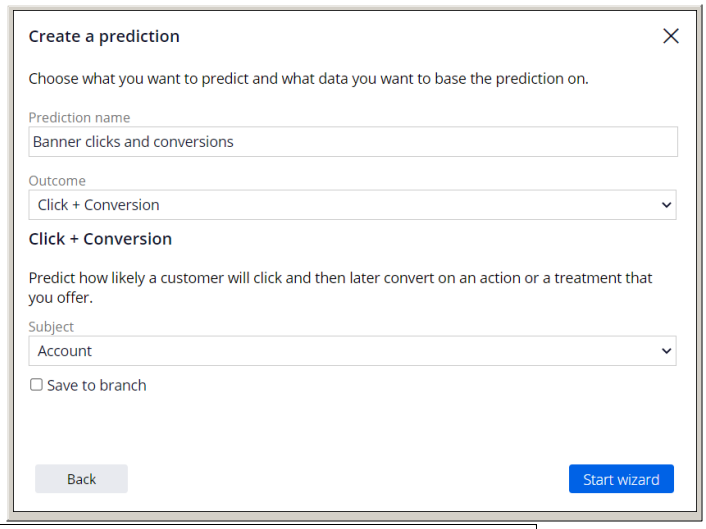
   To predict whether a customer is likely to accept an offer, select the following settings:

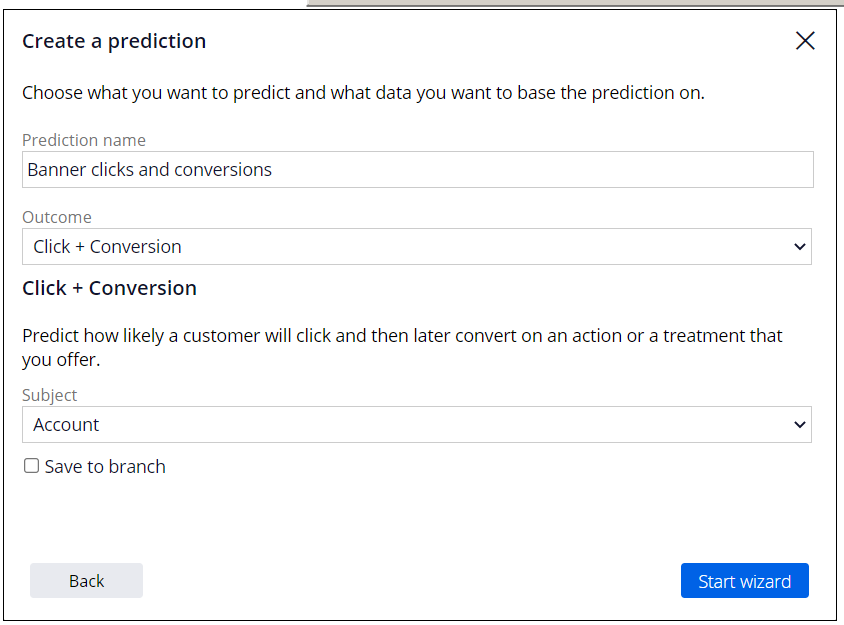
   1. In the **Outcome** list, select **Accept**.
   2. In the **Subject** list, select **Customer**.

   To predict whether a customer who is likely to click a web banner is also likely to accept the corresponding offer and convert, select the following settings:

   1. In the **Outcome** list, select **Click + Conversion**.
   2. In the **Subject** list, select **Account**.**Note:** A multistage prediction requires long-term feedback. You can define how long to wait for the customer to respond to your offer by configuring the response time-out in the prediction settings.

      Creating a multistage prediction





To predict an outcome that does not match any of the available templates, select the following settings:

1. In the **Outcome** list, select **Custom**.
2. In the **Outcome name** field, enter a name for the outcome, for example, Upsell sensitivity.
3. In the **Subject** list, select the subject of the prediction.

6. If you work with branches and want to save the prediction to a specific branch, select **Save to branch**, and then select a rule set and a branch.
7. Review your settings, and perform one of the following actions:
   - If you selected **Churn** as the outcome that you want to predict, click **Create**, and then go to step 14.
   - If you selected other outcomes, start the **Prediction** wizard by clicking **Start wizard**.
8. In the **Select data** wizard step, choose one of the following options:
   - To create a prediction without historical data, select **I do not have historical data**.

     In most cases, historical data is not available or necessary. You can continue without it.

   - To create a prediction with historical data, select **I have historical data**, and then select the data set that contains your historical data.**Note:**

     Historical data informs the prediction so that it performs better right from the start. To use this option, you need to import your historical data into a data set. Ensure that the data set contains a *pyOutcome* property for the outcome of each customer interaction, as well as properties that describe the context of each interaction, for example, *pyIssue*, *pyGroup*, *pyName*, *pyDirection*, and *pyChannel*. For more information, see Importing data into a data set.

9. Click **Next**.
10. In the **Prediction configuration** wizard step, review the response labels for the prediction, and then click **Next**.
11. In the **Select predictors** wizard step, select the fields that you want to use as input for the prediction.
    To increase the accuracy of your prediction, select a wide range of fields to use as predictors. Do not include fields that are not suitable as predictors, for example, the **Identifier** and **Date Time** fields. For more information, see Best practices for choosing predictors.
12. Confirm your choice of predictors by clicking **Next**.
13. In the **Review prediction** wizard step, review the prediction settings, and then complete the model creation process by clicking **Create**.
14. To change the prediction properties at this stage, in the **Prediction** window, click **Configure**, and then make your changes.
15. Click **Save**. **Result:** The prediction is now available in the **Predictions** workspace. Depending on the settings that you selected, Prediction Studio created an adaptive model or a scorecard as the basis of the prediction.
16. If you are using parameterized predictors, add them to the adaptive model that is the basis of the prediction.
    For more information, see Adding parameterized predictors. **Tip:** To open the model that is the basis of the prediction, in the **Prediction** workspace, on the **Models** tab, click the name of the corresponding model.
17. **Optional:** To customize your prediction, for example, by changing the control group settings or by configuring the response time-out, see Customizing predictions.
18. **Optional:** To replace the model that is the basis of the prediction with a different model, see Replacing models in predictions with MLOps.

What to do next: Include the prediction in an extension strategy within the next-best-action strategy framework, for example, *AIModelsExt* or *ActionLevelPropensityExt*. For more information, see NBA Strategy framework extension points.

- Creating predictions for customer engagement

Customizing predictions

Customize the predictions that you created in Prediction Studio. For example, define criteria for the control group to measure the accuracy of your prediction or to specify the response time-out for your offers.

You can configure the following prediction aspects:

Control group
: The control group is a group of customers who receive random offers, as opposed to the test group. Customers in the test group receive offers that they are most likely to accept, based on the propensity that the prediction calculates for each customer. The prediction calculates lift by comparing the success rate in the control group with the success rate in the test group. Based on the lift, you can determine the effectiveness of your prediction, for example, in increasing conversion rates.

  You can either define the control group as a percentage of all customers (the customers are then chosen randomly) or choose the customers for the control group yourself by assigning a specific field value to the customers that you want to include in the control group.

Response labels
: The response labels represent the possible outcomes of a prediction.

Response timeout
: The response time-out defines how long to wait for the customer to respond to your offer. After the specified amount of time elapses, the system automatically records the alternative outcome for the interaction.

  You can also choose to turn off the response time-out.

Pega Customer Decision Hub

Customize your prediction by performing the following steps:

1. In the navigation pane of Prediction Studio, click **Predictions**.
2. For the prediction that you want to configure, click **Open prediction**.
3. In the **Prediction** workspace, click the **Settings** tab.
4. Configure one or more of the following settings:

Choices	Actions
Define the control group as a percentage of all customers	In the **Control group** section, select **Percentage**, and then specify the percentage value. For example, if you want to randomly select customers that constitute 2% of all your customers, in the **Percentage** field, enter 2.0.
Define the control group as a group of customers with a specific field value	In the **Control group** section, select **Field**, and then select an available field and determine its value. For example, in the **Field** field, enter control_group and in the **equal to** field, enter yes.
Edit the response labels	In the **Response labels** section, enter a name for the target and alternative response. For example, for a prediction that calculates whether a customer is likely to click a web banner, in the **Target label** field, enter Click, and in the **Alternative label** field, enter NoResponse.
Set a response timeout	In the **Response timeout** section, click **Fixed time frame**, and then specify how long to wait for a customer to respond to your offer before Prediction Studio records the alternative outcome for that interaction. For example, if you set the response time-out to one minute, and a customer sees a web banner and does not click the banner for one minute, Prediction Studio records the NoResponse outcome for the interaction.
Turn off the response timeout	In the **Response timeout** section, click **Indefinitely**.

5. Click **Save**.

Monitoring predictions

Analyze how successful your predictions are in predicting the outcomes that bring value to your business. Gain insights by reviewing performance charts for predictions and the models that drive them.

Note: This procedure applies to predictions that are used in customer engagement and case management. For information about analyzing text predictions, see Monitoring text predictions.

Pega Customer Decision Hub

If a prediction uses more than one model, you can select the outcome for which you want to see performance statistics, for example, clicks, conversion, or clicks and conversion combined.

1. In the navigation pane of Prediction Studio, click **Predictions**.
2. From the list of predictions, open a prediction that you want to analyze.
3. On the **Analysis** tab, click **Prediction**.
   a. In the **Outcomes** field, select the outcome to analyze.
      If a prediction predicts a single outcome (single-stage prediction), the outcome is already selected. If a prediction predicts outcomes that occur in a sequence (multistage prediction), you can select each individual outcome or the two outcomes combined.
      For example: For a multistage prediction that predicts whether a customer who is likely to click a web banner is also likely to accept the corresponding offer and convert, you can select from the following outcomes:
      - **Clicks**
      - **Conversion**
      - **Clicks + Conversion**

   b. In the **Time frame** field, select the period that you want to analyze.

      The charts show the performance measures that are relevant to the selected outcome. For example, a performance analysis with regard to churn covers the churn rate, lift, AUC, and total cases, as in the following figure:

      Prediction analysis

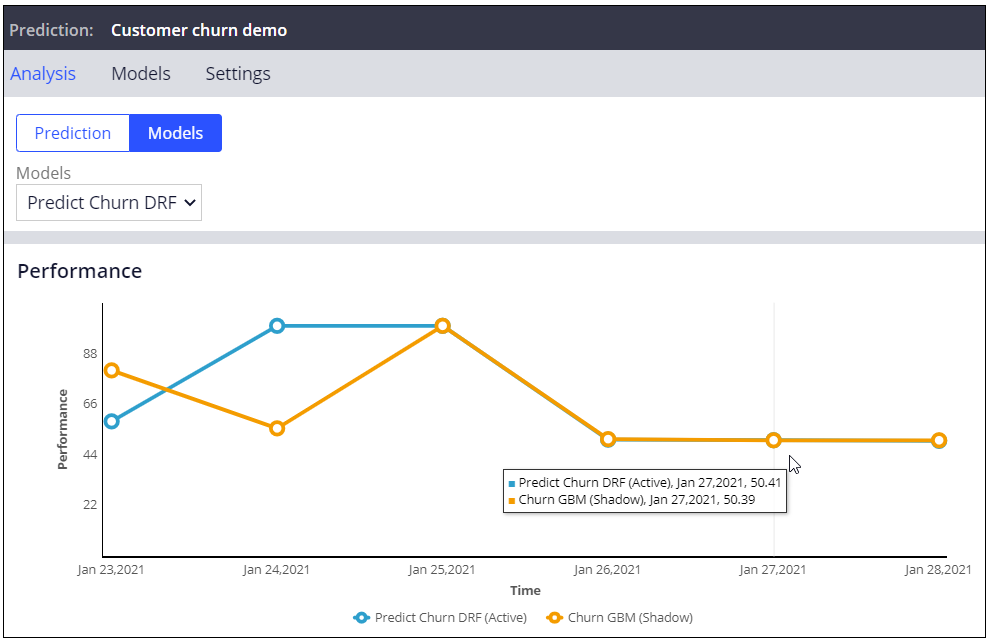


4. Click the **Models** tab.
5. If more that one model drives the prediction, in the **Models** field, select the model the performance of which you want to analyze.

   The performance chart shows how effective the model is over time. If a shadow model runs alongside an active model, the chart displays the performance curves for each model in the active-shadow pair. The comparison can help you determine which model is more effective and decide whether you want to promote the shadow model to the active model position. The analysis is available for both outcome-based and supporting models.

   The following figure shows the performance chart for the active model, Predict Churn DRF. This model is paired with a shadow model, Churn GBM. The chart shows two distinct lines that illustrate the performance of each model over time. Hover over the lines with your cursor to see the exact scores for each model at different times. In this example, the two lines converge around January 27, suggesting that the models performed equally well. The detailed information on mouse hover shows that the active model had a slightly higher score.

   A model performance chart in a prediction



- Replacing a model
- Promoting shadow models with MLOps

Updating active models in predictions with MLOps

As a data scientist, you can use Machine Learning Operations (MLOps) to approve changes to models that are used in predictions for deployment to the production environment. You can change models independently or by responding to a Prediction Studio notification that a prediction does not generate enough lift.

To improve the performance of a prediction, you can replace a low-performing model with a high-accuracy external model that you upload to a Pega repository or directly to Prediction Studio. As a result, you start a standard approval and validation process to deploy the model update to production. Before you approve any changes, you can compare the candidate model with the existing model based on data science metrics, such as score distribution or lift. For more information, see Active and candidate model comparison charts.

Pega Customer Decision Hub

Managing model updates

In your Business Operations Environment (BOE), you can start and manage the model update process from Prediction Studio or remotely by using the Prediction Studio API. For more information about using the API endpoints, see Updating active models in predictions through API with MLOps.

Model deployment

In Pega Customer Decision Hub environments, changes to models that you approve in Prediction Studio are deployed to production through Pega 1:1 Operations Manager and the Business Change pipeline.

For more information, see Understanding MLOps.

To replace a model in a prediction, and then deploy the model to production, perform the following procedures:

1. MLOps prerequisites
2. Replacing models in predictions with MLOps
3. Evaluating candidate models with MLOps
4. Promoting shadow models with MLOps
5. Rejecting shadow models with MLOps

MLOps prerequisites

To start updating models in your system with Machine Learning Operations (MLOps), configure the appropriate access rights for data scientist operators, update Prediction Studio settings, and ensure that your Business Change pipeline meets the requirements.

Pega Customer Decision Hub

Data scientist operators

Configure the following settings for the operator IDs of the data scientists who update models:

- Ensure that the operator's default access context is the same as the application context in which the predictions are updated.

  For example, if the prediction is updated in the *CDHSample* application, then the default access group for the operator must be *CDHSample*.

  For more information, see Defining operator contact information and application access.

- **Optional:** If you want to enable email notifications for the model update process:
  - Add the notification email to the operator ID.
  - Configure the operator ID with a default access group that includes the *PegaRULES:PegaAPI* access role.

    For example, if the default access group is *CDHSample:Administrators*, then that access group must include the *PegaRULES:PegaAPI* access role.

    For more information, see Defining work routing settings for an operator.

Prediction Studio settings

Enable the model update feature by configuring the following artifacts, and then adding them to the Prediction Studio settings:

- An analytics repository for storing machine learning models.
- **Optional:** A work queue for data scientists if you want to use a different work queue than the default *DataScientistWorkqueue*.
- **Optional:** An email account if you want to enable email notifications about model changes.

For more information, see Enabling Machine Learning Operations.

Business Change pipeline

Ensure that the Business Change pipeline meets the following requirements:

- All instances (Development, Sandbox, BOE, Production) are on the same versions of Pega Platform and Pega Customer Decision Hub.
- Pega 1:1 Operations Manager and Prediction Studio are on the Business Operations Environment (BOE) instance of your application.
- The models have the same names in all instances.
- The predictors for the models are configured in all instances and are in sync.

Note: You can also synchronize the model names, predictors, and properties through the Enterprise Change pipeline. For more information, see Creating a pipeline.

- Replacing models in predictions with MLOps
- Updating active models in predictions with MLOps
- Updating active models in predictions through API with MLOps

Replacing models in predictions with MLOps

Improve a prediction by replacing a low-performing model with a high-accuracy model. You can also replace a model with a scorecard or a field in the data model that contains a score.

For example, in a customer engagement context, you can replace a scorecard that is used to predict churn with a PMML or H2O machine learning model.

You can replace both the outcome-based and supporting models in predictions.

Replacing models models is part of the Machine Learning Operations (MLOps) of Pega Platform.

Pega Customer Decision Hub
Before you begin:

Ensure that Prediction Studio and your Business Change pipeline are configured to support the model update feature. For a list of prerequisites, see Updating active models in predictions with MLOps.

Plan your changes by checking which model types you can use to replace your current models, and which mode you can use to deploy the new models (shadow mode, replace, direct replace). For more information, see Active and candidate model type combinations.

1. Log in to your application:
   a. Log in as a data scientist to the Business Operations Environment (BOE).
   b. In the header of Dev Studio, click the name of your current application, and then click **Switch Application**.
   c. From the list, select the overlay application.
2. Switch the portal to Prediction Studio.
3. In the navigation pane of Prediction Studio, click **Predictions**.
4. From the list of predictions, open a prediction that you want to change, and then click the **Models** tab.
5. To the right of the model that you want to change, click the **More** icon, and then click **Replace model**.
   - If the active model that you want to replace is a scorecard or field, go to 6.
   - If the active model that you want to replace is an adaptive or predictive model, go to 7.
6. In the **Replace model** dialog box, choose the entity with which you want to replace the active model:

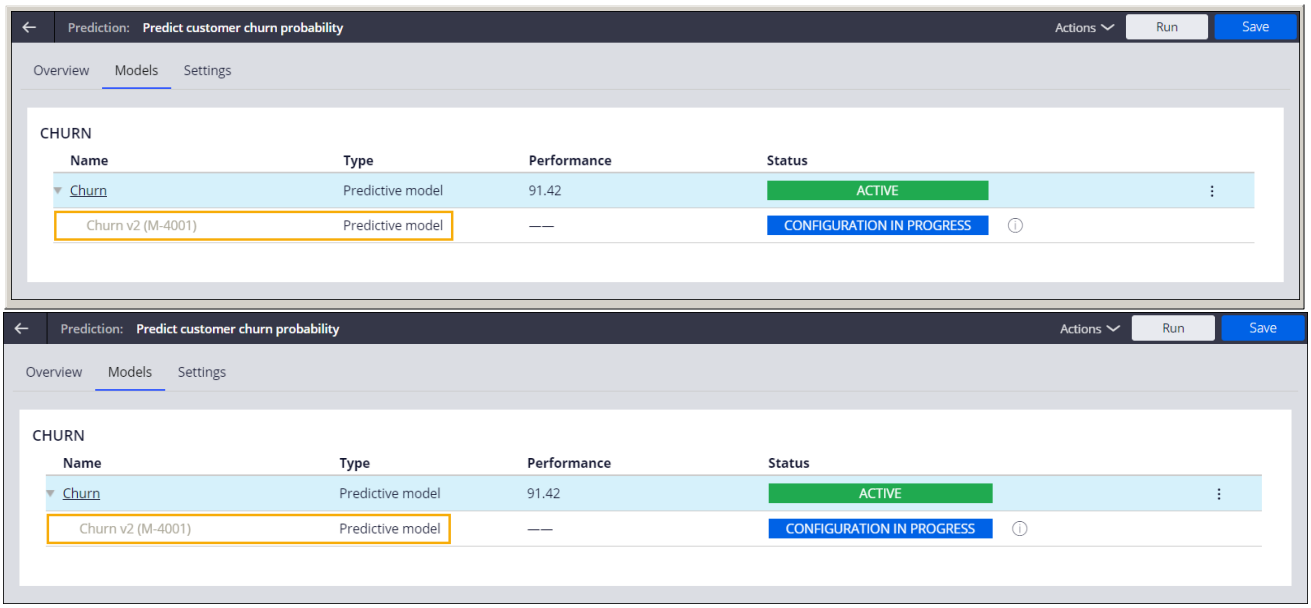
Options	Actions
Replace with a machine learning model	Click **Model**, and then go to 7.
Replace with a scorecard	a. Select **Scorecard**, and then click **Next**. b. Select the scorecard to directly replace the current model, and then click **Replace**. **Result:** The scorecard replaces the current model. This is the end of the procedure.
Replace with a field in the data model	a. Select **Field**, and then click **Next**. b. Select a field that you want to use to predict outcomes, and then click **Replace**. **Result:** The field replaces the current model. This is the end of the procedure.

7. **Optional:** If you want to use a validation data set to compare the active model and the candidate model, in the **Replace model** dialog box, select the **Compare the models** field.
   If you clear the **Compare the models** field, the **Replace model** window will not prompt you to select a data set for the comparison. You can add a validation data set later when you evaluate the model.
8. Select the source of the new model to replace the current model:

Choices	Actions
Use a file-based model	a. Click **Upload**. b. In the **Select a PMML, H2O MOJO or Pega OXL file** section, add a model to replace the current model.
Use a cloud-based model	a. Click **Machine learning service**. b. In the **Machine learning service** field, select the machine learning service from which you want to run the model. Pega Platform currently supports Google AI Platform and Amazon SageMaker models. c. In the **Model** field, select a model to replace the current model. d. **Optional:** In the **Upload model metadata file** section, add a metadata file with input mapping and outcome categories for the model. You can create a metadata file by clicking **Download template** and mapping the JSON input fields to Pega Platform fields in the template. For information about the metadata file properties and the available values, see Metadata file specification for predictive models.
Choose a model from Pega Platform	a. Click **Model list**. b. Select a model to replace the current model.

9. **Optional:** In the **Add model documentation (optional)** section, upload a documentation file for the model.
10. Click **Next**.
11. If you selected the **Compare the models** field in step 7, add a validation data set and the outcome column to compare the two models, and then click **Next**.
    The outcome column contains the response labels for the outcome.
12. In the **Summary** section, in the **Model name** field, enter a name for the new model, and then click **Add model**. **Result:** You can view the new model by expanding the twist arrow to the left of the current model. The new model name includes the number of the branch that contains the model change, for example, Validate Churn (M-7012).

A candidate model in a prediction



13. Wait a few seconds for the analysis of the new model to complete. If necessary, refresh the page by clicking **Actions Refresh** .

Result: The model update work group gets notification emails when the candidate model has any of the following statuses:

- CONFIGURATION FAILED
- VALIDATION FAILED
- READY FOR REVIEW

If the configuration of artifacts or validation of predictors fails, open the model by clicking its name, and read the error message in the **Model comparison** window. Resolve the error and resume the analysis or reject the model. For more information, see Model update statuses and notifications.

When the configuration and validation are successful, the model is ready for review.

What to do next: Evaluate the candidate model. For more information, see Evaluating candidate models with MLOps.

Evaluating candidate models with MLOps

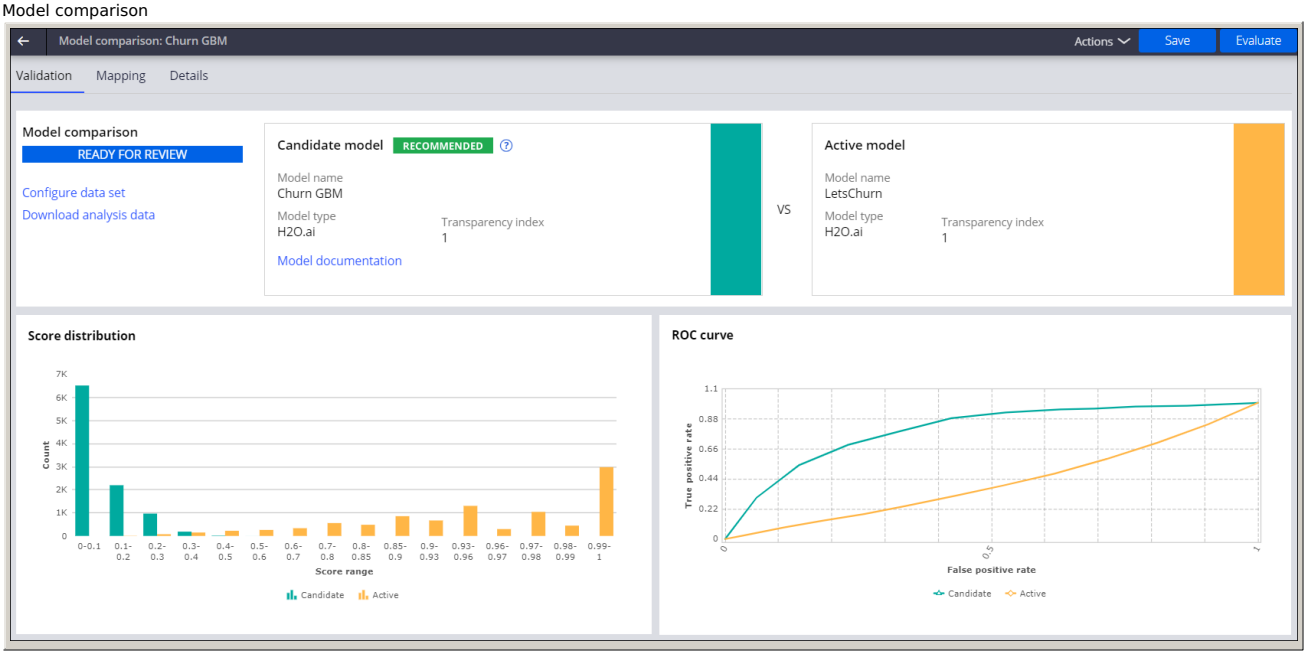
After you add a candidate model to a prediction, Prediction Studio configures and validates the new model, and provides comparison data to help you evaluate the new model. Decide whether you want to approve the new model for deployment to production or reject it.

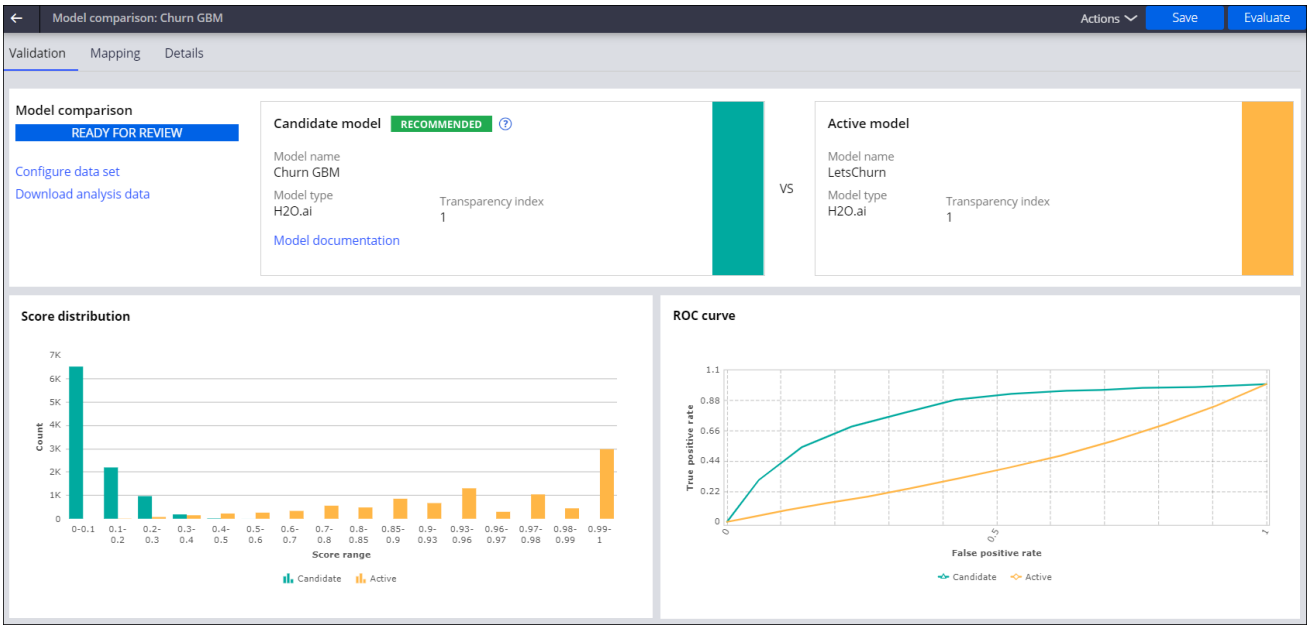
Model evaluation is part of the Machine Learning Operations (MLOps) feature of Pega Platform.

Pega Customer Decision Hub

1. In the navigation pane of Prediction Studio, click **Predictions**.
2. From the list of predictions, open a prediction that contains a candidate model, and then click the **Models** tab.
3. Display the candidate model by expanding the twist arrow to the left of the current model.
4. Ensure that the status of the candidate model is READY FOR REVIEW. If the configuration or validation of the model is still in progress, wait several seconds, and then refresh the page by clicking **Actions Refresh** .
5. Open the candidate model by clicking its name. **Result:** In the **Model comparison** window, the **Validation** tab displays a basic comparison of the two models. Prediction Studio marks the better model as "Recommended" based on the receiver operating characteristic (ROC) and transparency. If a data set was added for use in the comparison, Prediction Studio generates charts that show how the two models compare in terms of score distribution, ROC, lift, and gains. For more information, see Active and candidate model comparison charts.

Model comparison





6. **Optional:** If you want to add a data set for the comparison or use a different data set, click **Configure data set**.
   a. In the **Validation data set** window, select the data set and the outcome column to compare the two models.
      The outcome column contains the response labels for the outcome.
   b. Review the properties in the data set and their values.
      In the **Records** field, you can select the number of records for the data set preview. The values are recalculated after each change.
   c. Click **Submit**.
   d. In the top-right corner of the **Model comparison** window, click **Save**.
   e. In the **Confirm save** dialog box, click **Yes**. **Result:** You return to the **Prediction** window.
   f. Wait several seconds for the validation to complete, and then refresh the page by clicking **Actions Refresh** .
   g. When the status of the model changes to READY FOR REVIEW, open the model by clicking its name, and review the new comparison charts.
7. Review the comparison charts to decide which model is better for your use case.
8. **Optional:** Export the analysis data to a CSV file by clicking **Download analysis data**.
9. In the top-right corner, click **Evaluate**.
10. In the evaluation window, approve or reject the model:
    - To let the new model shadow the current model, click **Approve new candidate model and start shadowing (recommended)**.
    - To replace the current model, click **Approve candidate model and replace current active model**.
    - To reject the new model, select **Reject candidate model**.

    Note: Shadowing means that the new model receives production data and the system tracks the outcomes of the model, but does not use the outcomes to make business decisions. This option is only available for predictive models. For more information, see Active and candidate model type combinations.

    Approving a candidate model deletes other candidate models that are associated with the active model.

    A rejected model and its related assets are archived. You can view them in the model history.

11. In the **Reason** field, provide a comment on the approval or rejection.
12. Click **Save**.

Result:

In Pega Customer Decision Hub environments, if you approve a model update, the system automatically creates and resolves a *Change prediction* request in Pega 1:1 Operations Manager. A revision manager can then deploy the model to production.

What to do next:

When the deployment is complete, log in to the production instance and verify that the model is updated.

If you approved a predictive model in shadow mode, review its performance after a period of time. For more information, see Monitoring predictions.

Based on your analysis and your business needs, you can decide to promote the model as the active model or reject it. For more information, see:

- Promoting shadow models with MLOps
- Rejecting shadow models with MLOps

Promoting shadow models with MLOps

A shadow model runs alongside an active model. Both models receive production data and generate outcomes, but the outcomes of the shadow model are not used to make business decisions. Check how the shadow model performs over time, and if the model proves better suited to your business needs, promote it as the active model.

Promoting shadow models is part of the Machine Learning Operations (MLOps) of Pega Platform.

Pega Customer Decision Hub
Before you begin: Review the performance of the shadow model. For more information, see Monitoring predictions.

1. In the navigation pane of Prediction Studio, click **Predictions**.
2. From the list of predictions, open a prediction that contains a shadow model, and then click the **Models** tab.
3. Display the shadow model by expanding the twist arrow to the left of a current model.
4. Ensure that the status of the model is SHADOW.
5. To the right of the shadow model, click the **More** icon, and then click **Promote model**.
6. In the **Promote *model*** dialog box, click **Promote model** to confirm.

Result: The model is promoted for deployment to production as the active model. For more information, see Understanding MLOps.

Rejecting shadow models with MLOps

A shadow model runs in your production environment alongside an active model. The shadow model receives production data and generates outcomes, but does not impact your business decisions. The system tracks the outcomes to help you evaluate how the shadow model performs in production. If the model is not suitable for your needs, you can reject the model, and then replace it with a different one.

Rejecting shadow models is part of the Machine Learning Operations (MLOps) of Pega Platform.

Pega Customer Decision Hub

Before you begin: Review the performance of the shadow model. For more information, see Monitoring predictions.

1. In the navigation pane of Prediction Studio, click **Predictions**.
2. From the list of predictions, open a prediction that contains a shadow model, and then click the **Models** tab.
3. Display the shadow model by expanding the twist arrow to the left of the current model.
4. Ensure that the status of the model is SHADOW.
5. To the right of the shadow model, click the **More** icon, and then click **Reject model**.
6. In the **Reject *model*** dialog box, click **Reject model** to confirm.

Result: The shadow model is removed. You can replace the currently active model with a new candidate model. For more information, see Replacing models in predictions with MLOps.

Updating active models in predictions through API with MLOps

You can create and deploy models directly from your modeling tool to Pega Platform, by using a scripting language of your choice and the V2 Prediction API endpoints. The API provides you with options to remotely perform Machine Learning Operations (MLOps): add a model to your application, review (approve or reject) a model update, and retrieve the status of a model update.

Use this API to integrate the model approval process in Pega Platform with any external model deployment process that your organization uses.

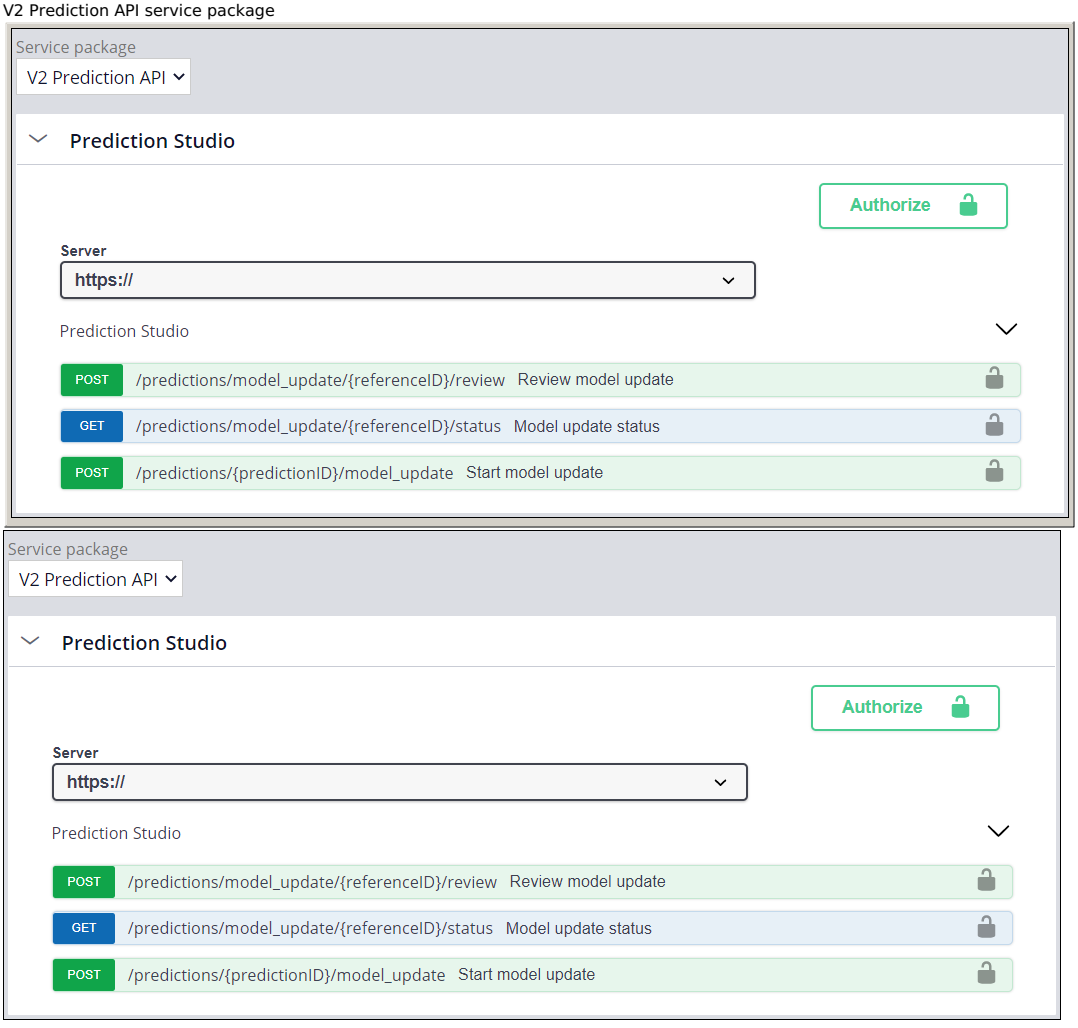
To learn about the work flows, components, and architecture behind model updates, see Understanding MLOps.

Pega Customer Decision Hub

V2 Prediction API service package

The V2 Prediction API service package, which you can access from Dev Studio, describes the model update endpoints, requests, and responses. Use the service package to test the endpoints by generating a sample response to a simulated endpoint. For more information, see Accessing and testing V2 Prediction API.

V2 Prediction API service package



Model update prerequisites

To replace models in predictions through the V2 Prediction API, you need to configure your Business Change pipeline, Prediction Studio settings, and data scientist operators to meet the requirements for the model update feature. For more information, see MLOps prerequisites.

Model update endpoints

Learn more about the model update endpoints by reviewing sample responses and parameter descriptions:

- Starting a model update
- Reviewing a model update
- Retrieving the model update status

Starting a model update

Use this endpoint to start an update process to replace a model in a prediction by using Machine Learning Opertions (MLOps).

Pega Customer Decision Hub

POST /predictions/{predictionID}/model_update

Method
POST
Endpoint
/predictions/*{predictionID}*/model_update
Sample URL
http://*IP:PORT*/prweb/api/application/v2/predictions/CDHSAMPLE-DATA-CUSTOMER!CUSTOMERCHURNPREDICTION/model_update

{predictionID} is an identifier of a prediction in which you want to update a model. You can view this prediction ID in Prediction Studio:

1. In Prediction Studio, click **Predictions**.
2. Open a prediction, and then click **Actions Show more info** .

The **Prediction ID** field contains the prediction ID that you need to add to your URL, for example, CDHSAMPLE-DATA-CUSTOMER!CUSTOMERCHURNPREDICTION.

Request

For example: {"model_source": "GBM_Churn.zip","validation_dataset": "churn-all.csv","outcome_field_name": "Outcome","model_label": "Churn GBM","supporting_documents": [{ "description": "Behavior report", "name": "BehaviorReport.pdf" }, { "description": "Population report", "name": "PopulationReport.pdf" },],"override_mappings": [{ "predictor": "Credit_History", "property": "pyCreditHistory" }]}

Model update request parameters

Name	Type	Description
model_source	String	For file-based models, the path to the model that you want to add. For cloud-based models, the path to the metadata file of the model that you want to add. Ensure that the model is uploaded to a folder in the repository that is configured as the Prediction Studio analytics repository. For more information, see: • Enabling Machine Learning Operations • Metadata file specification for predictive models
validation_dataset	String	The path to the validation data set that you want to use to compare the active and candidate models before you approve or reject a model update.
perform_validation	Boolean (True, False)	This parameter allows you to validate the model.
outcome_field_name	String	The name of the column that contains the outcome in the validation data set CSV file.
target_component_name	String	For supporting models, the name of the component that holds the model in a prediction strategy. For outcome-based models, the outcome name.
model_label	String	The name of the candidate model.
supporting_documents	Array	An array of documentation files for the model.
override_mappings	Array	An array of mappings of predictors in the models to the corresponding Pega Platform properties.

Supporting documents parameters

Name	Type	Description
name	String	The path to a documentation file for the model.
description	String	The description of the documentation file.

Override mappings parameters

Name	Type	Description
predictor	String	A field in the candidate model that you want to select as predictor data and map to a corresponding Pega Platform property.
property	String	A Pega Platform property that corresponds to the field in the candidate model.

Response

For example: { "referenceID": "M-9012"}

Response parameters

Name	Type	Description
referenceID	String	Contains a reference ID that uniquely identifies a model update. Other endpoints use reference IDs to process requests about specific model updates, for example, to review (approve or reject) the model update or to retrieve the status of the model update.

Reviewing a model update

Use this endpoint to approve or reject a candidate model in a prediction by using Machine Learning Operations (MLOps).

Pega Customer Decision Hub

POST /predictions/model_update/{referenceID}/review

Method
POST
Endpoint
/predictions/model_update/*{referenceID}*/review
Sample URL
http://*IP:PORT*/prweb/api/application/v2/predictions/model_update/M-5104/review

{referenceID} is an identifier of a model update that you want to approve or reject, for example, M-5104. A request that starts a model update returns this reference ID. For more

information, see Starting a model update.

Request

For example: { "action": "Approve", "review_note": "Approving model for shadow", "shadow_mode": true}

Review request parameters

Name	Type	Description
action	String (Approve, Reject)	This parameter allows you to approve a model for deployment to production or to reject the model.
review_note	String	A comment from the data scientist to indicate the reason for the approval or rejection of the model.
shadow_mode	Boolean (True, False)	True places the model in shadow mode. False replaces the active model in the prediction with this model.

Response

For example: { "message": "referenceID M-5104 is Approved. New status Approved"}

Response parameters

Name	Type	Description
message	String	The output of the review command.

Retrieving the model update status

Use this endpoint to load the status of a model update that was started by using Machine Learning Operations (MLOps).

Pega Customer Decision Hub

GET /predictions/model_update/{referenceID}/status

Method
GET
Endpoint
/predictions/model_update/*{referenceID}*/status
Sample URL
http://*IP:PORT*/prweb/api/application/v2/predictions/model_update/M-5104/status

{referenceID} is an identifier of a model update that you want to approve or reject, for example, M-5104. A request that starts a model update returns this reference ID. For more information, see Starting a model update.

Response

For example:

The following example shows a response when the model configuration is in progress:

{ "ModelUpdateStatus":"Configuration in progress", "LastMessage":"Model update case M-6028 created. Preparing for artifacts creation.", "UpdateTimeStamp":"December 30, 2020 01:44 EST", "BranchName":"M-5104", "BranchAvailable":false}**For example:**

The following example shows a response when the model configuration failed:

{ "ModelUpdateStatus":"Configuration failed", "LastMessage":"Unable to find the model to update.Specify the component name of the model to be updated in the request.", "UpdateTimeStamp":"January 13, 2021 07:49 EST", "BranchName":"M-5104", "BranchAvailable":false}**For example:**

The following example shows a response when the model is ready for review:

{ "ModelUpdateStatus":"Ready for review", "LastMessage":"Awaiting approval", "UpdateTimeStamp":"December 29, 2020 02:18 EST", "BranchName":"M-5104", "BranchAvailable":true}

Response parameters

Name	Type	Description
ModelUpdateStatus	String	The status of the model update.
LastMessage	String	The last status in the model update process before the current status was reached.
UpdateTimeStamp	String	The date and time of the last status change.
BranchName	String	The branch in Pega Platformthat contains the status update.
BranchAvailable	Boolean (True, False)	True indicates that a branch exists. False indicates that a branch has not been created yet or was deleted.

Accessing and testing V2 Prediction API

Explore the possibilities of the V2 Prediction API by testing the endpoints. Each test generates a sample response to a simulated endpoint call, which helps you understand how the endpoint works and what types of responses to expect.

The *PredictionStudio* service package in Dev Studio holds the settings for the V2 Prediction API. By default, the service package uses the OAuth 2.0 authentication type. OAuth 2.0 is the preferred option that provides more security than basic authentication. For more information about configuring service package settings, see Defining processing and authentication for service packages.

Pega Customer Decision Hub
Before you begin: Create an OAuth 2.0 client registration data instance and save the credentials. For more information, see Creating and configuring an OAuth 2.0 client registration.

1. In the header of Dev Studio, click the **Resources menu** icon, and then click **Pega API**.
2. In the **Pega API Help** window, from the **Service package** list, select **V2 Prediction API**.
3. Click **Authorize**.
4. In the **Available authorizations** window, enter the required credentials for the scopes that you want to use:
    a. In the **username** and **password** fields, enter your operator credentials.
    The V2 Prediction API is server-specific and not application-specific. You can work with all the predictions on the server to which you have access. Your operator's default access group determines which predictions you can access.
    b. In the **type** field, select the authorization type:

        - **Request body**
        - **Query parameters**
    c. In the **client_id** and **client_secret** fields, enter the OAuth credentials that you saved in Creating and configuring an OAuth 2.0 client registration.
5. Click **Authorize**, and then click **Close**. **Result:** You can now test the endpoints.
6. Click the endpoint that you want to test. **Result:** The details of the endpoint are displayed, including the description, parameters, and sample responses.
7. Test an endpoint:
    a. Click the name of the endpoint that you want to test to expand the section.
    b. Click **Try it out**.
    c. If the endpoint requires additional parameters, enter the information in the required fields.

    For example: If you want to test the *GET /predictions/model_update/{referenceID}/status* endpoint, which retrieves the status of a specific model update, after you click **Try it out**, provide the reference ID of the model update in which you are interested. For more information, see Updating active models in predictions through API with MLOps.
    a. Click **Execute**.

    Result: The system generates a response to the simulated query.

Understanding MLOps

The workflow for updating models by using Machine Learning Operations (MLOps) is a standardized process that data scientists can use to replace active models in predictions with other models, scorecards, or fields that represent scores, and to deploy the new models to the production environment.

Pega Customer Decision Hub

Model update workflow

A model update process consists of the following stages:

1. The process starts when you update a prediction in your non-production environment by replacing an active model with a predictive model, scorecard, or field in the data model that contains a score. You can select a model from Prediction Studio, upload a PMML and H2O model to Prediction Studio, or connect to an external model that you developed on Amazon SageMaker or Google AI Platform.

   When creating a machine learning model in a third-party environment, you can use historical data from Pega Platform and other sources to train the model. For more information, see Replacing models in predictions with MLOps or Updating active models in predictions through API with MLOps.

2. You can validate the candidate model against a data set, and compare the new model with the current model. This analysis provides relevant metrics to help you decide which model has better performance with a static data set.
3. After you evaluate the models, you can approve or reject the candidate model for deployment to production. You can place an approved model in shadow mode (supported for predictive models) or replace the current model with the new model. For more information, see Evaluating candidate models with MLOps or Updating active models in predictions through API with MLOps.
4. If you approve a model, the decision strategy that uses the updated prediction is changed, and the strategy as well as the candidate model are added to a new branch.
5. Depending on how your environment is configured, deployment involves different stages and components:
    - In a Pega Customer Decision Hub environment, the system creates and resolves a change request in Pega 1:1 Operations Manager. A team lead can verify the changes in the rules and the relevant documentation. The change request is packaged into a revision, and a deployment manager can promote the prediction with the candidate model to production. For more information, see Understanding the change request flow.
    - In other environments, in which predictions are used for other purposes, such as customer service and intelligent automation, components such as Pega 1:1 Operations Manager and Revision Manager are not present. A system architect merges the branch with the model update to the application ruleset. A deployment manager can then deploy the application ruleset to production.
6. If you deploy the candidate model to production in shadow mode, it runs alongside the original model, receives production data, and generates outcomes, but the outcomes do not impact business decisions.

    - If the candidate model performs well in production, you can promote it to the active model position. For more information, see Promoting shadow models with MLOps or Updating active models in predictions through API with MLOps.

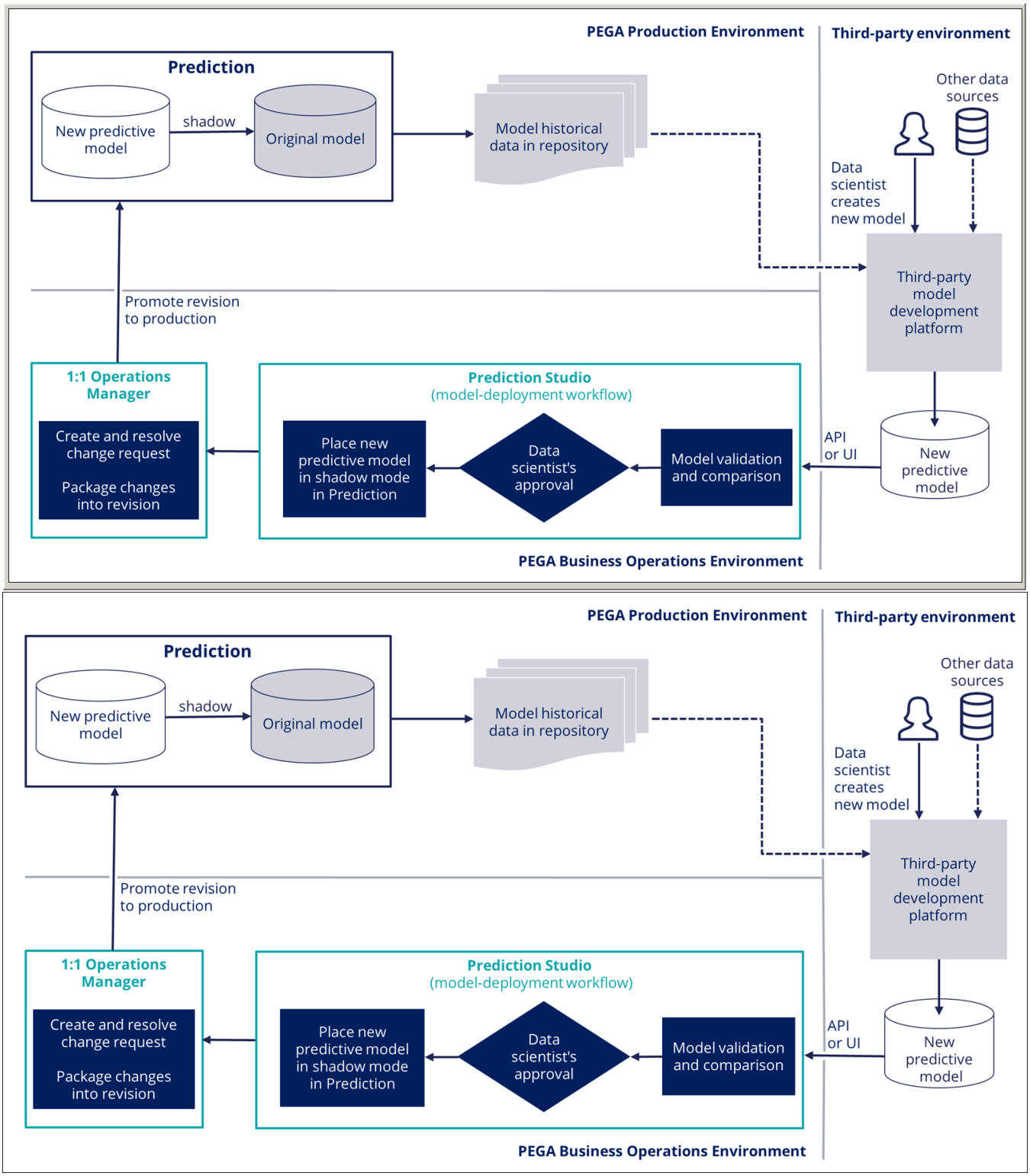
      The model is deployed through the business change pipeline by using Pega 1:1 Operations Manager and Revision Manager. For more information, see the *Pega 1:1 Operations Manager User Guide* on the Pega Customer Decision Hub product page.

    - If the model proves ineffective, you can reject it, and add another model to the prediction. For more information, see Rejecting shadow models with MLOps or Updating active models in predictions through API with MLOps.

Note: The system sends emails and notifications to the data scientists work group at key stages of the model update process.

The following diagram shows an example of the model update process. In this example, a data scientist approves a predictive model for deployment to production as a shadow of the original model.

Model update workflow



PEGA Production Environment
Third-party environment
Prediction
New predictive model
shadow
Original model
Model historical data in repository
Other data sources
Data scientist creates new model
Third-party model development platform
Promote revision to production
1:1 Operations Manager
Create and resolve change request
Package changes into revision
Prediction Studio
(model-deployment workflow)
Place new predictive model in shadow mode in Prediction
Data scientist's approval
Model validation and comparison
API or UI
New predictive model
PEGA Business Operations Environment

- Updating active models in predictions with MLOps
- Updating active models in predictions through API with MLOps

Active and candidate model type combinations

Use the Machine Learning Operations (MLOps) feature to add a candidate model to a prediction, either as a shadow of an active model or as the new active model that replaces the previous active model. When planning a model update, review the supported combinations of active and candidate model types to see what changes you can make.

Pega Customer Decision Hub

You can replace the following active models in predictions:

- Adaptive models
- Predictive models
- Scorecards
- Fields

You can replace the active models with the following entities:

- Predictive models
- Scorecards

- Fields

The MLOps feature allows you to update outcome-based and supporting models in single-stage and two-stage predictions:

Outcome-based models
: Models that are used in non-customized predictions.

Supporting models
: Supporting models are those that are included in strategies and substrategies in customized predictions. You can use adaptive and predictive models, as well as scorecards, as supporting models.

You can update a model by using one of the following methods:

Shadow mode
: Use this option to deploy a candidate model to the production environment as a shadow of the active model. The system tracks the outcomes of the shadow model, but does not use the outcomes in business decisions. You can monitor how the shadow performs against production data, and use this insight to decide whether to promote the shadow as the new active model.

  Shadow mode is only available for predictive models. If an active model has a shadow, the **Replace model** option is disabled for that model. You need to either promote the shadow as the new active model or reject the shadow before you can use the **Replace model** option again on that model.

Replace
: A replace operation allows you to deploy a candidate model to production as a replacement of the currently active model. A replace operation triggers the model update validation and approval process. This option is only supported for predictive models.

Direct replace
: A direct replace operation does not involve an approval process. The change is saved to a branch that a system architect can merge to the application ruleset. A deployment manager can then deploy the application ruleset to production.

The following tables show which options are supported for each model type that you might want to replace. For example, if the active model type is adaptive, you can replace it with a predictive model, and deploy the candidate model in shadow mode or as a replacement for the adaptive model. A scorecard or field cannot replace an adaptive model.

When the active model type is ADAPTIVE

Candidate model	Active model type	Shadow mode	Replace	Direct replace
Predictive	Outcome-based or supporting	✓	✓	Not available
Scorecard	Outcome-based or supporting	Not available	Not available	Not available
Field	Outcome-based	Not available	Not available	Not available

When the active model type is PREDICTIVE

Candidate model	Active model type	Shadow mode	Replace	Direct replace
Predictive	Outcome-based or supporting	✓	✓	Not available
Scorecard	Outcome-based or supporting	Not available	Not available	Not available
Field	Outcome-based	Not available	Not available	Not available

When the active model type is SCORECARD

Candidate model	Active model type	Shadow mode	Replace	Direct replace
Predictive	Outcome-based or supporting	✓	✓	Not available
Scorecard	Outcome-based or supporting	Not available	Not available	✓
Field	Outcome-based	Not available	Not available	✓

When the active model type is FIELD

Candidate model	Active model type	Model update: Shadow mode	Model update: Replace	Model update: Direct replace
Predictive	Outcome-based	✓	✓	Not available
Scorecard	Outcome-based	Not available	Not available	✓
Field	Outcome-based	Not available	Not available	✓

- Updating active models in predictions with MLOps
- Understanding MLOps
- Promoting shadow models with MLOps
- Rejecting shadow models with MLOps

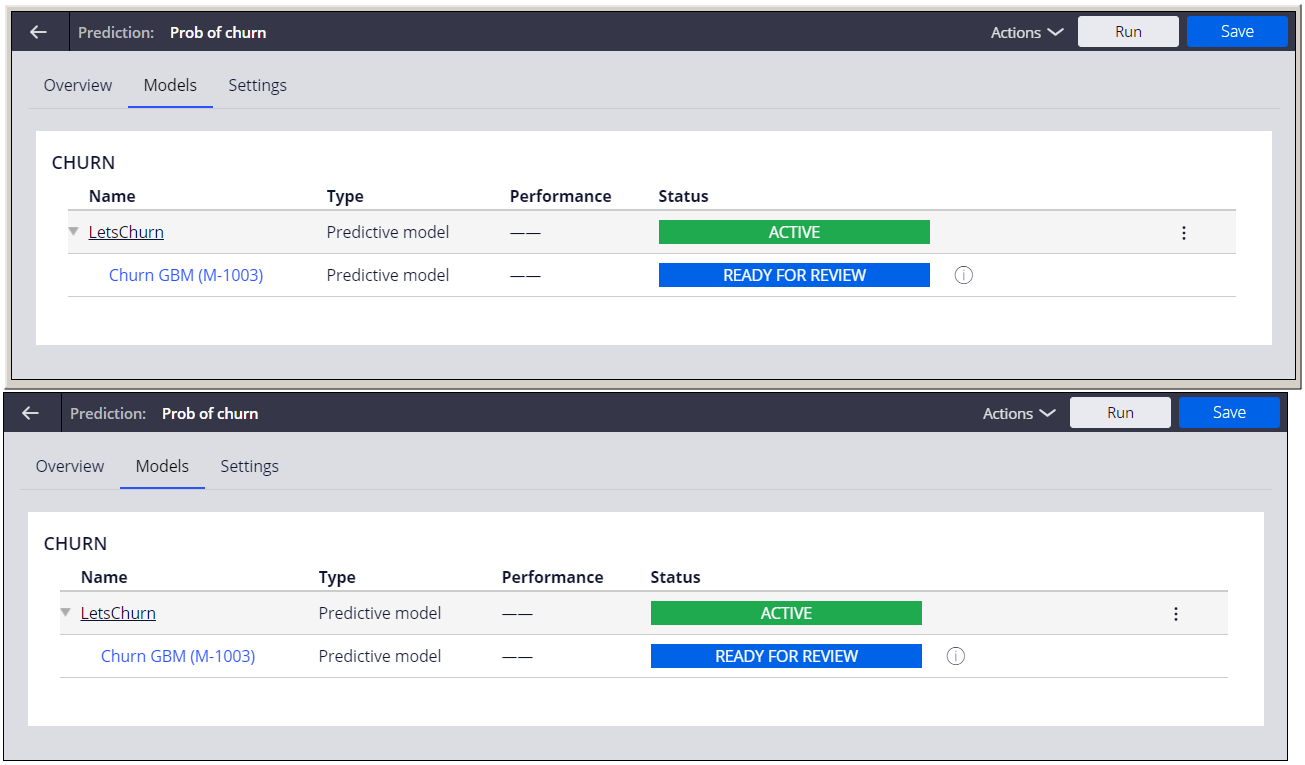
Model update statuses and notifications

As a candidate model passes through the various stages of the model update process, such as validation, review, or shadow mode, its status changes in Prediction Studio. Data scientists and other stakeholders then receive email notifications about the key events in the life cycle of a model update.

Operations related to updating models are part of the Machine Learning Operations (MLOps) feature of Pega Platform.

Learn about these statuses and notifications to better understand the model update process.

An active model and a candidate model in a prediction



Pega Customer Decision Hub

Candidate model statuses in Prediction Studio

CONFIGURATION IN PROGRESS
: The creation of artifacts is in progress. This status appears after you trigger a model update request.

CONFIGURATION FAILED
: The creation of artifacts failed. The mapping of predictors was not successful. Open the model by clicking its name, and read the error message in the **Model comparison** window. Resolve the error (for example, by fixing the mapping of predictors) and resume the analysis by saving your changes. If necessary, you can reject the model.

VALIDATION IN PROGRESS
: The creation of artifacts is complete. The comparison of the current model with the candidate model is in progress.

VALIDATION FAILED
: The creation of artifacts failed. This status appears when an error occurs during the creation of the required artifacts, or during analysis. Open the model by clicking its name, and read the error message in the **Model comparison** window. Resolve the error and resume the analysis by saving your changes or reject the model.

READY FOR REVIEW
: Validation is complete and a data scientist can review the model update.

ACCEPTED
: A data scientist approved the model update with a comment on the approval. The candidate model was added to the prediction in shadow mode or as an active model.

REJECTED
: A data scientist rejected the candidate model with a comment on the rejection. For example, by reviewing the model comparison charts, the data scientist can evaluate the performance of the candidate model as insufficient.

SHADOW
: The new model shadows the old model, that is, the new model receives production data and generates outcomes. The system tracks the outcomes of the shadow model, but does not use them to make business decisions. You can monitor the performance of the shadow model, and decide whether to promote it as the new active model or reject it.

Email notifications

Prediction Studio sends email notifications to data scientists and other stakeholders to inform them about the following key events in a model update:

- A model update is ready for review and pending approval.
- A model update failed during analysis (configuration and validation stages). The email contains the reason for the failure. For example, a model update can fail because the predictors are not mapped correctly or because a shadow model is already deployed for the active model.

In environments with Pega Customer Decision Hub, the system creates a *Change prediction* request in Pega 1:1 Operations Manager to process the model update that a data scientist triggers from Prediction Studio. Data scientists and other stakeholders receive email notifications to inform them about the status of the change request in the following situations:

- The change request is successfully resolved. The email contains the expected deployment date on production.
- A revision manager locked the revision that is associated with the change request. Deployment of the model is on hold.
- A revision manager withdrew the revision that is associated with the change request. As a result, the change request is also withdrawn. The branch (for example, M-7012) that contains the model update is deleted in Prediction Studio.

- Updating active models in predictions with MLOps
- Promoting shadow models with MLOps
- Rejecting shadow models with MLOps

Active and candidate model comparison charts

Before you deploy a candidate model in shadow mode or as a replacement of the active model, evaluate the efficiency of the candidate model by adding a validation data set and generating charts that compare the two models in terms of score distribution, ROC curve, gains, and lift.

Operations related to updating and evaluating models are part of the Machine Learning Operations (MLOps) feature of Pega Platform.

Pega Customer Decision Hub

Sample model comparison charts



Learn more about the model comparison charts, and how to use them:

Score distribution
: The score distribution chart displays a non-aggregated classification of the results of a model. The scores from the model are classified into intervals with increasing likelihood of positive behavior. With model analysis based on score distribution, you can compare models on consistent terms: how they distribute cases over the score bands, as well as their behavior in each band. The more predictive power a model has, the more distinct the bands are in terms of their performance. To compare model performance, Pega Platform distributes model results into 16 predefined intervals and compares the corresponding score bands of the two models.

ROC curve
: The receiver operating characteristic (ROC) curve represents the predictive ability of a model by plotting the true positive rate (sensitivity) against the false positive rate (1 - specificity or inverted specificity). The true positive rate shows how often the model predicts a positive outcome when the actual outcome is positive. The false positive rate shows how often the model predicts a positive outcome when the actual outcome is negative. The closer the area under the curve (AUC) is to the top-left corner of the chart, the more accurately the model distinguishes positives from negatives.

Gains
: A gains chart is a tool to compare the cumulative gains that each model provides. A gains curve represents the cumulative percentage of positive responses (gains) that you get by targeting increasing percentages of the population based on the predictions of a model, compared to the gains obtained by targeting the population at random. The closer the curve is to the top-left corner of the chart, the better the model is at predicting positive responses. A model with a better gains curve allows you to obtain the highest number of positive responses when reaching the smallest percentage of the population. A gains chart also shows the tipping point at which you have reached the most positive responses. Beyond this point the cost of contacting the remaining portion of the population might outweigh the gains.

Lift
: Lift is a measure of how much more likely you are to get positive responses by using a predictive model than by targeting a random sample of the population. A lift chart graphically represents the lift that you get from each model as you target increasing percentages of the population. A typical lift curve runs down diagonally from the top-left corner of the chart to the value of 1, which represents no improvement in efficiency compared to targeting the population at random. A model with a lift curve that extends closer to the top-left corner has more predictive power and allows you to get a greater number of positive responses by targeting a smaller portion of the population. For example, a lift of 3 for the top 10% of the population (identified by the model as the group with the highest response ratio) means that you can get three times as many positive responses by targeting that group as when you do not use any model.

- Updating active models in predictions with MLOps

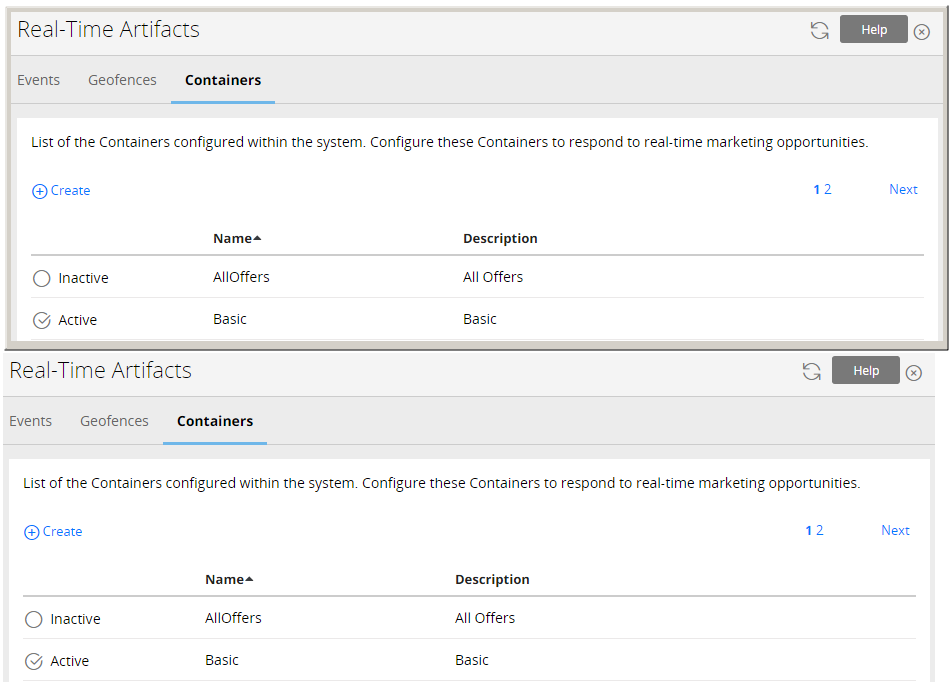
Managing content in real-time channels with real-time containers

Real-Time Containers provide an easy low-code or no-code way for marketers to manage content that appears in real-time channels, such as web and mobile. As an example, a marketer may define a Container to represent a region of a web page. They can then associate the Container with a Campaign or a Next-Best-Action in the Next-Best-Action Designer. In real-time when the Container is called (say on a web page), the associated Campaign or a Next-Best-Action is executed and the Container gets populated with results.

Pega Customer Decision Hub

Pega Customer Decision Hub provides a set of APIs, which can be invoked from real-time channels, to facilitate the population of a Container and to capture customer responses.

Real-Time Containers are listed on the **Containers** tab of the Real-Time Artifacts landing page. This landing page is accessible in the Pega Customer Decision Hub portal via: **Content Real-Time Artifacts**



The Containers landing page provides a quick overview of the Containers configured in the application.

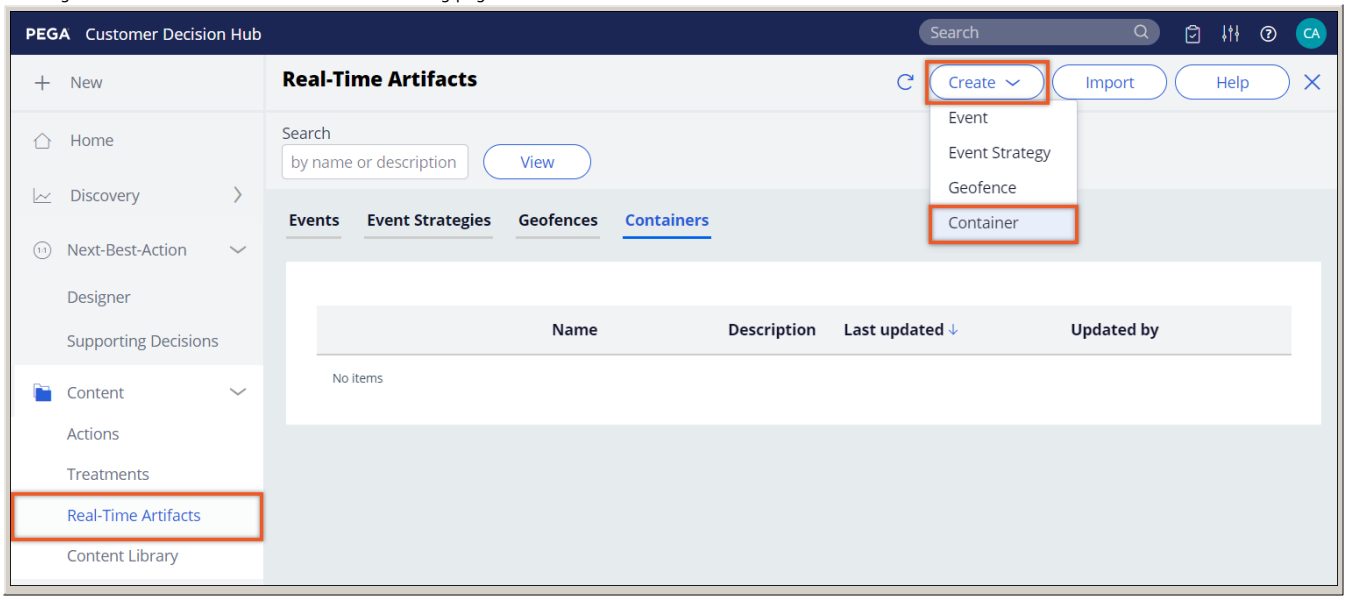
- Creating a real-time container
- Using real-time container APIs
- Configuring a container invocation script
- Invoking a fallback strategy
- Identifying customers who interact with a real-time container

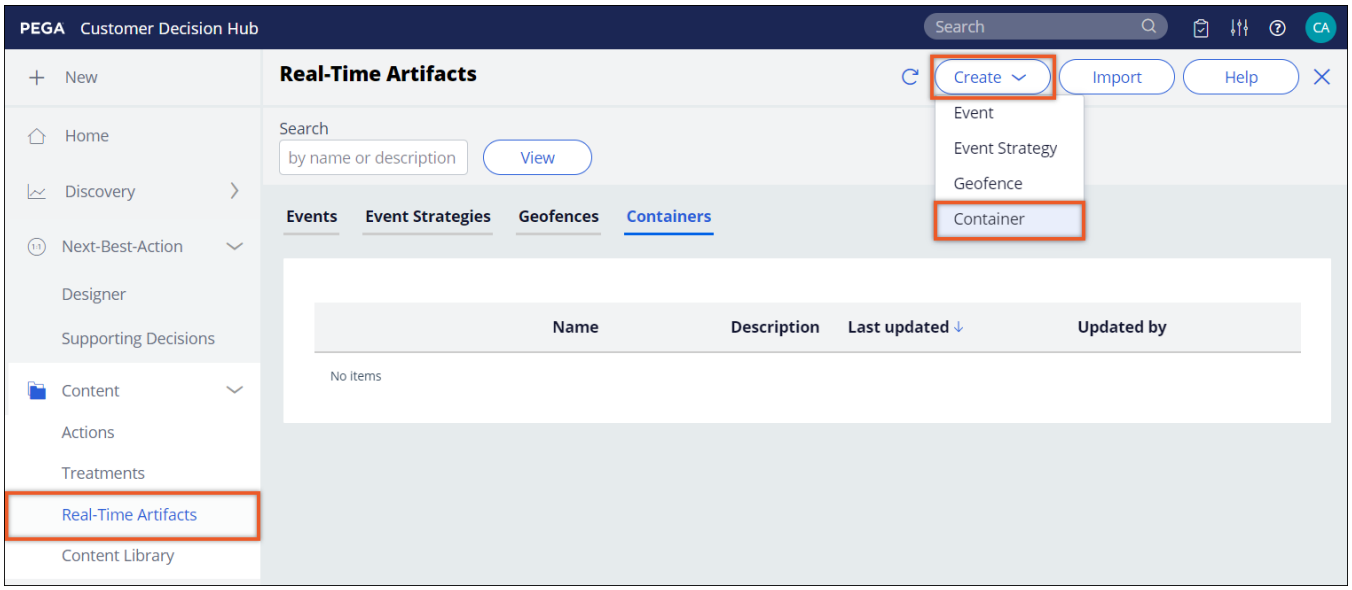
Creating a real-time container

To create a new Container, go to **Content Real-Time Artifacts** and in the **Create** menu in the Containers landing page select **Container**. Enter a unique name for the Container in the **short description** field and click **Create and open** to launch the Container rule form.

Pega Customer Decision Hub

Creating a real-time container in the Containers landing page





The Container rule form has the following tabs:

Tab Name	Purpose
Container Details	Specify the availability details for the container; view Campaigns and Next-Best-Action configuration associated with this container. Is this container currently active? **No, this Container will not respond to requests at this time** – select this option if this container is not yet supposed to respond to external channels with actions. **Yes, this Container will respond to requests** – select this option if this container is supposed to respond to external channels with actions. Impression capture **Captured on retrieval** – select this option if the impression records are to be captured in the Interaction History table. Take into consideration the volume and traffic before selecting this option, as Interaction History table can grow very fast, which can affect the performance of the application. **Captured by Channel** – select this option if the external system is capturing the impression records. Click through behavior **Capture click through and initiate offer flow** – select this option if the container is supposed to initiate an offer flow as well. This option is normally not selected for inbound implementations. **Capture click through only** – select this option, if the container is supposed to capture click trough statistics only.
History	Specify the Description and Usage information for this Container.

Using real-time container APIs

This section provides information on the APIs available for use with Real-Time Containers.

Pega Customer Decision Hub

When invoked, each API sets one of the following HTTP status codes in its response:

- 200 - Successful invocation
- 400 - Bad request
- 500 - Internal error

- Triggering a real-time container with the Container REST service
- Making custom Strategy Result properties available for real-time container data flows
- Capturing responses to actions triggered by a real-time container
- Accessing a customer's interaction history with the CustomerOfferHistory REST service
- Capturing clicks on the click-through URL

Triggering a real-time container with the Container REST service

The Container service is available to make a Next-Best-Action decision in inbound, real-time channels such as web self-service, call center and mobile applications. The Container REST API supports both the POST and GET HTTP methods. The Container service provides a layer of abstraction between the calling channel and the specific decision logic that is executed so a user can easily change the content delivered to the channel without changing the interface or parameters in the API.

Pega Customer Decision Hub
Note: This service supports both Pega and third-party identity matching for the Web and Mobile channels.

The URL pattern for this service is:

http://<host>:<port>/prweb/PRRestService/PegaMKTContainer/V3/Container

Swagger document

To help you configure the service, you can refer to the following Swagger document: Swagger document for Container and Capture Response v3.

POST Service Request

For the POST request, the service expects a JSON Object with the following attributes:

Name	Required	Description
SubjectID	Optional for the Web and Mobile channels	The identifier for the specified context.
ContextName	✓	The singular friendly name of the context entity that indicates the type for SubjectID.

Name	Required	Description
EnvironmentID	Optional	When using third-party identity management, this represents the identifier for the user.
ContainerName	✓	Name of the Container or Containers to trigger. Can be a single value or a comma-separated list of Containers, for example, NextBestAction, BestOffer.
Channel	✓	Channel calling the container, where the results will be rendered.
Direction	✓	Direction of communication (Inbound or Outbound)
Placements	Optional	Comma-delimited list of placement types (for example, Tile, Hero) that will limit the number of treatments returned to the top treatments that match the specified placement types.
Contexts	Optional	Situation data from channel such as customer mood, current web page.

Contexts attributes

Name	Description
Key	Identifier for context info.
Type	Type of context data.
Value	Value for the specified key.

A sample request JSON Object is shown below:

{"SubjectID":"AID-100","ContextName":"Account","ContainerName": "NextBestAction","Channel": "Web","Direction": "Inbound","Placements":"Hero,Tile,Tile,Tile","Contexts":[{"Type":"","Value":"","Key":""}]}

GET Service Request

For the GET request, the service expects a JSON Object with the following attributes:

Name	Required	Description
SubjectID	Optional for the Web and Mobile channels	The identifier for the specified context.
ContextName	✓	The singular friendly name of the context entity that indicates the type for SubjectID.
ContainerName	✓	Name of the Container to trigger.
Channel	✓	Channel calling the container, where the results will be rendered
Direction	✓	Direction of communication (Inbound or Outbound).
Placements	Optional	Comma-delimited list of placement types (for example, Tile, Hero) that will limit the number of treatments returned to the top treatments that match the specified placement types.
ContextKey	Optional	Situation data from channel such as customer mood, current web page.
ContextValue	Optional	Situational data value.

A sample GET URL is shown below:

http://<HOST>:<PORT>/prweb/PRRestService/PegaMKTContainer/V3/Container?SubjectID=AID-100&ContextName=Account&ContainerName=NextBestAction&Channel=Web&Direction=Inbound&Placements=Hero,Tile,Tile,Tile

Service Response

For the response, the service returns a JSON Object with the following attributes:

Name	Description
Status	OK for successful result.
ContainerName	Name of the container.
RankedResults	An array containing one or more actions returned by Strategy execution.

RankedResults attributes for each action:

Name	Description
Name	Name of the action.
Issue	Issue to which the action belongs.
InteractionID	Decision results identifier for the Container call. Decision results are stored in a Cassandra datastore (*pxDecisionResults*) as part of the Adaptive Analytics delayed learning feature. The InteractionID is required during the CaptureResponse process to match the response to the original decision result, so that adaptive models can learn from the interaction.
Rank	Action rank based on Strategy configuration.
SubjectID	The identifier for the specified context.
ContextName	The singular friendly name of the context entity that indicates the type for SubjectID.
SubjectName	Business-friendly name for the SubjectID (value set in TransformSROutput data transform).
Category	Category to which the action belongs.
CustomerCost	Cost of this product/service to the customer.
ContentFormat	Type of content in ImageURL (HTML fragment/Image).
Label	Business friendly description of the action.
Benefits	Benefits to the customer.
WhyRelevant	Why this action is relevant for the customer.
Direction	Direction in which the interaction was initiated, Inbound or Outbound.
ShortDescription	Short description of the action.
GroupID	Identifier for the SubjectID's parent entity.

Name	Description
PaidAudienceName	Name of the paid audience.
Identifier	Unique identifier for the action.
Placement	Type of treatment.
Pricing	Description of the pricing.
Journey	Name of the associated customer journey.
JourneyStage	Name of the associated customer journey stage.
JourneyStep	Name of the associated customer journey step.
EligibilityDescription	Criteria for customers to be eligible for this action.
Propensity	Action propensity value.
AgentCompensation	Agent compensation amount if customer accepts action.
ClassIdentifier	Internal class for the action results.
Priority	Action priority based on Strategy configuration.
Channel	Channel the interaction occurred on.
BundleName	Name of the action bundle.
ImageURL	URL to the associated action image.
BundleParent	Indicates if this action is a bundle parent.
Variant	Variation of the action (for example, 64gb, 128gb, 256gb).
Treatment	The name of the action treatment.
InternalCost	The internal company cost of the action.
CampaignID	Work ID of the active Campaign associated with this Container.
OfferValue	The value the customer receives from this action.
ClickThroughURL	URL to invoke the CaptureWebClickThrough API. The encrypted parameter in this URL contains the actual click-through URL specified as part of the action's details. Typically, this leads to a link or file that provides more details about the action.
DecisionTime	The time of the decision.

A sample response from the invocation of this service is shown below:

{"Status": "OK","ContainerList": [{"Status": "OK","ContainerName": "UPlusBankWSS","RankedResults": [{"Group": "CreditCards","Issue": "Sales","InteractionID": "-5826878382873476460","Category": "","SubjectName": "14","CustomerCost": 0.0,"ContentFormat": "Image","Label": "Signup for Platinum Rewards Card", "Benefits": "The Platinum Card …", "WhyRelevant": "U+ Financial Personal Card...", "Direction": "Inbound","Name": "PlatinumRewardsCard","ShortDescription": "Great benefits with access to a higher credit line","GroupID": "","PaidAudienceName": "","Identifier": "/Sales/CreditCards/PlatinumRewardsCard","Placement": "Tile","Pricing": "INTEREST ...", "JourneyStage": "","JourneyStep": "","EligibilityDescription": "Offer only available to U.S. residents 18 and older.","Propensity": 0.008158007388236195,"AgentCompensation": 4,"ClassIdentifier": "Offer","Priority": 0.05588235060941794,"Channel": "Web","BundleName": "","ImageURL": "http://host:port/image.jpg","Rank": 7,"BundleParent": "false","Variant": "","Treatment": "Plat__Card_Tile_02","InternalCost": 0,"CampaignID": "NBA","OfferValue": 0,"ClickThroughURL": "https://PM85-autotest2.pegamkt.io/prweb/PRHTTPService/PegaMKTContainer/Services/CaptureClickThrough?Px=%7Bpr%7DyB%","SubjectID": "14","Journey": "","DecisionTime": "20200805T164224.957 GMT","ContextName": "Customer"}]}]}

- Identity management for the Container REST service

Identity management for the Container REST service

The Web and Mobile channels support using identity matching to uniquely identify and track anonymous users when making Next-Best-Action decisions. The Container service supports using browser-based identity matching (such as Pega's identity matching, or a third-party identity solution), as well as server-side OAuth authentication.

Pega Customer Decision Hub

Browser-based identity matching

Browser-based identity matching is used when the Container service is deployed on a company web site as an unauthenticated service that runs on a customer's browser. In this case, the service uses browser cookies and the customer ID to identify an individual and present personalized content in the real-time container. The matching has the following stages:

1. Check if identity matching is enabled using the *MKTLoadCustomerFromIdentityMatch* Dynamic system setting.
2. Check if the *SubjectID* is provided.
3. Check if the *ExternalID* is provided.
4. Look up *SubjectID* from *ExternalID* or *MKTID*.

Pega Customer Decision Hub provides native identity management functionality. You can also use a third-party solution to track anonymous users and link them to identified profiles based on their past actions, such as logging in, clicking on an email link, or completing a form.

Pega identity matching

In order to use the Pega identity matching, set the *MKTLoadCustomerFromIdentityMatch* Dynamic System Setting to **true**. Once enabled, Pega Customer Decision Hub will automatically generate a unique ID and place a MKTID cookie in the users' browser.

The MKTID cookie will contain the unique identifier and all interactions will be captured using this ID. If the Container service passes the *SubjectID* (representing identified customer), the Container service will automatically create an association between the *MKTID* and the *SubjectID*. This allows Pega Customer Decision Hub to merge the anonymous and identified profiles and behaviors and make a Next-Best-Action decision based on the combined history.

Third-party matching

You can also leverage an existing third-party identity management solution with the Container service. The optional *ExternalID* parameter of the service is used for the third-party identifier. If only the *ExternalID* is provided then all interaction history will be captured using this identifier. If the *SubjectID* and the *ExternalID* are both provided, an association between these will be created and the *SubjectID* will be used to capture the interaction history.

OAuth authentication

For cases where the Container service is deployed on a server instead of running in a browser, browser-based identity matching is not used. Instead, Pega Customer Decision Hub supports server-to-server OAuth 2.0 authentication through the *PegaCDHSecuredContainer* service package.

In order to use OAuth 2.0 authentication, create an instance of OAuth 2.0 client registration, and then use the generated Client ID and Client Secret to allow the external application to access Pega Customer Decision Hub. For more information, see Creating and configuring an OAuth 2.0 client registration.

Making custom Strategy Result properties available for real-time container data flows

Now that you have passed in Context data as part of the Container request, you will want to access this data within your decision strategies. There are two utility functions that allow you to access this data depending on your requirements.

IsValueInContext

The IsValueInContext function checks if a value exists in the list of Contexts passed in the Container Request.

IsValueInContext function parameters:

Name	Description
Type	This parameter identifies the type of context.
Key	This parameter identifies the name of context.
Value	This parameter contains the data to be evaluated.

For example: @PegaMKTUtilities.IsValueInContext("Intent","IntentName","CloseAccount",Primary.ContainerPayload.Contexts)

GetContextValue

The GetContextValue function returns the value for the specified Type and Key in the Container Contexts.

GetContextValue function parameters:

Name	Description
Type	This parameter identifies the type of context.
Key	This parameter identifies the name of context.
Value	This parameter contains the data to be evaluated.

For example: @PegaMKTUtilities.GetContextValue("Intent","IntentName",Primary.ContainerPayload.Contexts)

Container settings

You can toggle the following additional settings for Context in Strategies:

- Identity matching - Enabled by default. To disable, toggle the MKTEnableIdentityMatching dynamic system setting.
- Customer lookup - Enabled by default. To disable, toggle the MKTEnableCustomerLookup dynamic system setting.
- Customer validation - Disabled by default. To enable, toggle the MKTEnableCustomerValidation dynamic system setting.

Capturing responses to actions triggered by a real-time container

CaptureResponse is a REST service that supports the POST HTTP method. This service allows callers to capture responses to actions triggered via Real-Time Containers.

Pega Customer Decision Hub

The URL pattern for this service is:

http://<host>:<port>/prweb/PRRestService/PegaMKTContainer/V3/CaptureResponse

This service can also be used to initiate the first action in its request. To utilize this feature and include action initiation, use the following URL pattern:

http://<host>:<port>/prweb/PRRestService/PegaMKTContainer/V3/CaptureResponse/Initiate

Caution: When using a multilevel Context Dictionary with actions associated to the top level context which is not the primary context, the initiation of the action will NOT load the full data context when that action expires or when a response is received. Because of that, action flows and treatments that are part of the Wait expiration or Response received paths cannot access primary context entities other than the subject of the action. If the action was communicated to the primary contact on behalf of another entity, there will be an error because the *QualifiedEntity* page cannot be loaded.

Swagger document

To help you configure the service, you can refer to the following Swagger document: Swagger document for Container and Capture Response v3.

POST Service Request

For the POST request, the service expects a JSON Object with the following attributes:

Name	Required	Description
SubjectID	Optional	Customer identifier. Set here if the same for all responses, or in RankedResults below.
ContextName	Optional	Singular user-friendly name of the context entity (for example, Customer, Account). Set here if the same for all responses, or in RankedResults below.
ContainerName	✓	Name of the Container to trigger.
RankedResults	Optional	List of actions for which you want to capture a response. Results from the Container service can be mapped into this array.

RankedResults attributes:

Name	Required	Description
SubjectID	✓	Customer identifier for the context entity
ContextName	✓	Singular user-friendly name of the context entity (for example, Customer, Account).
Name	✓	Name of the action.
Issue	✓	Issue to which the action belongs.
Group	✓	Group to which the action belongs.
InteractionID	✓	The identifier of the original decision that resulted in the actions being selected for the customer. All actions yielded for the customer in the same decision share the same InteractionID. The CaptureResponse API requires the InteractionID from the Container response in order to match the response to the original decision result, so that the adaptive models learn from the interaction. If the response is matched to the correct decision result, the merged response, with all SR properties, is captured in Interaction History. If the InteractionID isn't matched, the response is still captured in Interaction History but no adaptive model learning will take place.
Rank	✓	Action rank based on Strategy configuration.

Name	Required	Description
Direction	✓	Direction in which the interaction was initiated, Inbound or Outbound.
Channel	✓	Channel the interaction occurred on.
Outcome	✓	The customer disposition of the action.
Journey	Optional	Name of the associated customer journey.
JourneyStage	Optional	Name of the associated customer journey stage.
JourneyStep	Optional	Name of the associated customer journey step.
Category	Optional	Category to which the action belongs.
CustomerCost	Optional	Cost of this product or service to the customer.
Label	Optional	Business friendly description of the action.
Benefits	Optional	Benefits to the customer.
WhyRelevant	Optional	Why this action is relevant for the customer.
ShortDescription	Optional	Short description of the action.
PaidAudienceName	Optional	Name of the paid audience.
Identifier	Optional	Unique identifier for the action.
Pricing	Optional	Description of the pricing.
EligibilityDescription	Optional	Criteria for customers to be eligible for this action.
Propensity	Optional	Action propensity value.
AgentCompensation	Optional	Agent compensation amount if customer accepts action.
ClassIdentifier	Optional	Internal class for the action results.
Priority	Optional	Action priority based on Strategy configuration.
BundleName	Optional	Name of the action bundle.
ImageURL	Optional	URL to the associated action image.
BundleParent	Optional	Indicates if this action is a bundle parent.
Variant	Optional	Variation of the action (for example, 64gb, 128gb, 256gb).
Treatment	Optional	The name of the action treatment.
InternalCost	Optional	The internal company cost of the action.
CampaignID	Optional	Work ID of the active Campaign associated with this Container.
OfferValue	Optional	The value the customer receives from this action.
ClickThroughURL	Optional	URL to invoke the CaptureWebClickThrough API. The encrypted parameter in this URL contains the actual click-through URL specified as part of the action's details. Typically, this leads to a link or file that provides more details about the action.

A sample invocation request is shown below:

{"SubjectID":"","ContextName":"","ContainerName":"TopOffers","RankedResults": [{"SubjectID":"Customer-100","ContextName":"Subscriber","Name": "PlatinumCardOffers","Issue": "Sales","Group": "CreditCards","Rank": 1,"CampaignID": "NBA","InteractionID": "-563481321460449721","Outcome": "Accepted","Channel": "Web","Direction": "Inbound","Journey":"","JourneyStage":"","JourneyStep":""}]}

Service Response

For the response, the service returns a JSON object with the following attributes:

Name	Description
Status	OK for successful result, Error otherwise.
Message	Confirmation message when successful, error message in case of an error.

A sample service response is shown below:

{"Status": "OK","Message": "Response captured successfully" }

Accessing a customer's interaction history with the CustomerOfferHistory REST service

CustomerOfferHistory is a REST service that supports both GET and POST HTTP methods. This service returns the Interaction History records for a customer.

Pega Customer Decision Hub

The URL pattern for this service is:

http://<host>:<port>/prweb/PRRestService/PegaMKTContainer/V2/CustomerOfferHistory

Service Request

For the request, the service expects a JSON Object with the following attributes:

Name	Required	Description
CustomerID	✓	Customer identifier.

A sample invocation request is shown below:

{"CustomerID":"CONNOR",}

Service Response

For the response, the service returns a JSON object with the following attributes:

Name	Description
OfferHistory	An array of History records for the customer.

OfferHistory attributes for each record:

Name	Description
Name	Name of the action.
Issue	Issue to which the action belongs.
Group	Group to which the action belongs.
Channel	Channel through which this action was communicated
Direction	Direction of communication for this action.
InteractionID	Interaction History record identifier for this action.
ExternalID	Associated work object identifier.
OutcomeTime	Time when the outcome was captured.
Outcome	Response recorded for this action (for example, Accepted, Rejected, Impression)
Behaviour	Behavior of the specified outcome (for example, Positive, Negative, Neutral)
Identifier	Identifier for the action.

Note: For Negotiation history records, there is a nested OfferHistory element which is an array of action names.

A sample service response is shown below:

{"OfferHistory": [{"Name": "Negotiation","Issue": "Negotiation","Channel": "CallCenter","InteractionID": "7382338912749122452","ExternalID": "S-13805","OutcomeTime": "20160225T125501.526 GMT","Behaviour": "Negative","OfferHistory": [{ "Name": "Handsets_fromGoogleNexus4" },{ "Name": "DataPlan100Min" },{ "Name": "MemoryCard_V2_M3" }],"Outcome": "Rejected","Identifier": "/Negotiation/Negotiation/Negotiation","Direction": "Inbound","Group": "Negotiation"},{"Name": "MI","Issue": "Sales","Channel": "CallCenter","InteractionID": "-7714067498846096636","ExternalID": "","OutcomeTime": "20160224T101058.055 GMT","Behaviour": "Positive","Outcome": "Accepted","Identifier": "/Sales/Handsets/MI","Direction": "Inbound","Group": "Handsets"}]}

Capturing clicks on the click-through URL

CaptureClickThrough is a REST service that allows callers to request the handling of a click on the Container for an authenticated channel. This service records a click in IH before redirecting to the requested URL. Based on the Container's configuration, this service may also initiate the specified action for the customer.

Pega Customer Decision Hub

To invoke this service, callers must use the ClickThroughURL that was returned for the action by the ExecuteContainer API.

The invocation URL for this service resembles the following:

```
http://<host>:<port>/prweb/PRHTTPService/PegaMKTContainer/Services/CaptureClickThrough?
```

Configuring a container invocation script

To facilitate the invocation of Real-Time Container APIs, Pega Customer Decision Hub includes an out-of-the-box sample JavaScript file - webwb/RealTimeContainerScript.js

Pega Customer Decision Hub

This script includes a service controller object which has methods that invoke the various Container APIs.

To utilize this script, perform the following steps:

1. Download and copy the JavaScript file to your web directory.
2. Include the file in your web content. For example:

   ```
   <script src="https://docs-previous.pega.com/realtimecontainerscript.js" />
   ```

3. Instantiate the service controller object. For example:

   ```
   var nbamServiceCtrl = getNBAMServiceControl();
   ```

   Tip:

   To invoke a Container with the V2 service, instantiate:

   ```
   var nbamServiceCtrl = getNBAMServiceControl()("V2");
   ```

   To invoke multiple Containers with the V2 service, instantiate:

   ```
   var nbamServiceCtrl = getNBAMServiceControl()("V2", true);
   ```

4. Initialize the service controller object using one of the following mechanisms.
   1. Directly update serverHostName and serverPort in realtimecontainerscript.js, as shown in the snippet below. When using this mechanism, the instantiation of the service controller (Step 3 above) will already initialize the controller using these values.

      ```
      var serverHostName = "localhost";var serverPort = "8080";
      ```

   2. Invoke the initialize method on the service controller after instantiating it. For example:

      ```
      nbamServiceCtrl.initialize(window.location.hostname, window.location.port);
      ```

5. Invoke the desired API using the corresponding service controller method.

Service API	JavaScript Method	Method Parameters
Container	getOffers	customerID, containerName, channel, previousPage, currentPage, callback
V2 Container	loadOffers	jsonObj, methodType, callback
CaptureWebImpression	captureSingleWebImpression	containerID, customerID, offerID, issue, group, interactionID, callback
CaptureWebResponse	captureWebResponse	containerID, customerID, offerID, issue, group, interactionID, outcome, behavior, channel, direction, callback
CaptureResponse	captureResponse	containerID, customerID, offerID, issue, group, interactionID, outcome, behavior, channel, direction, callback
CapturePaidResponse	capturePaidClickResponse	CustomerID, ExternalAudienceId, ReferrerUrl, Utm_medium, callback

Invoking a fallback strategy

If a Strategy execution fails due to an exception, Pega Customer Decision Hub invokes a fallback Strategy. The fallback Strategy results in a response when a service activity is executed. To configure the required action data, save the strategy *FallbackStrategy* into your implementation layer.

Pega Customer Decision Hub

When your Container Service exceeds the Service Level Agreement defined for the Container Service by more than a specified number of consecutive violations, Pega Customer Decision Hub invokes a fallback service activity instead of the default service activity. The fallback service activity returns the default set of actions for every request. To change the set of actions, save the *FallBackContainerResponse* data transform into your implementation layer and configure the default Container Service response.

A sample response from the invocation of this service is shown below:

For example:


```
"ContainerList": [
    {   "Status": "OK",
        "ContainerName": " TopOffers ",
        "RankedResults": [
          {
            "Group": "Default",
            "Issue": "Fallback",
            "InteractionID": "-8654833423861133119",
            "Category": "Category",
            "CustomerCost": "0",
            "WhyRelevant": "WhyRelevant",
            "Benefits": "Benefits",
            "Direction": "Direction",
            "Name": "FallbackOffer",
            "ShortDescription": "ShortDescription",
            "PaidAudienceName": "PaidAudienceName",
            "Identifier": "Identifier",
            "Pricing": "Pricing",
            "JourneyStage": "JourneyStage",
            "JourneyStep": "JourneyStep",
            "EligibilityDescription": "EligibilityDescription",
            "Propensity": "0",
            "AgentCompensation": "0",
            "Priority": "0",
            "ClassIdentifier": "Offer",
            "Channel": "Channel",
            "ImageURL": "ImageURL",
            "Rank": "1",
            "BundleParent": "BundleParent",
            "Variant": "Variant",
            "Treatment": "Treatment",
            "InternalCost": "0",
            "CampaignID": "CampaignID",
            "OfferValue": "0",
            "ClickThroughURL": "ClickThroughURL",
            "Journey": "Journey"
          }
        ]
    } ]
```


Identifying customers who interact with a real-time container

Identity matching allows marketers to track anonymous individuals and their interactions with Real-Time Containers in the inbound real-time channels, such as web self-service, call center, and mobile applications. When a marketer defines a Container to present some action in the inbound channel, the container captures associations to perform identity matching.

Pega Customer Decision Hub

For example, when a marketer defines a Container to represent a portion of a web page, the Container uses browser cookies to perform identity matching.

The following associations are captured during interactions with the Real-Time Container:

- CustomerID to UniqueID
- CustomerID to IP Address

Additionally, identity matching is implicitly performed by the following Real-Time Container services:

- Container

- Identifying customers with identity matching
- Viewing how anonymous users interact with Pega Customer Decision Hub actions

Identifying customers with identity matching

Identity matching is a feature that allows a Pega Customer Decision Hub application to identify individuals across their addressable devices. This feature also enables a seamless transition between an anonymous individual and a recognized customer.

Pega Customer Decision Hub

When an individual is anonymous (e.g. not logged in), relevant information is captured during their various interactions with the application. This information can then be utilized to determine the Next-Best-Action for this individual. When the individual authenticates themselves as a customer, their anonymous interaction data is merged with their previous interaction history. This allows the application to maintain a complete view of the customer journey.

Identity matching is incorporated in the following functional areas:

- Email
- Microsites
- Real-Time Containers

Identity matching utilizes browser cookies to store identifying information. For individuals yet to be identified, a unique id is stored in the cookie. Once an individual has been identified as a customer, their (encrypted) customer id is stored. During the individual's interactions with the application, additional data is collected and associated with the appropriate identifier (customer id or unique id). The captured associations are specific to the functional area and are listed in the corresponding section.

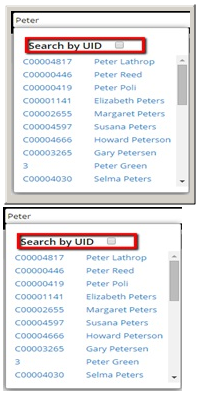
Viewing how anonymous users interact with Pega Customer Decision Hub actions

You can use Marketing Profile to search for anonymous users and view their interactions with the application.

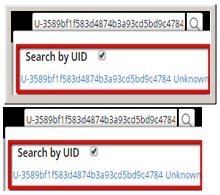
Pega Customer Decision Hub
Note: To allow for the searching of anonymous users, the MKTProfileEnabled When rule must be enabled (i.e. set to true).

The Marketing Profile search functionality provides an additional **Search by UID** checkbox to allow for searching of anonymous users.

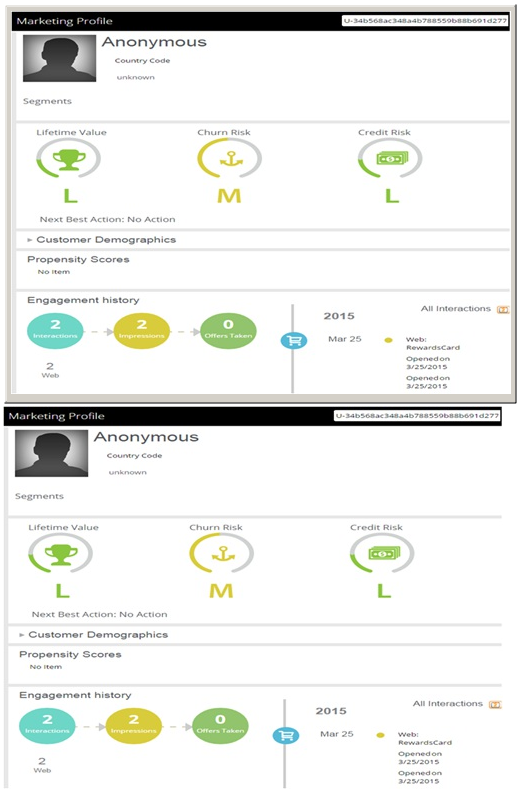
When this checkbox is unchecked, the search looks for existing customers based on their identifier and name.



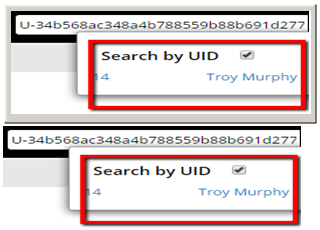
When this checkbox is checked, the search looks for anonymous customers using their unique identifier.



Upon clicking the link for the anonymous customer, their interaction details are displayed.



At a later point, if the anonymous individual identifies themselves as a known customer, the anonymous interaction details are merged with the customer's interaction history. Searching for the anonymous individual (by their unique id) now reveals the identified customer in the search results.



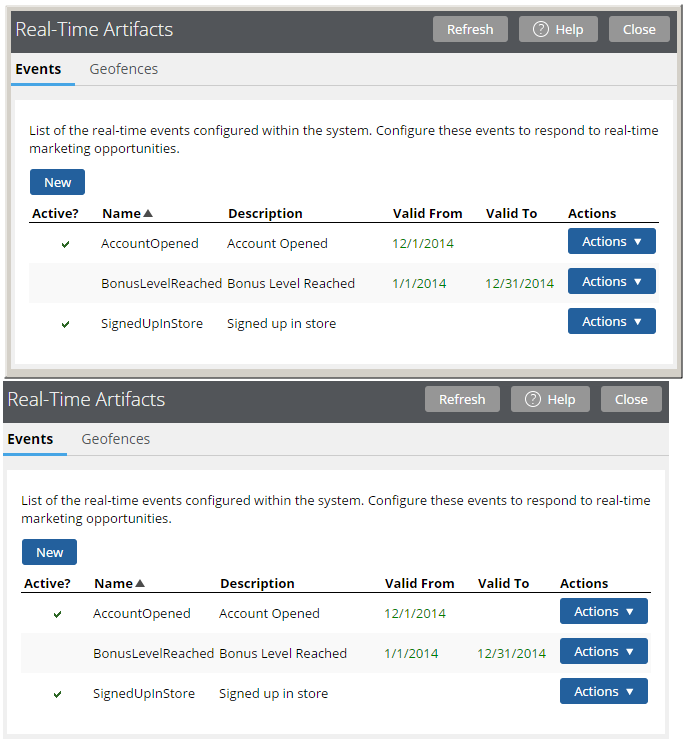
Responding to real-time customer engagement with real-time event triggers

Events provide a mechanism for responding to real-time marketing opportunities. An Event is essentially an entry-point into the application which can be initiated by internal or external systems. Events can be mapped to Multi-Channel Campaigns and can trigger their execution.

Pega Customer Decision Hub

Real-Time Events are listed on the **Events** tab of the Real-Time Artifacts landing page. This landing page is accessible in the Pega Customer Decision Hub portal via: **Content**

Real-Time Artifacts .



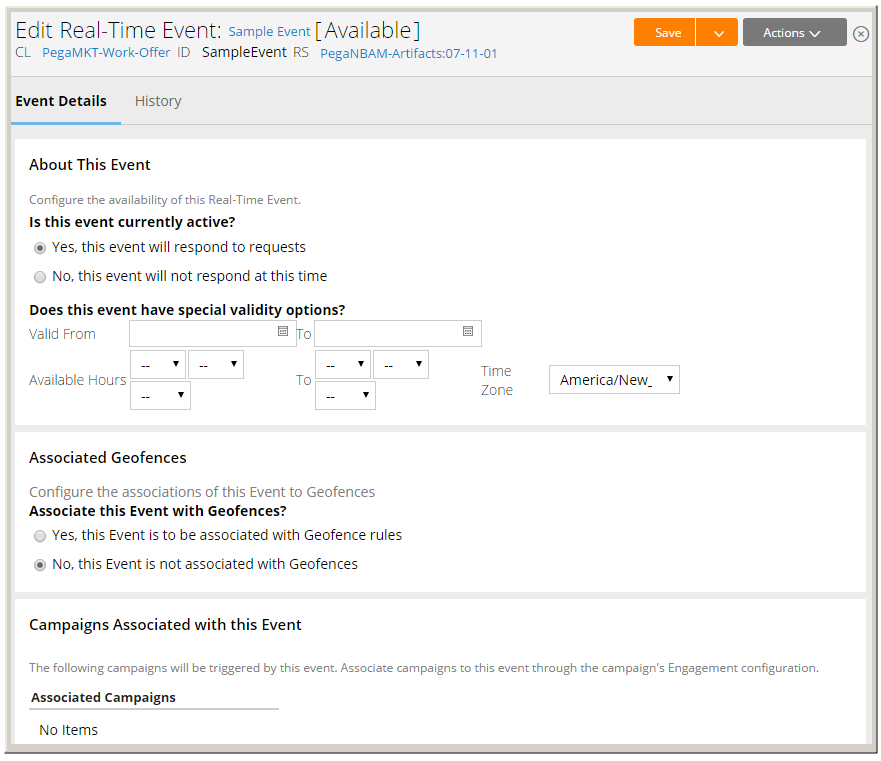
The Events landing page provides a quick overview of the Events rules configured in the application. The landing page indicates whether an Event is active or inactive. The **Valid From** and **Valid To** values (if configured) are displayed in green if they are currently satisfied.

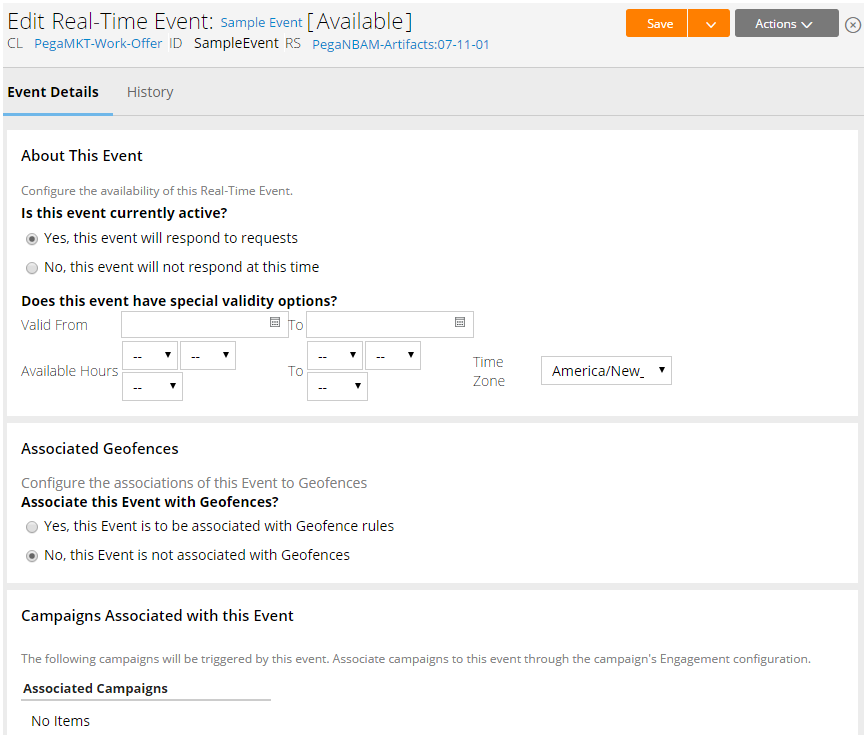
- Creating a real-time event
- Triggering a real-time event with the Event Stream service
- Triggering an Event with the SOAP, HTTP and REST services
- Using a geofence to trigger a real-time event
- Integrating real-time events with Pega Event Strategy Manager

Creating a real-time event

Users can use the **Create** link in the Events landing page to create a new Event rule. A new Real-Time Event rule resembles the following:

Pega Customer Decision Hub





The Event rule has the following tabs:

Tab Name	Purpose
Event Details	Specify the availability details for the Event and associate Geofences with the Event. View Campaigns associated with this Event.
History	Specify the Description and Usage information for this Event.

Triggering a real-time event with the Event Stream service

Use the out-of-the-box Event Stream service to trigger Events and respond to them immediately. This method is recommended if the client calls are coming from within the cluster that is running Pega Customer Decision Hub.

Pega Customer Decision Hub
You can create real-time runs for data flows that have a data set that can be streamed in real-time as the primary input. Data flow runs that are initiated through the Data Flows landing page process the data using the checked-in instance of the Data Flow rule and the rules that are referenced by that Data Flow rule.

1. Log in to Pega Customer Decision Hub as an operator with access to Dev Studio.
2. Click **Configure Decisioning Decisions Data Flows Real-time processing** .
3. On the **Real-time processing** tab, click **New**.
4. Associate a Data Flow rule with the data flow run by doing the following steps:
   a. In the **Applies to** field, select the event class, *PegaMKT-Data-Event*.
   b. In the **Access group** field, select *PegaNBAM:Agents* as the access group context for the data flow run.
   c. In the **Data flow** field, select the *ProcessFromEventSource* Data Flow rule.
   d. In the **Service instance** name field, select *RealTime*.
5. **Optional:** Configure the number of threads for the data flow nodes.
   The recommended default value is 5. The number of threads should not exceed the number of cores on the data flow nodes. Check your hardware specifications in order to determine the maximum number of possible threads to use.
6. **Optional:** To keep the run active and restarted automatically after every modification, select **Manage the run and include it in the application** , and then select the ruleset.
   If you move the ruleset to a new environment, the application will move the run with the ruleset and keep it active.
7. **Optional:** Specify any activities that you want to run before the data flow starts or after the data flow run has completed.
   a. Expand the **Advanced** section.
   b. In the **Additional processing** section, perform the following actions:
      - Specify a preprocessing activity that you want to run before running the data flow.
      - Specify a postprocessing activity that you want to run after running the data flow.
8. **Optional:** In the **Resilience** section, specify the data flow run resilience settings for resumable or non-resumable data flow runs. You can configure the following resilience settings:

   Record failure
   - **Fail the run after more than x failed records** – Terminate the processing of the data flow and mark it as failed after the threshold for the allowed total number of failed records is reached or exceeded. If the threshold is not reached or exceeded, the data flow run finishes with errors. The default value is 1000 failed records.

   Node failure
   - **Restart the partitions on other nodes** – For non-resumable data flow runs, transfer the processing to the remaining active Data Flow service nodes. The starting point is based on the first record in the data partition. With this setting enabled, each record can be processed more than once.
   - **Fail the entire run** – For non-resumable data flow runs, terminate the data flow run and mark it as failed when a Data Flow service node fails. This setting provides backward compatibility with previous versions of Pega Platform.

   Snapshot management
   - **Create a snapshot every x seconds** – For resumable data flow runs, specify the elapsed time for creating snapshots of the data flow runs state. The default value is 5 seconds.

9. Click **Done**. **Result:** Data flow nodes are required to start the run. If the service contains no nodes in the cluster, a message is displayed with a link to the **Services** landing page, where you can add nodes.
10. To analyze the lifecycle of the run and troubleshoot potential issues, in the **Run details** tab of the data flow run, click **View Lifecycle Events**. **Result:** In the window that opens, each event has a list of details, for example, reason, which you can analyze to better understand the event or debug an issue. You can use the **Actions** menu to export the events to a single file.**Note:** By default, events from the last 10 days are displayed. You can change this value by editing the **dataflow/run/lifecycleEventsRetentionDays** dynamic data setting.

What to do next:

When an event is triggered, the system determines whether the referenced event is valid, enabled, and available (date and time are valid). If so, any campaigns that are

associated with the events are initiated. The payload for the event (including customer id) is propagated to the campaign, and from there to the strategy, and finally to the action. Refer to the Multi-Channel Campaigns chapter for more information on associating Events with Campaigns.

Pega Customer Decision Hub 8.6 includes the *CDHEventSource* data set and the related *ProcessFromEventSource* data flow which can be used to trigger real-time events. In addition, you can configure an external Kafka cluster for event processing, as described in Configuring Kafka to process real-time events.

Triggering an Event with the SOAP, HTTP and REST services

If the client calls are coming from outside the cluster that is running Pega Customer Decision Hub, you can use the SOAP, REST and HTTP services to trigger Events.

Pega Customer Decision Hub

Using the built-in SOAP and HTTP services

Pega Customer Decision Hub provides built-in support for triggering events with the SOAP and HTTP services. The following ryles are available:

- *Rule-Service-SOAP*
- *Rule-Service-HTTP*

Important: If you are planning to use the predefined SOAP or HTTP entry points in conjuction with the Event Stream service, you must create the following dynamic system setting to send the events to the Event Stream service:

- **Rule set** - PegaMKT-Engine
- **Name** - MKTKafkaEvents
- **Value** - true

The following URL provides the WSDL for your application's Event handling SOAP Service: http://<host>:<port>/prweb/PRSOAPServlet/SOAP/PegaMKTDataMktEvent/Services?WSDL

The following is an example request triggering the AccountOpened Event, passing in the Customer's ID, their location (latitude/longitude), and information about the Event:

```
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:even="EventPayload"
xmlns:urn="urn:PegaRULES:SOAP:PegaMKTDataMktEvent:Services">
  <soapenv:Header>
    <even:EventPayload>?</even:EventPayload>
  </soapenv:Header>
  <soapenv:Body>
    <urn:EventPayload>
      <CustomerID>JR</CustomerID>
      <EventName>AccountOpened</EventName>
      <EventType>Branch</EventType>
      <Latitude>42.9613</Latitude>
      <Longitude>-71.4798</Longitude>
    </urn:EventPayload>
  </soapenv:Body>
 </soapenv:Envelope>
```

Using a REST service

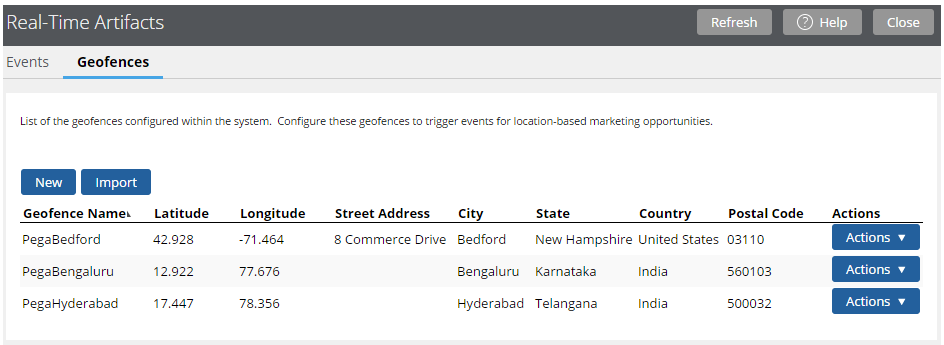
Pega Customer Decision Hub does not provide a built-in REST service to trigger events. However, you can define one in your rule set layer and configure it to write to the Stream data set, similar to how the HTTP or SOAP services do it. For an example, see the entry point activity *HandleHTTPEvent* under *PegaMKT-Data-Event*.

Using a geofence to trigger a real-time event

A Geofence is a virtual boundary around a particular location. You can define Geofences in the system and then use them to trigger Events. For example, a customer entering (or nearing) a store could trigger the Geofence for that store. This could in turn trigger an Event whereby the customer is presented with an appropriate action. You can also reference Geofences in strategies and can harness this to determine the most relevant action for a particular customer.

Pega Customer Decision Hub

Geofences are accessible in the Pega Customer Decision Hub portal, via: **Content Real-Time Artifacts Geofences** .



The Geofences landing page provides a quick overview of the various fences configured in the application. The Geofence rule can be opened by double-clicking the Geofence row.

- Understanding geofences
- Creating a geofence
- Importing geofences
- Associating a real-time event with geofences
- Understanding geofence services
- Understanding the Preferred Geofences services
- Understanding the Detect/Trigger services

Understanding geofences

The rendered map is a Google map and allows for various interaction patterns, including the following:

Pega Customer Decision Hub

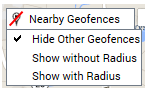
- Zooming in and out
- Panning in all directions
- Using Street View
- Viewing satellite imagery

In addition to the above common interactions, the following Geofence-specific interactions are also available:

- Dragging the Geofence pin  to relocate the Geofence's location. The details section will display the new address and coordinates.
- Resizing the Geofence by dragging any of the four endpoints (north, south, east, and west). The details section will display the new radius value.

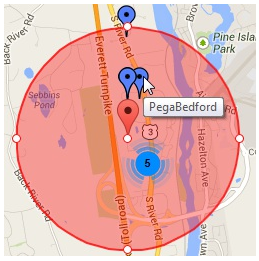
Viewing Nearby Geofences

The map on the Geofence rule form can also be used to view nearby Geofences. This is helpful in determining potential overlaps across Geofences. By default, nearby Geofences are not displayed. Use the "Nearby Geofences" menu on the map to toggle the display of nearby fences.

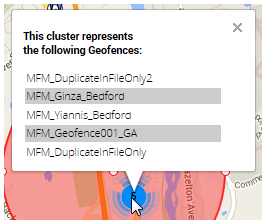


The following three options are available for nearby Geofences:

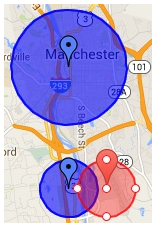
1. **Hide Other Geofences** - This is the default option. In this mode, nearby Geofences are not displayed.
2. **Show without Radius** - Select this option to view nearby Geofence locations. Each Geofence is represented by a blue pin and the radius of the Geofence is not displayed. Users can hover over the pin to view the name of the Geofence.

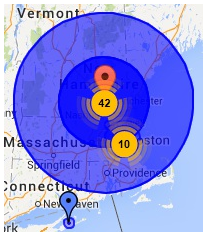
   If multiple Geofences are located in too close a radius to render separately, they are represented together by a circle with the number of Geofences it contains. Users can click on the number to view a list of the contained Geofences.

3. **Show with Radius** - Select this option to view nearby Geofences along with their radii. Each neighboring Geofence is represented by a blue pin as its center and a blue circle corresponding in size to its radius.

   As previously, if there is a high density of Geofences, they are represented by a circle with the number of contained Geofences. The shading of the circle also provides a visual cue to its density. Denser circles will be more opaque than less dense circles.

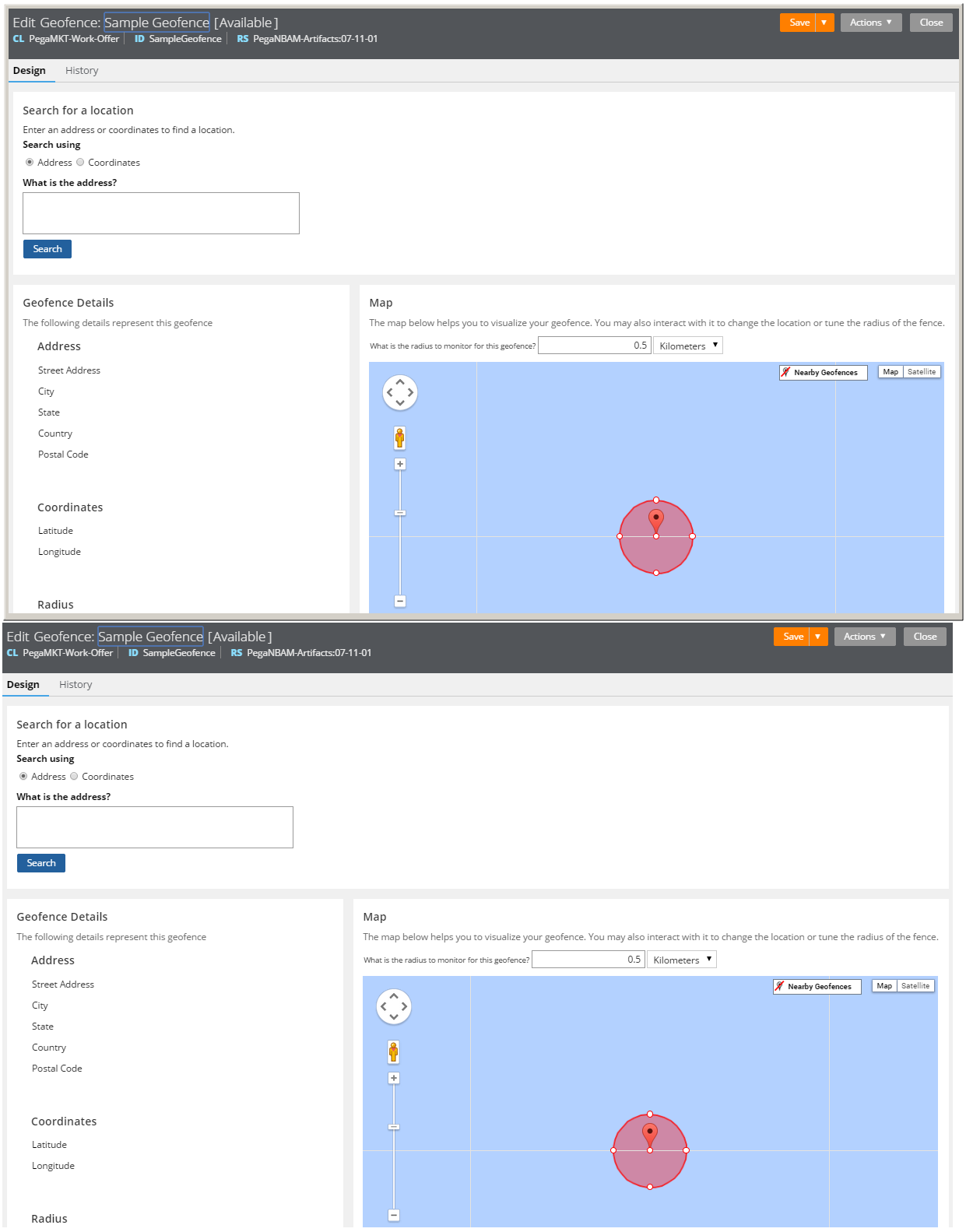
   

Creating a geofence

Users can use the **Create** link in the Geofences landing page to create a new Geofence.

Pega Customer Decision Hub

A new Geofence rule resembles the following:

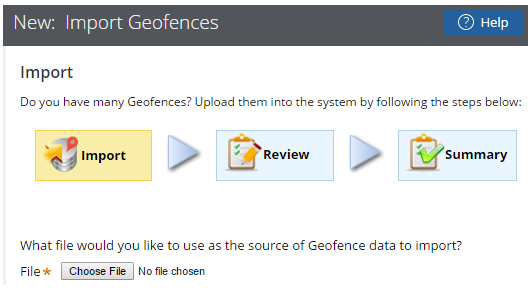


The Geofence rule has the following tabs:

Tab Name	Purpose
Design	Specify the location details and radius of the Geofence. View other nearby fences.
History	Specify the Description and Usage information for this Geofence.

Importing geofences

Use the **Choose File** button to select the comma-separated values (CSV) file containing the list of Geofences to import. After selecting the CSV file, use the “Next” button to proceed to the review stage.



Notes on the import file:

- File must be in CSV format
- The entries in the CSV must be in the following order:

  Geofence Name, Radius, Radius Unit, Latitude, Longitude, Street Address, City, State, Postal Code, Country

- Geofence Name must be provided
- Either location coordinates (latitude/longitude) or address (street/city/state...) must be provided for each Geofence, but not both.
- Radius unit should be either “Miles” or “Kilometers”. If radius unit is not provided, it is defaulted to “Kilometers”.
- If radius is not provided, it is defaulted to 1

Example Geofence entries:

- Using coordinates:

  PegaBedford,2,Miles,42.928,-71.464,,,,,

- Using address fields:

  PegaBedford2,2,Kilometers,,,8 Commerce Drive,Bedford,NH,03110,USA

Associating a real-time event with geofences

Events that have been associated to this Geofence are displayed in the “Associated Events” grid. When a customer enters this Geofence, the Events listed in the grid will be triggered. Refer to the Events chapter for details on associating Geofences with Events.

Pega Customer Decision Hub

Associated Events
Sample Event with Geofences
Signed up in store

Understanding geofence services

In addition to configuring Geofence rules in Pega Customer Decision Hub , the app that wishes to detect when a Geofence is entered (tripped) must make service requests to the system to determine if a Geofence is tripped. If a Geofence is tripped, all Events associated with it will be triggered (assuming they are currently applicable). The system provides multiple services which can be used separately or in tandem to facilitate this functionality.

Pega Customer Decision Hub

Understanding the Preferred Geofences services

The Preferred Geofences set of services can be used to determine the most relevant set of Geofences for a particular location. For example, when a customer launches the app on their device, the app can invoke the service to determine the list of nearby Geofences to monitor. The app can then provide this list to the device’s OS to monitor, thereby avoiding having to continually poll whether the current location has tripped a Geofence.

Pega Customer Decision Hub

The following two variants of the service exist:

- **HandleMktPreferredFences HTTP Service**

  http://<host>:<port>/prweb/PRHTTPService/PegaMKTGeolocation/Services/HandleMktPreferredFences?Key=Closest&BaseLatitude=<DeviceLatitude>&BaseLongitude=<DeviceLongitude>&ClosestCount=<NumberOfGeofences>

- **HandleMktPreferredFences REST Service**

  http://<host>:<port>/prweb/PRRestService/PegaMKTGeolocation/Services/HandleMktPreferredFences/Closest?BaseLatitude=<DeviceLatitude>&BaseLongitude=<DeviceLongitude>&ClosestCount=<NumberOfGeofences>

Both variants expect the following inputs:

- **BaseLatitude** – Latitude of the device
- **BaseLongitude** – Longitude of the device
- **ClosestCount** – Number of closest Geofences to return for the specified location. If this value is not provided, the system returns up to ten Geofences.

The content type for the response differs between the variants:

Service Type	Response Content Type
HTTP	text/plain (content is still JSON)
REST	application/json

A sample (formatted) response is shown below:

{ GeolocationArray : [{ "GeofenceName" : "PegaBedford", "RadiusNormalized" : "1000", "Latitude" : "42.928", "Longitude" : "-71.464" }, { "GeofenceName" : "ManchesterBRAirport", "RadiusNormalized" : "3218.688", "Latitude" : "42.928", "Longitude" : "-71.438" }] }

In error scenarios, such as when the location coordinates are not provided, an error response is returned.

{ ErrorsList : [{ "pyMessage" : "Error: One or more input parameters are invalid." }] }

The system provides out-of-the-box support for closest Geofence lookup (“Key” request parameter is set to “Closest” in the example URLs above). Custom implementations may extend this functionality to provide support for other lookup modes.

Understanding the Detect/Trigger services

The Detect/Trigger set of services are used to detect whether a customer's device has tripped a Geofence in the system and to trigger Events associated with tripped Geofences. These services support handling requests for multiple customers in a single request.

Pega Customer Decision Hub

The following two variants of the service exist:

- **HandleMktGeolocations HTTP Service**

  http://<host>:<port>/prweb/PRHTTPService/PegaMKTGeolocation/Services/HandleMktGeolocations?CustomerID=<CustomerID1,2,..n>&DeviceID=<DeviceID1,2,..n>&DeviceLatitude=<DeviceLatitude1,2,..n>&DeviceLongitude=<DeviceLongitude1,2,.n>

  When using this service for multiple customers, ensure that each of the values are comma-separated (e.g. CustomerID=A1,A2,B1) and that the length of each of the request fields (such as CustomerID, DeviceLatitude, etc.) is the same.

- **HandleMktGeolocations SOAP Service**

  The following URL provides the WSDL for your this service: http://<host>:<port>/prweb/PRSOAPServlet/SOAP/PegaMKTGeolocation/Services?WSDL

  The following is an example SOAP request for two customers:

  <soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope" xmlns:geol="GeolocationPayload"> <soap:Header/> <soap:Body> <geol:GeolocationPayload> <Geolocation> <CustomerID>JR</CustomerID> <DeviceID>A

  Both variants of this service expect the following inputs:

  - **CustomerID** – Customer identifier
  - **DeviceID** – Device identifier
  - **DeviceLatitude** – Latitude of the device
  - **DeviceLongitude** - Longitude of the device

For each customer location, the system evaluates whether the location trips any of the defined Geofences. For each Geofence that is tripped, the system triggers all Events associated to that Geofence.

Integrating real-time events with Pega Event Strategy Manager

Event Strategy Manager (ESM) is a Pega Platform feature that enables users to design and execute event strategies that detect patterns in streams of customer data and react to these patterns by taking the Next Best Action for the customer.

Pega Customer Decision Hub

Further, users can integrate ESM with Pega Customer Decision Hub Real-Time Events. This integration enables the triggering of a Real-Time Event in Pega Customer Decision Hub by an ESM Data Flow.

Pega Customer Decision Hub provides integration support for ESM in a variety of deployment scenarios:

1. Pega Customer Decision Hub and ESM are deployed on same instance, ESM Data Flows have access to the Pega Customer Decision Hub application.
2. Pega Customer Decision Hub and ESM are deployed on same instance, ESM Data Flows do not have access to the Pega Customer Decision Hub application.
3. Pega Customer Decision Hub and ESM are deployed on different instances.

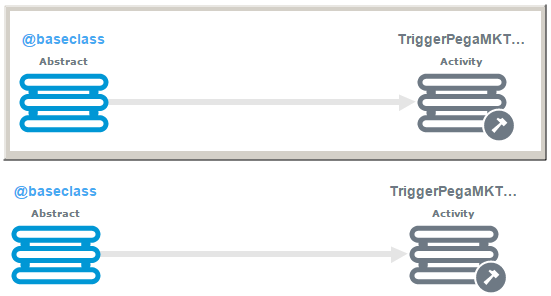
In scenario 1, integrators can directly use a hook activity. For scenarios 2 and 3, integrators will need to use an integration ruleset.

Directly use hook activity

In this scenario, the integration steps are minimal since the two components are both available as part of the same application. To trigger a Real-Time Event from an ESM Data Flow, configure the destination of the Data Flow to be the TriggerPegaMKTEvent activity.

Use the parameters of this activity to specify the values necessary for triggering the Real-Time Event, such as EventName and CustomerID. The EventType parameter determines which Event service will be invoked. Pass in either HTTP or SOAP for this parameter based on the protocol you wish to utilize.

The SampleCEPDF Data Flow rule provides an example configuration that invokes the hook activity.



Use integration ruleset

In scenarios where the ESM application doesn't have access to Pega Customer Decision Hub, integrators need to first add the PegaMKT-Integration ruleset to their ESM application.

Note: The PegaMKT-Integration ruleset is provided in PegaMarketing_Integration_0722.zip, which is located under the Rules directory of your Pega Customer Decision Hub installation.

This ruleset includes the following items that enable the triggering of Pega Customer Decision Hub Events:

- *TriggerPegaMKTEvent* Activity rule - Use this activity (as described previously) to trigger a Real-Time Event in Pega Customer Decision Hub.
- *ConnectMKTURL* Dynamic System Setting - Update this setting to the URL where the Pega Customer Decision Hub application is deployed.
- *MKT_Integration:SysAdmins* Access Group - Update this access group to reference the application that has access to the PegaMKT-Integration ruleset.

Increasing customer engagement with microsites

A Microsite is a collection of one or more pages which can function as a standalone entity. Microsites can be also be embedded into existing websites. They can be used for various marketing purposes, such as product announcements and lead generation.

Pega Customer Decision Hub

Microsites are listed in the Microsites landing page, which is accessed from the Pega Customer Decision Hub portal via: **Content Microsites** .

Note: Pega Customer Decision Hub leverages the Pega Platform stage-based Case Management functionality for building Microsites. Refer to official documentation for more details on this functionality.

- Creating a microsite

- Configuring microsite stages
- Embedding a microsite link in an email treatment
- Launching a microsite
- Identifying customers who respond to a microsite
- Managing microsite subscriptions

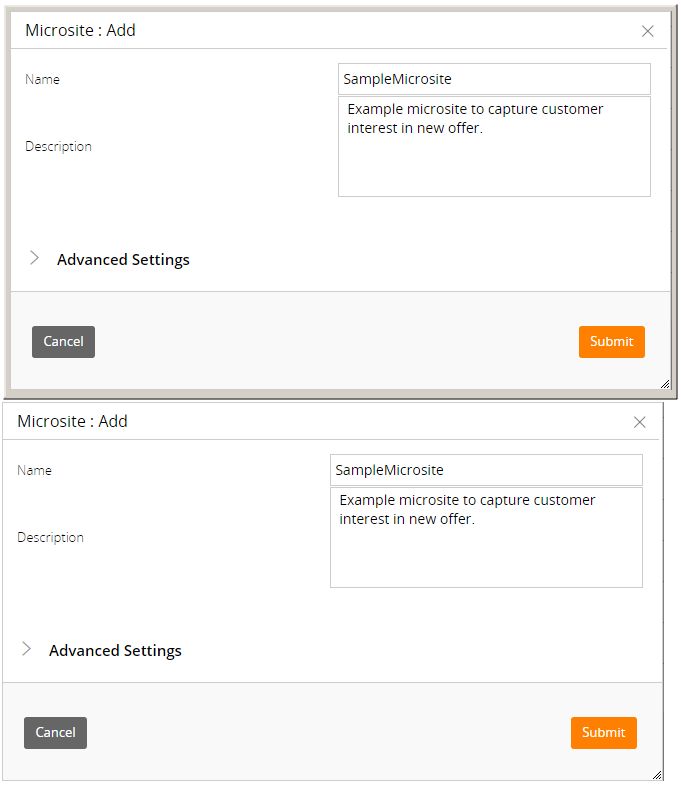
Creating a microsite

To create a new Microsite, use the **Create** button on the Microsites landing page.

Pega Customer Decision Hub
Note: The **Create** button is only visible to users that have the CanCreateMicrosite Privilege. The PegaNBAM_FW:MicrositeAdmin Access Role encapsulates this privilege and can be added to desired Access Groups.

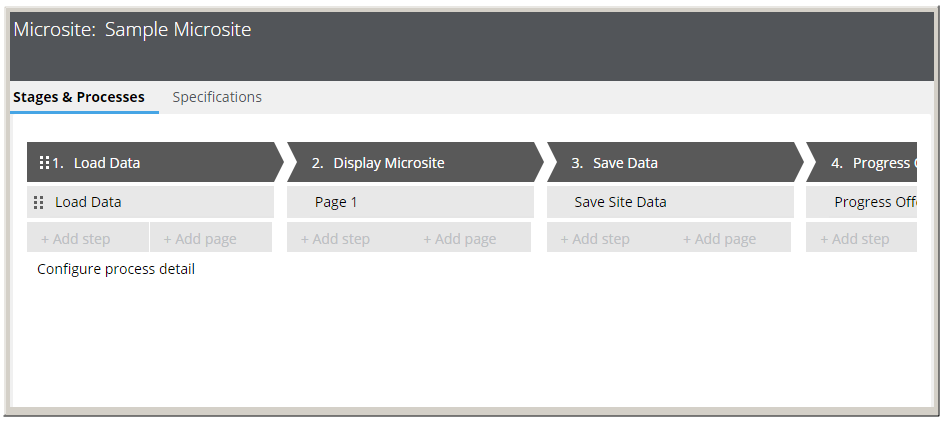
Start Control Panel

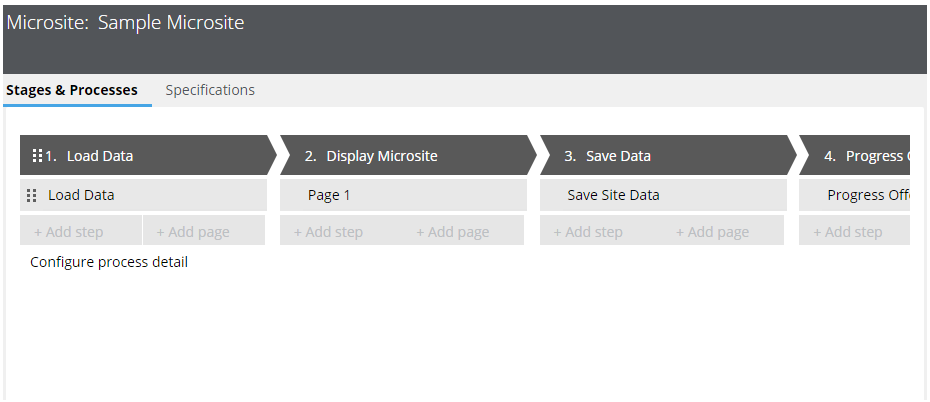
Provide a name and description for the Microsite. Users can also utilize the **Advanced Settings** section to configure advanced options, such as inheritance and ruleset information.



To nest the new Microsite under an existing one, select the existing Microsite in the **Derives From (Pattern)** field (under Advanced Settings).

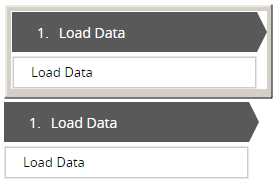
Upon clicking **Submit**, the system creates a new class for the Microsite (with class name PegaMKT-Work-Microsite-<MicrositeName>). The system also creates starter rules for the Microsite in this class and then presents the stage-based Microsite view.





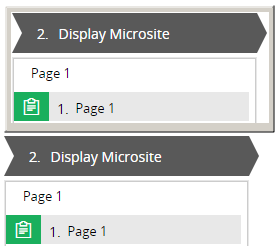
A newly created Microsite starts out with the following stages:

- **Load Data**

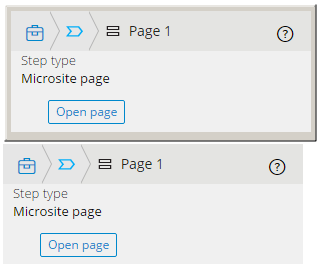


In this stage, customer and action data (if available) is loaded so that it can be used by later stages. For this data to be loaded, at runtime, the Microsite must be launched from a customer and/or action-aware context. An example of such a context would be an action email that was sent to a customer with a link to the Microsite.

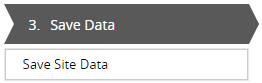
- **Display Microsite**



In this stage, the Microsite is displayed to the customer. A new Microsite starts with a single page, titled **Page 1**. Behind the scenes, the system has automatically generated corresponding Flow, Flow Action, and Section rules to support this page. Primarily, users should only modify the section to specify the contents of the Microsite. To do so, click the step in the stage to select it. Then, click the **Configure view** button in the right pane to open the backing Section rule.



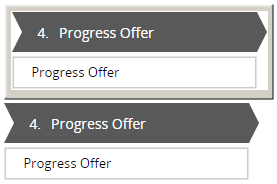
- **Save Data**



In this stage, values entered by the customer on Microsite pages are saved. This stage saves two types of data:

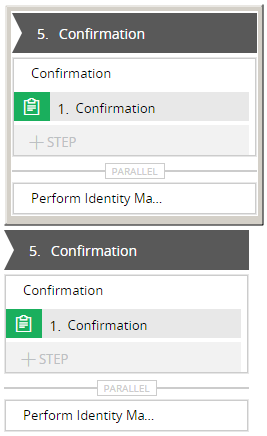
  - Action data – These values can later be used in the progressed flow.
  - Custom Site data – These values are custom fields created and specified by the users. See Defining and Using Custom Data Fields for details.

- **Progress Offer**



In this stage, the action that initiated the Microsite (e.g. via an email containing a link to the Microsite) will be progressed along the ResponseReceived path. By default, the response value is set to Accepted.

- **Confirmation**



In this stage, a confirmation page is displayed to the customer. The identifier of the out-of-the-box generated confirmation page Section rule is Confirm.

Note: The system performs identity matching at relevant stages in the Microsite case lifecycle. Refer to the Identity Matching in Microsites section in the Identity Matching chapter for details.

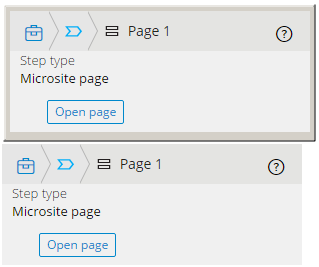
Configuring microsite stages

Each stage can have one or more steps that it performs. Typical steps for Microsites are provided out-of-the-box and are associated with the default stages, e.g. load data, save site data, progress action, etc.

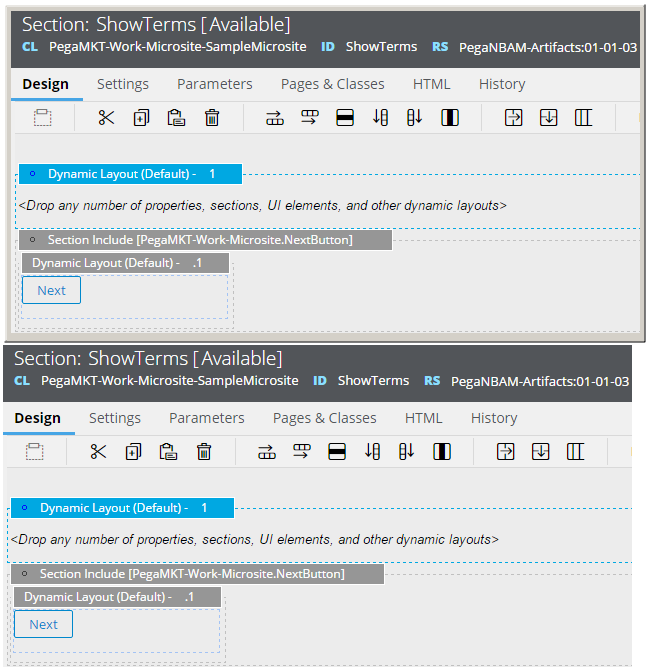
Pega Customer Decision Hub

A marketing-specific step type is the page. Page steps are quick ways to inject a screen (or page) into the Microsite process. To add a new page, the user can select the **+STEP** item in the steps area of a stage. After the user specifies the desired page name, the system will automatically create the backing structure (Flow, Flow Action, and Section).

To open the generated Section rule, users can select the page step and click the **Configure view** button in the right pane.



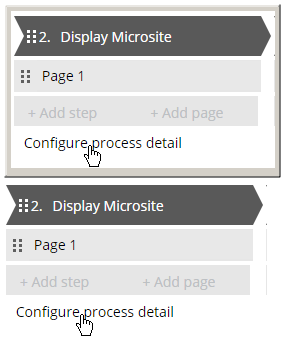
A newly created section is shown below.



Users can customize the section content to include any pertinent information to display to or gather from the customer. The **Next** button in the section can be customized as well. This can be done by overriding the NextButton section in the Microsite class and applying any desired styles to it.

In the Microsite stage view, the **X** icon on a step facilitates the deletion of the step while the four-directional cursor allows for steps to be moved to their desired position within a stage. When a step is deleted from a stage, the backing artifacts are not deleted and this page step can be added back to the stage at a later point.

For advanced configuration, select the process item (first item) in a stage and then use the **Open process** link in the right pane to open the backing Flow rule for the selected stage.

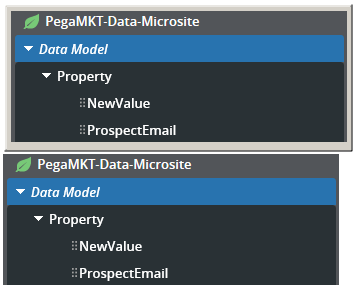


- Defining custom fields for microsites
- Configuring a microsite to display and save custom site data
- Configuring a microsite to be aware of customer and action context
- Configuring a microsite to launch a different microsite

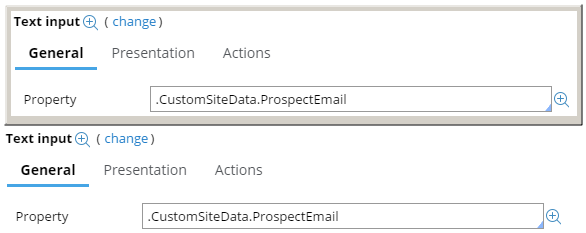
Defining custom fields for microsites

Users can define custom data fields to use in the pages of the Microsite. Any such field should be defined in the PegaMKT-Data-Microsite class. In the following example, two data fields NewValue and ProspectEmail have been defined in the custom data class.

Pega Customer Decision Hub



These fields can then be used in a Microsite page to display or capture information. Access these fields via the CustomSiteData page property on the Microsite class.

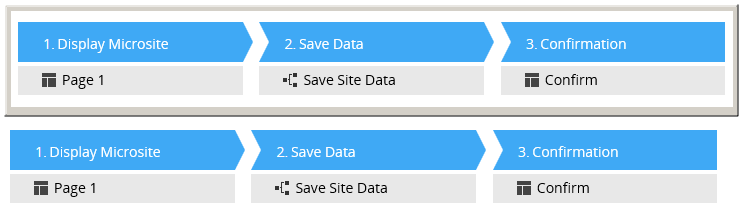


Use the **Save Site Data** step to save the values entered by the customer for custom data fields.

Configuring a microsite to display and save custom site data

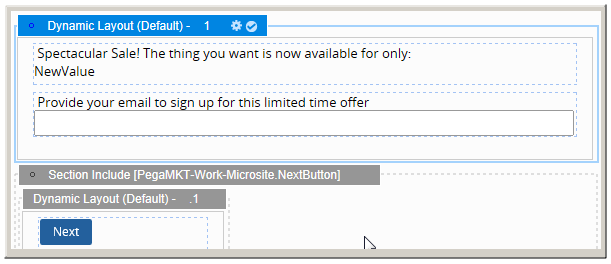
The following Microsite serves as a basic example of displaying and saving custom site data. This Microsite is not action- or customer-context aware.

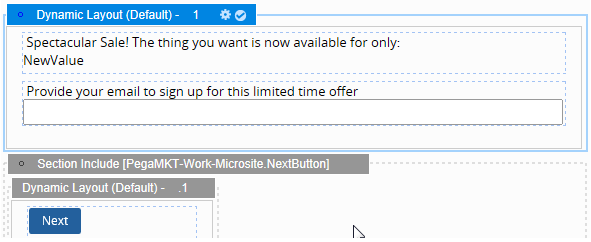
Pega Customer Decision Hub



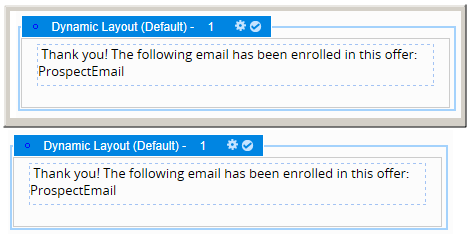
This Microsite shows a simple page with the discounted value for the product and provides an input field for customers to sign up.

The Page1 Section rule uses two custom site fields (via the .CustomSiteData page property). A snippet of this section is shown below:

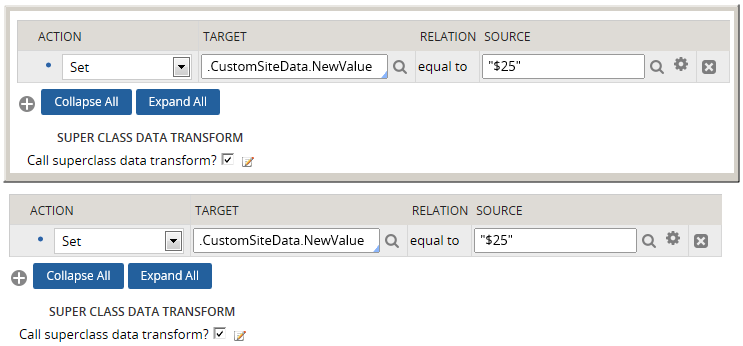




The confirmation section simply displays the previously entered (and then saved) email.

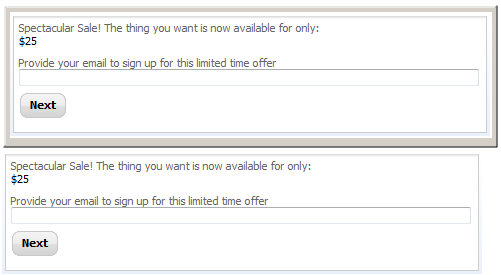


Since this Microsite does not have an action context, the NewValue custom data property is initialized via the pyDefault Data Transform rule in the class of this Microsite.

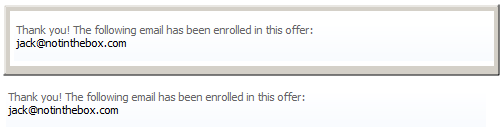


Note: It is critical that this Data Transform must be in the Microsite -specific class and must have the **Call superclass data transform?** checkbox enabled.

Upon launching the Microsite launch URL with Site=SampleMicrosite, the following is displayed:



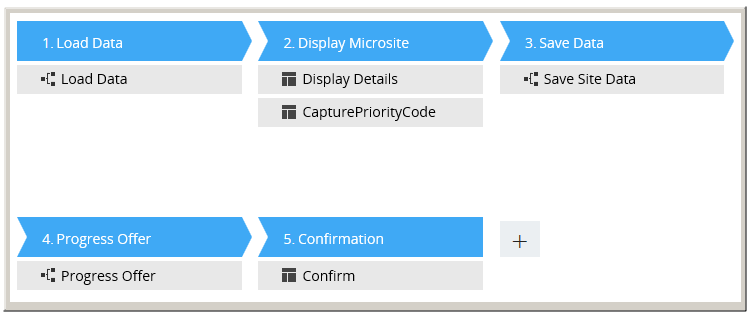
After entering an email and clicking **Next**, the confirmation page is displayed:

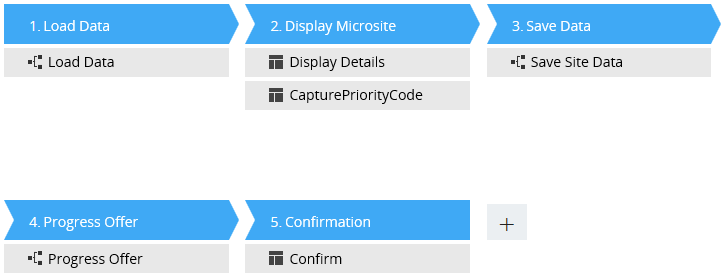


Configuring a microsite to be aware of customer and action context

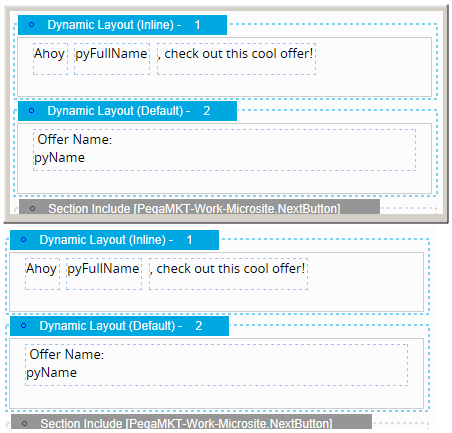
The following Microsite serves as a basic example of utilizing the customer and action context.

Pega Customer Decision Hub



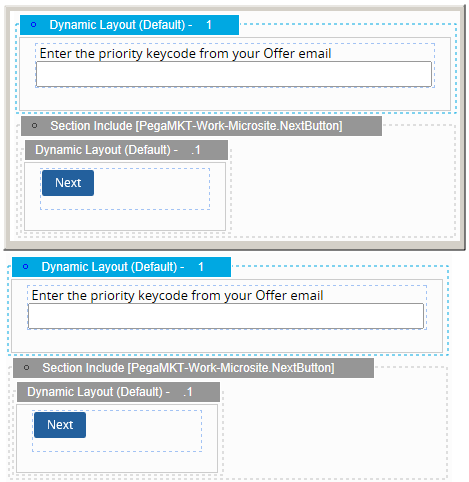


This Microsite first loads relevant action and customer data. At runtime, the action context is determined from the URL used to launch the Microsite. This context is then used to load the action and customer data. The Microsite then uses this data to display a personalized page to the customer with pertinent customer and action information.

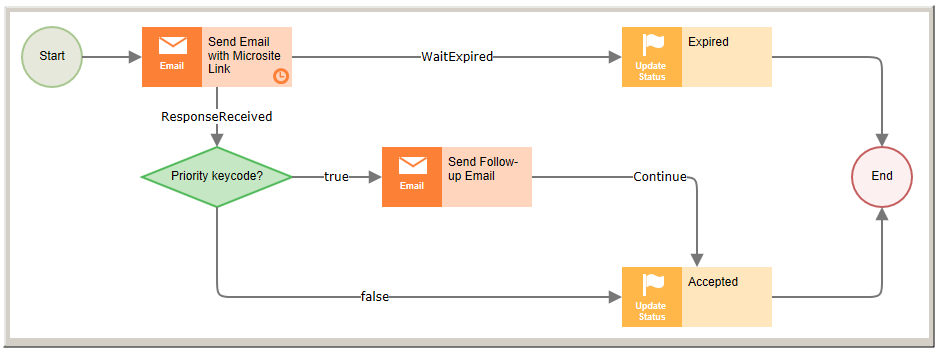


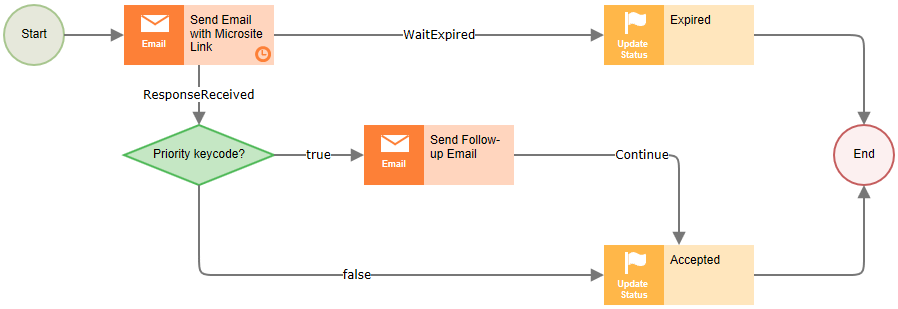
In the above section, customer data is referenced via the .Customer page property, while action data is referenced via the .Offer.OfferData page property.

On the next page, the customer is prompted for their priority code (an action data field).



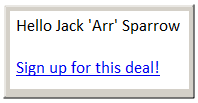
Once the customer enters this information, the **Save Site Data** step saves the action data value (since the Microsite is action-aware). The **Progress Offer** step resumes the action along the ResponseReceived path with an **Accepted** response. The action can utilize the saved action data value (Priority Code, in this example) and use it to branch the flow, as shown below:





In the above action, the initial email shape sends out a link to the Microsite. If the customer responds to this action (e.g. via the Microsite), the flow uses the priority code (entered via the Microsite) to branch flow execution.

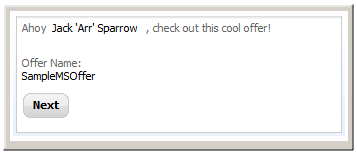
At runtime, when the above action is initiated, the customer gets an email with a link for the Microsite.

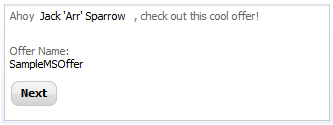


Hello Jack 'Arr' Sparrow

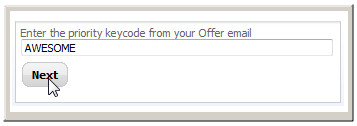
Sign up for this deal!

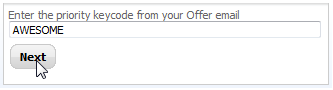
Upon clicking the email link, the Microsite is launched with an action and customer context. The Microsite loads the necessary data from these contexts and displays the first page.



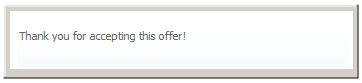


Upon clicking **Next**, the customer enters their priority code on the next screen.





Upon clicking **Next** again, the priority code is saved (as action data) and the action is progressed. The action uses the entered code value to branch flow execution. Finally, the confirmation page is displayed to the customer.



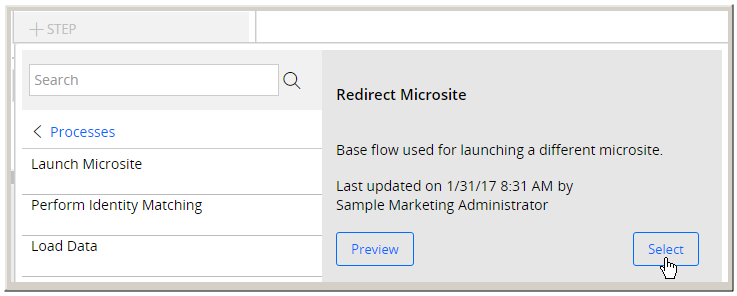
Thank you for accepting this offer!

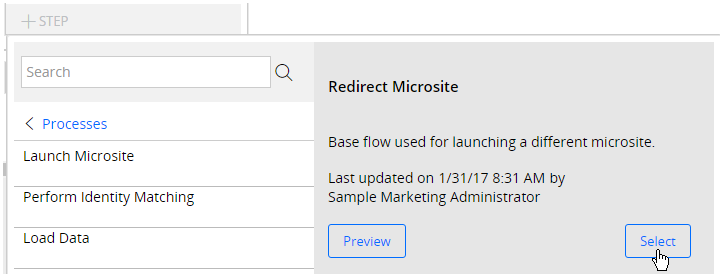
Configuring a microsite to launch a different microsite

Sometimes, users may wish to launch a different Microsite at a particular stage in their Microsite. This can be done by using either the **Redirect Microsite** or the **Launch Microsite** step, as outlined below.

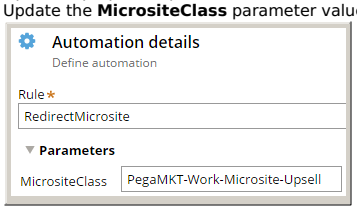
Pega Customer Decision Hub

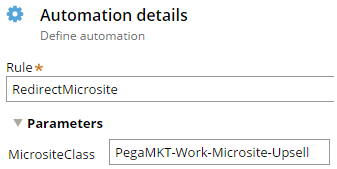
1. Add a new step or determine the existing step to replace. In this example, a step has been added to the confirmation stage to redirect to another Microsite.
2. Select **More Processes Redirect Microsite** and then click **Select**.





3. Save the Microsite configuration.
4. Select the newly added step and click the **Open process** link in the right pane. This will open the RedirectMicrosite Flow rule.
5. Create a copy of this Flow into your Microsite class.
6. Check out the specialized Flow.
7. Open the properties panel for the RedirectMicrosite utility shape.
8. Update the **MicrositeClass** parameter value to the class of the Microsite to redirect the customer to.





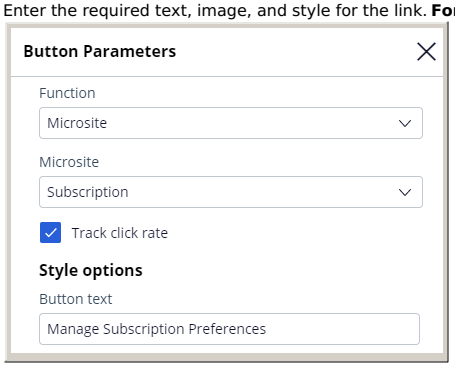
9. Save the Flow and check it back in (if applicable).

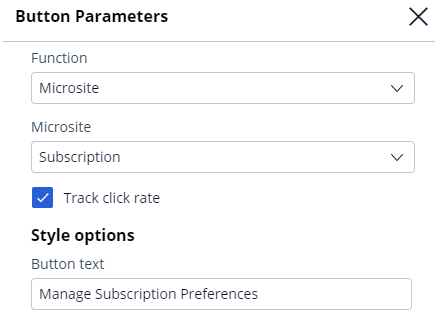
Embedding a microsite link in an email treatment

You can configure email treatments to provide links to a microsite. The link contains the customer and action context, so that the microsite is customer- and action-aware. Microsites can load customer and action data, display it to the user, save action data if applicable, and progress the action.

Pega Customer Decision Hub
To add a microsite link to an email treatment, follow the steps below:

1. In an email treatment, click **Insert Button**.
2. Select **Microsite** as the button function.
3. Select the microsite from the **Microsite** list.
4. Enter the required text, image, and style for the link. **For example:**





Launching a microsite

Microsite can be launched and tested via the following URL: http://<server>:<port>/MS/index.html?Site=<MicrositeName>.

Pega Customer Decision Hub

The above URL pattern can only be used for testing and launching Microsite once the Microsite war (provided as part of Pega Customer Decision Hub) has been installed and configured by the application administrator. Administrators should refer to the installation guide for more details.

Important: You must have browser cookies enabled to view a Microsite.

Identifying customers who respond to a microsite

Identity matching also allows marketers to track unidentified customers and their interactions with Microsites. When an anonymous individual visits a Microsite, the system identifies the individual with a unique identifier. Later, when customer information is available to the Microsite, the system associates the anonymous user's interactions with the customer.

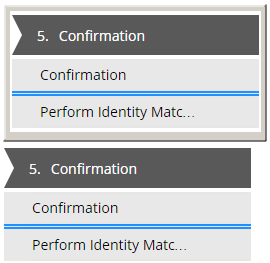
Pega Customer Decision Hub

The following associations are captured during interactions with the Microsite:

- CustomerID to UniqueID
- CustomerID to Email Address
- UniqueID to Email Address

Identity matching in Microsites is handled by the following stages in the lifecycle:

1. **Load Data** - In addition to loading customer data, this stage also saves the aforementioned association data. For example, if the browser cookie contains the unique id but the Microsite is aware of the customer, this stage associates the unique id and the customer id. Additionally, the cookie is updated with customer information.
2. **Confirmation** - In this stage, the **Perform Identity Matching** step associates available Microsite attributes. For example, if the Microsite collects email information, this information is associated with the available identifier (customer id or unique id).



Note: To perform identity matching during a different stage, include the **Perform Identity Matching** step in that stage.

In addition to the above stages, identity matching also occurs in the following areas in Microsites:

- **Impression capture** - When the section RecordImpressionForMicrosite is included in a Microsite stage to capture impressions, it calls the IMPR service. For anonymous users, this service records the impression using the unique id.
- **Launching another Microsite** - When a Microsite launches (or redirects to) another Microsite, identity matching attributes - such as customer id or unique id - are propagated to the new Microsite.

Note: If the Perform Harness rule is specialized in the new Microsite class, ensure that the IdentityMatchingConfiguration Section rule is included in this Harness. This section enables the Internet Application Composer (IAC) gadget to update the browser cookie.

Managing microsite subscriptions

Pega Customer Decision Hub provides an out-of-the-box sample implementation of a Microsite that manages a customer's subscription preferences. This is labeled the Subscription Microsite and its implementation class is PegaMKT-Work-Microsite-Subscription.

Pega Customer Decision Hub

This Microsite is associated as a Case Type on the shipped sample application. To use this Microsite with a different application, administrators must add it as a Case Type to the desired application.

The shipped implementation provides a basic infrastructure for supporting customer unsubscribes and management of subscription preferences. As such, custom implementations are expected to extend the provided functionality and integrate the saved preference data into their marketing processes (such as handling customer unsubscribes).

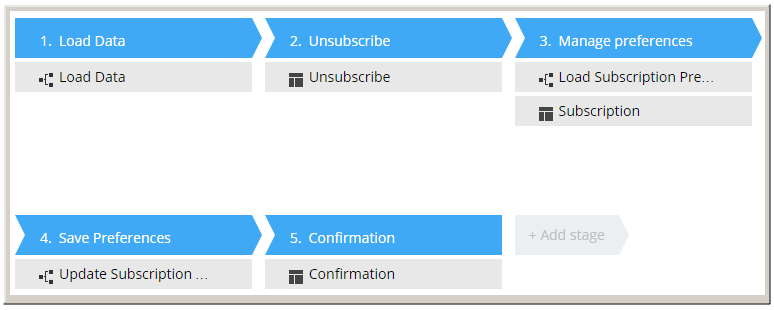
- Configuring a subscription microsite

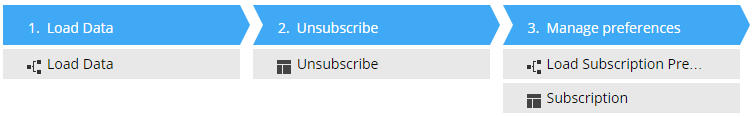
Configuring a subscription microsite

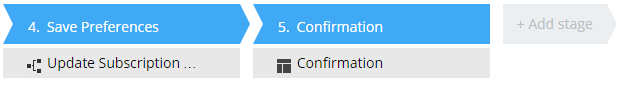
The Subscription Microsite is backed by the PegaMKT-Work-Microsite-Subscription class.

Pega Customer Decision Hub

This Microsite has the following primary stage flow:



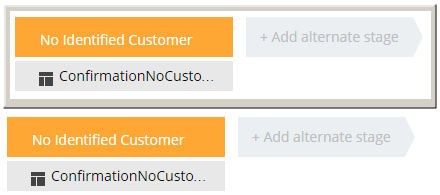




This includes the following stages:

1. **Load Data** – This stage uses the LoadSupportingData Flow to load data for the referenced customer. If a customer context cannot be established, this flow switches to the **No Identified Customer** alternative stage.
2. **Unsubscribe** – This stage renders the primary unsubscribe screen.
3. **Manage Preferences** – This stage loads the customer's previous preferences and renders the manage preferences screen.
4. **Save Preferences** – This stage utilizes the UpdateSubscriptionPreferences Flow rule to save any changes to the customer's subscription preferences. This Flow also handles the cancel path, when the customer decides to discard their changes.
5. **Confirmation** – This stage renders the confirmation screen.

The Subscription Microsite also includes the following alternate stage flow:



This includes the following alternate stage:

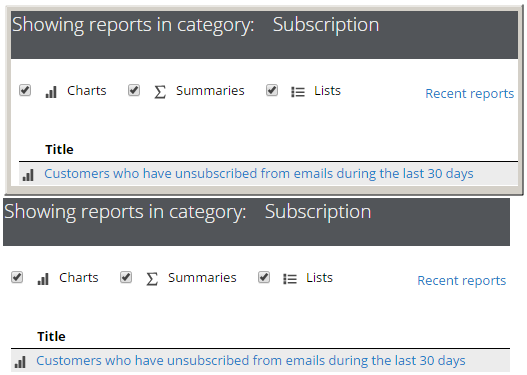
1. **No Identified Customer** – This stage renders the **Customer Not Found** screen.

- Understanding unsubscribe data with the Unsubscribed Customers report

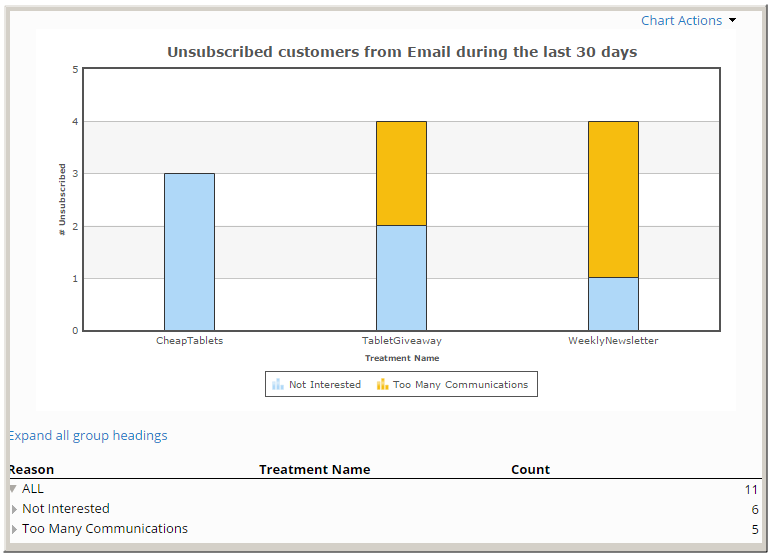
Understanding unsubscribe data with the Unsubscribed Customers report

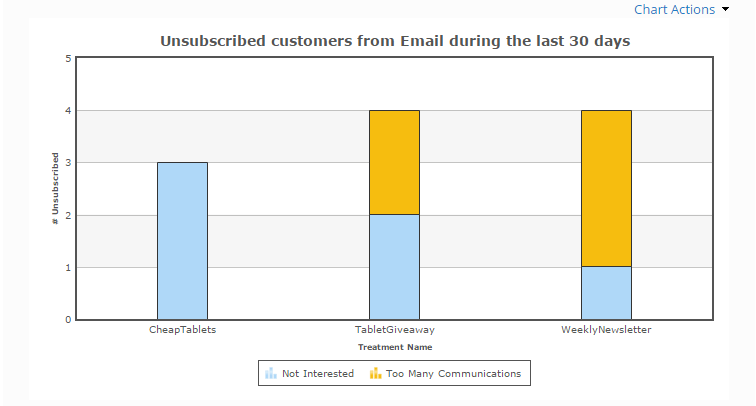
The shipped application also includes a sample Report Definition that allows users to review details on customer unsubscribes for the email channel. This report is available in the Report Browser under the Subscription category.

Pega Customer Decision Hub



Clicking the above link launches the report, which displays a chart for the email unsubscribes in the last 30 days. The chart groups unsubscribes by Treatment Name. Within each group (bar), there is a further classification based on the unsubscribe reason.





Expand all group headings

Reason	Treatment Name	Count
ALL		11
Not Interested		6
Too Many Communications		5

In addition to the chart, the report also displays an expandable breakdown of unsubscribes by reason:

Reason	Treatment Name	Count
ALL		11
Not Interested		6
	CheapTablets	3
	TabletGiveaway	2
	WeeklyNewsletter	1
Too Many Communications		5
	WeeklyNewsletter	3
	TabletGiveaway	2

Reason	Treatment Name	Count
ALL		11
Not Interested		6
	CheapTablets	3
	TabletGiveaway	2
	WeeklyNewsletter	1
Too Many Communications		5
	WeeklyNewsletter	3
	TabletGiveaway	2

Clicking on a region in the chart (or an expanded row in the reason breakdown) switches the display to a list. This list includes relevant details that were captured as part of the unsubscribe request, such as the Issue, Group, and action name.

Customers who have unsubscribed from emails during the last 30 days > Too Many Communications•TabletGiveaway

Filters: Reason = Too Many Communications AND Treatment = TabletGiveaway AND (Last 30 Days AND Email channel AND Unsubscribed)

Displaying 2 records

Customer	Issue	Group	Offer	Last Updated On	
Reason: Too Many Communications					2
C00001316	Sales	Tablets	2013Tablets	11/30/14 7:01 PM	
C00001318	Sales	Tablets	2013Tablets	11/30/14 7:01 PM	

Customers who have unsubscribed from emails during the last 30 days > Too Many Communications•TabletGiveaway

Filters: Reason = Too Many Communications AND Treatment = TabletGiveaway AND (Last 30 Days AND Email channel AND Unsubscribed)

Displaying 2 records

Customer	Issue	Group	Offer	Last Updated On	
Reason: Too Many Communications					2
C00001316	Sales	Tablets	2013Tablets	11/30/14 7:01 PM	
C00001318	Sales	Tablets	2013Tablets	11/30/14 7:01 PM	

Extending the capabilities of next best action with customized strategies

In Pega Customer Decision Hub, strategies are the underlying mechanism that determines which action are selected for which customers.

Pega Customer Decision Hub
Important:

Before the introduction of Next-Best-Action Designer and the Next-Best-Action strategy framework, strategies were configured individually by using the Strategy Builder, or by manually adding shapes to a strategy canvas. In newer versions of the product, this functionality has been superseded by the practices of 1:1 Customer Engagement, embodied within the product through the use of Next-Best-Action Designer. As a best practice, you should configure your customer engagement strategy in Next-Best-Action Designer, as described in Improving customer experience with next-best-action on owned channels. However, for use cases such as configuring the various extension points available in the Next-Best-Action strategy framework, you may need to create additional strategies.

For more information about the use cases where custom strategies are required, see Creating and understanding decision strategies on Pega Academy.

- Understanding globally optimized strategies
- Creating strategies
- Configuring Pega Customer Decision Hub strategies
- Configuring ethical bias checks

Understanding globally optimized strategies

Introduced in Pega Customer Decision Hub version 8.6, globally optimized strategies (GOS) are a new type of decision strategies that focus on providing the best performance at runtime. Strategies created by using Next-Best-Action Designer support GOS out of the box, but you can also configure other strategies to benefit from the performance

improvements that GOS provides.

Pega Customer Decision Hub

Globally optimized strategies are defined as top-level decision strategies in a data flow, that is, the first strategy that the data flow calls during the decisioning process.**Caution:** Do not reference globally optimized strategies from legacy data flow components such as Run Strategy, or legacy rule types such as Interaction rules. GOS does not support legacy ways of invoking strategies, so referencing GOS from an unsupported component may result in issues. Instead, to use globally optimized strategies, edit the data flow to include a Strategy component.

When a globally optimized strategy runs, it brings several types of decisioning artifacts into a single program and executes them all together. Because rules are executed together rather than called one by one, overall strategy execution performance is greatly increased.

Use GOS when you want to improve the performance of your decisioning strategies that are referenced from a data flow.

Globally optimized strategies cannot be used as sub-strategies, in proposition filter conditions, or as target for the **Check decision strategies for optimizations** upgrade tool.

Note: Globally optimized strategies can be used in simulations that support them, such as distribution test, value finder, and ethical bias simulations. In the current version of Pega Platform, decision funnel simulations do not support GOS. If you use a globally optimized strategy in a decision funnel simulation, it will instead run the strategy that the GOS references.

Supported rule types

GOS can import and execute the following rule types:

- Proposition Filter rules
- When rules, if used in a proposition filter
- Strategies, if used as a sub-strategy on the strategy canvas, or in a proposition filter
- Decision tables, if supported by SSA.

Supported strategy components

Strategy optimization is done on the component level. The following strategy components can be optimized:

- External sub-strategy
- Embedded sub-strategy
- Decision table
- Data join
- Set property
- Filter (including expression and proposition filters)
- Prioritize
- Champion-challenger
- Exclusion
- Switch
- Group by

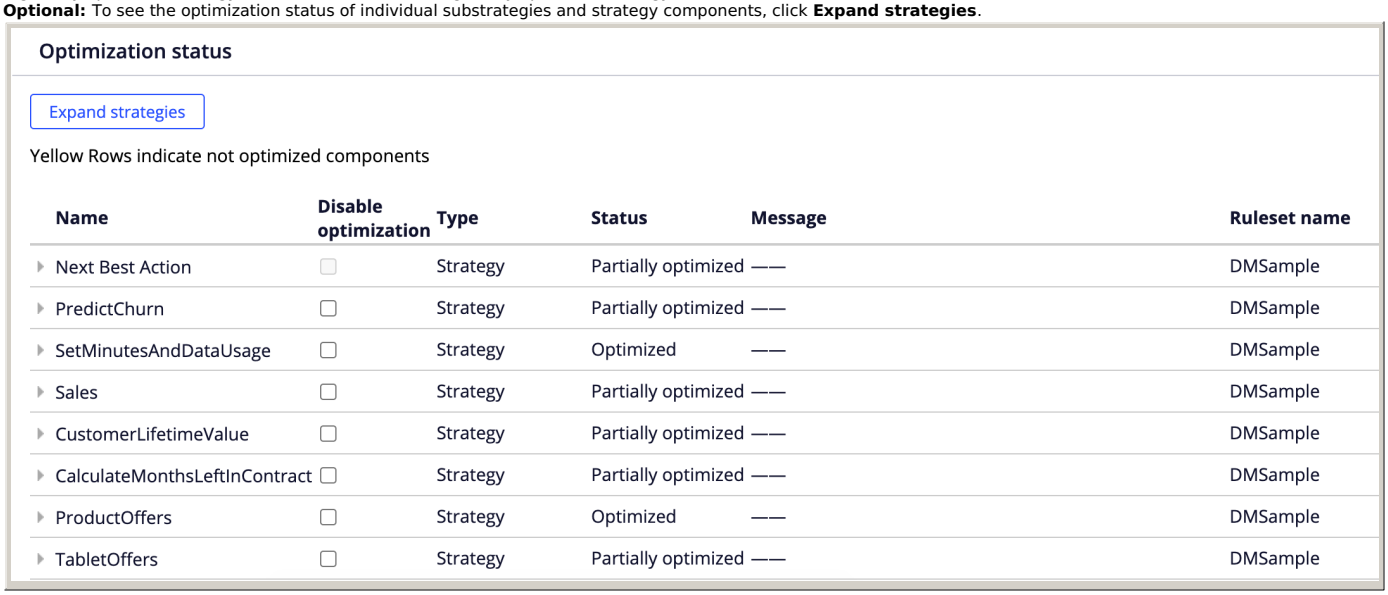
- Creating globally optimized strategies
- Frequently asked questions about globally optimized strategies

Creating globally optimized strategies

Create a globally optimized strategy by referencing an existing strategy that you want to optimize.

Pega Customer Decision Hub
Note: If you use the Next-Best-Action Designer component of Pega Customer Decision Hub, you do not need to manually optimize the resulting strategy framework. For more information, see Understanding the Next-Best-Action strategy framework. However, you can still choose to optimize strategies created manually outside of the Next-Best-Action strategy framework.

1. Log in to your application as an operator with access to Dev Studio.
2. In Dev Studio, in the left navigation pane, click **Records Decision Strategy** .
3. Click **Create** .
4. On the **Create Strategy** form, select **Create globally optimized strategy**.
5. Enter a label for the strategy, and then click **Create and open**.
6. On the **Global optimization** tab, select a top-level strategy that you want to optimize.
   A globally optimized strategy cannot reference another globally optimized strategy.
7. **Optional:** To see the optimization status of individual substrategies and strategy components, click **Expand strategies**.



Optimization status

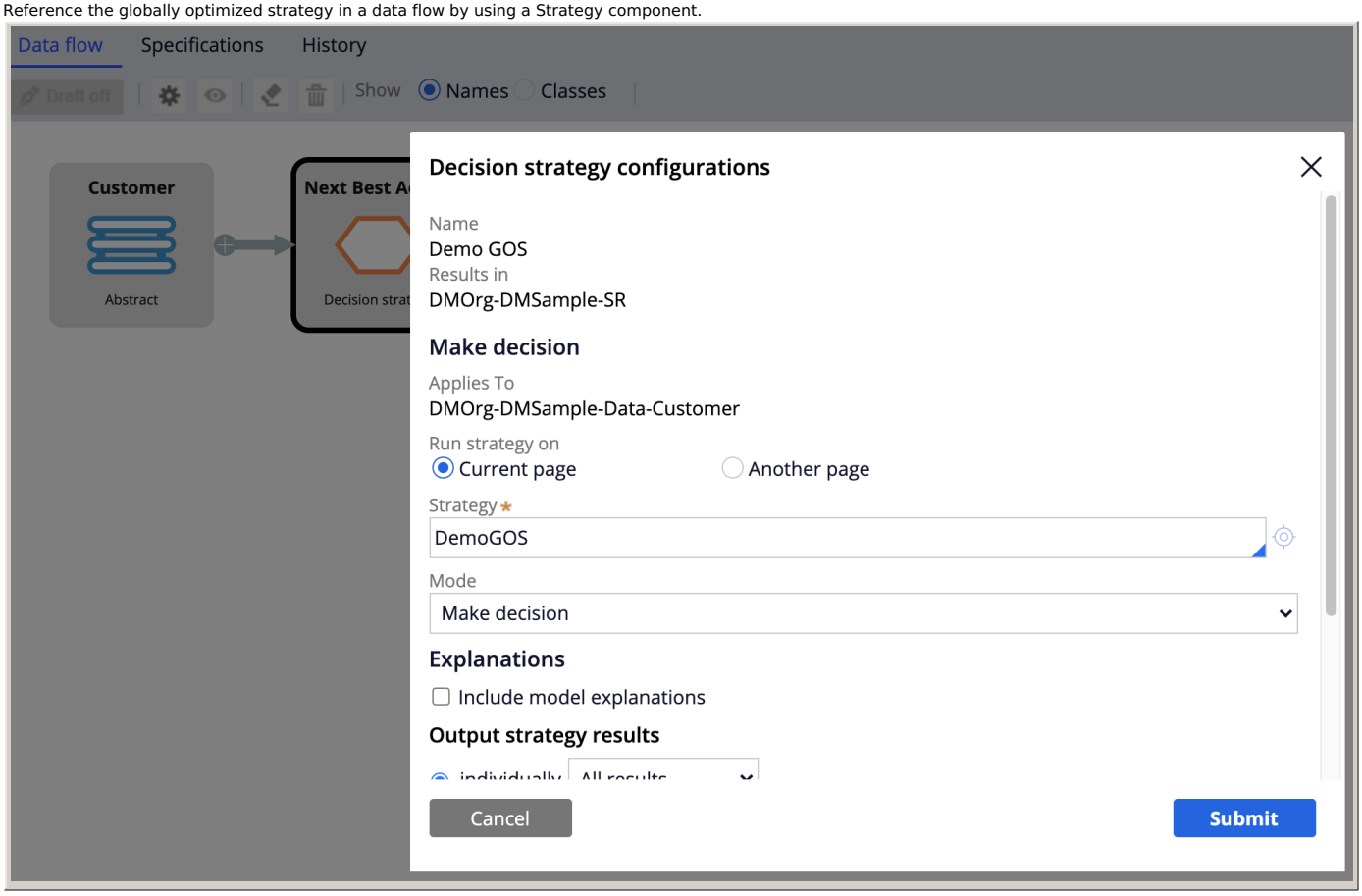
Expand strategies

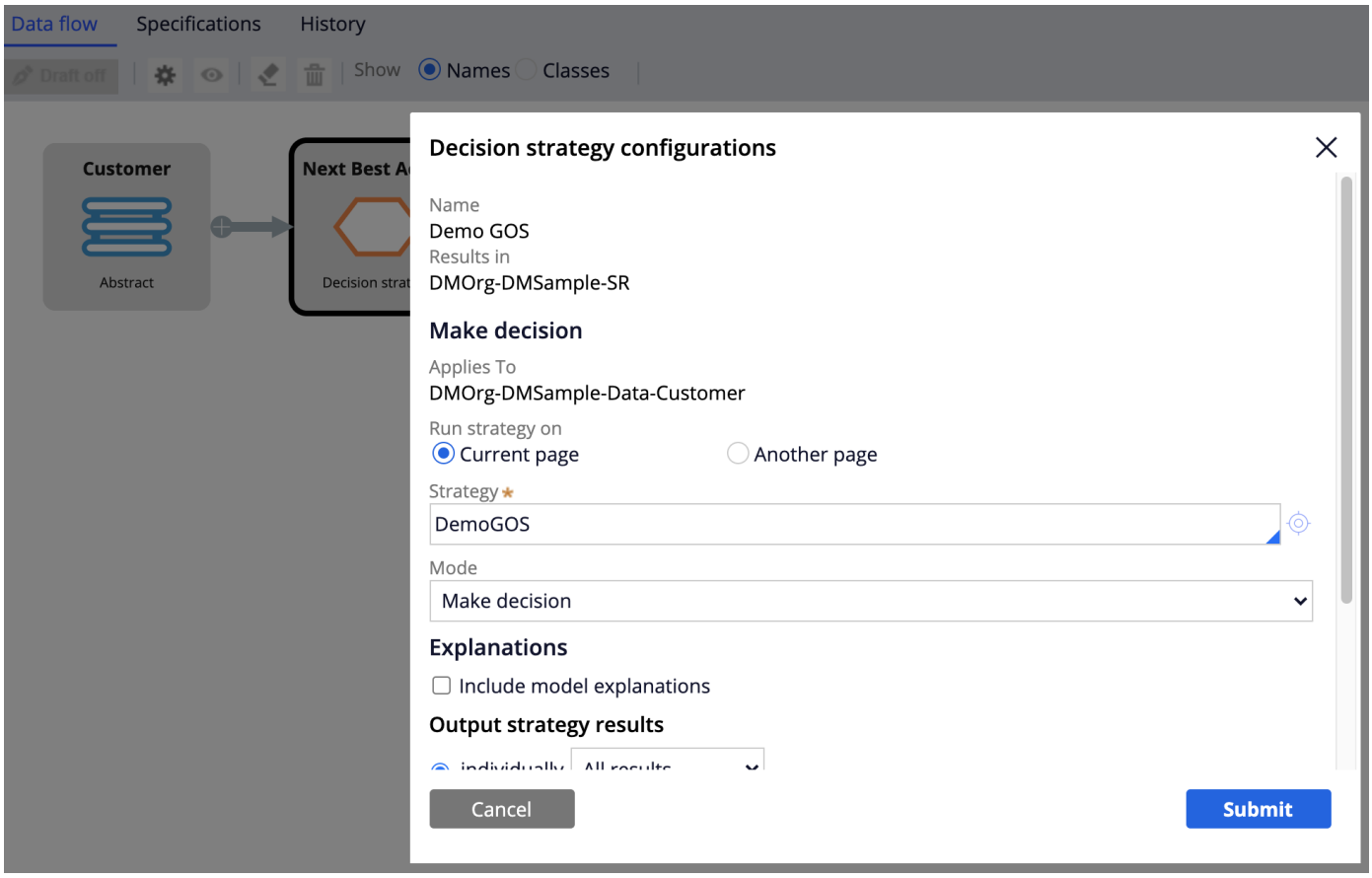
Yellow Rows indicate not optimized components

Name	Disable optimization	Type	Status	Message	Ruleset name
Next Best Action		Strategy	Partially optimized	——	DMSample
PredictChurn		Strategy	Partially optimized	——	DMSample
SetMinutesAndDataUsage		Strategy	Optimized	——	DMSample
Sales		Strategy	Partially optimized	——	DMSample
CustomerLifetimeValue		Strategy	Partially optimized	——	DMSample
CalculateMonthsLeftInContract		Strategy	Partially optimized	——	DMSample
ProductOffers		Strategy	Optimized	——	DMSample
TabletOffers		Strategy	Partially optimized	——	DMSample

Note: A strategy may have Partially optimized or Not optimized status when its components contain expressions with unsupported functions. To learn how to optimize such strategies, see Optimizing strategies with allow list functions.

8. Click **Save**.
9. **Optional:** In the **Test** tab, compare the results of the optimized strategy to the normal version by clicking **Compare**.
   Note: Most fields should have exactly the same value for GOS execution and normal strategy execution, except some special fields, such as Interaction ID, Decision Reference, or Decision Time.
10. Reference the globally optimized strategy in a data flow by using a Strategy component.





Caution: Do not reference globally optimized strategies from legacy data flow components such as Run Strategy, or legacy rule types such as Interaction rules. GOS does not support legacy ways of invoking strategies, so referencing GOS from an unsupported component may result in issues. Instead, to use globally optimized strategies, edit the data flow to include a Strategy component.

Frequently asked questions about globally optimized strategies

See the following topics for additional information, tips and tricks, and best practices.

Pega Customer Decision Hub

What happens if I update a rule that GOS references?

By default, if you update any of the filters, When rules, strategies, or decision tables that the globally optimized strategy imports and executes, GOS will automatically recompile after a short period of time.

Can I manually recompile GOS without waiting for it to recompile automatically?

It is possible to switch off automatic recompilation by editing the *Pega-DecisionEngine.decision/strategy/cacheGOSProgram* dynamic system setting. By default, the setting is set to SMART_CACHE. To enable manual recompilation of globally optimized strategies, change the setting value to STATIC_CACHE. You can manually recompile any GOS on the **Configure Decisioning Decisions Strategy optimizations** page in Dev Studio.

Caution: Only disable automatic recompilation on environments where rules are not frequently updated and performance is important, for example, on production environments.

How do I manage changes to GOS rules?

As a best practice, use the Pega change management tool, Pega 1:1 Operations Manager, to manage business changes. GOS natively supports Pega 1:1 Operations Manager's standard business change management process. GOS rules are automatically included in relevant revision packages.

If you do not use Pega change management tools, you must propagate the changes manually, and then recompile GOS rules on the destination environment.

What can I do when my GOS cannot be optimized?

GOS cannot be optimized when their components contain expressions with unsupported functions. However, you can optimize a strategy by adding these functions to the *pyAllowlistFunction* data transform or by changing the strategy logic.

For more details, see Optimizing strategies with allow list functions.

Creating strategies

You can create new Strategies from the Strategies landing page in the Pega Customer Decision Hub portal.

Pega Customer Decision Hub

1. Click **Intelligence Strategies** .
2. Click **Create**.
3. Select the Strategy creation mode:
   - If you want Pega Customer Decision Hub to guide you through the Strategy creation, select **Guide me through it**. For more information, see Using the Strategy Builder. The Strategy Builder allows you to create Strategies for the primary customer context in your application.**Note:** Use this option for retail banking strategies in Pega Customer Decision Hub.
   - If you want to create the Strategy manually, select **Start with new canvas**. For more information, see Creating Strategies Manually. Manual Strategy creation allows you to create Strategies for all customer contexts in your application.

- Using the Strategy Builder

- Creating strategies manually

Using the Strategy Builder

The Strategy Builder walks you through the process of designing your marketing Strategies.

Pega Customer Decision Hub

1. To create a new Strategy with the Strategy Builder, click **Create Guide me through it** on the Strategies landing page.
2. Enter the Strategy name.
   The system automatically generates an identifier for the Strategy based on the specified name. To change the identifier, click the **Edit** icon.
3. Enter the Strategy description.
4. In the **Purpose** section, click **Configure**.
5. For Pega Customer Decision Hub strategies, in the **Configure Objective** window, select a Strategy objective by clicking **Add**, and then click **Apply**.
   For more information about using this window, see the Using a Configuration Modal Window Using a Configuration Modal Window appendix. A new configuration section for the selected objective appears under the **Purpose** section. The contents of the configuration section depend on the objective which you selected.
6. For retail banking strategies in Pega Customer Decision Hub, in the **Configure Objective** window, select **Calculate Customer Lifetime Value** by clicking **Add**, and then **Apply**.
   For more information about using this window, see the Using a Configuration Modal Window appendix. A new configuration section for the selected objective appears under the **Purpose** section. The contents of the configuration section depend on the objective which you selected.
7. Configure the selected objective. For more information, see Configuring the Objective.
8. Click **Save**. Any configuration errors are displayed on top of the page. You must resolve them before the Strategy is generated.
9. Click **Actions Open canvas** to review the generated Strategy.
10. **Optional:** Manually configure the Strategy.
    - For general information about Strategy configuration in Pega Platform, see About Strategy rules.
    - For information about Strategy configuration specific to Pega Customer Decision Hub, see Configuring Pega Customer Decision Hub Strategies.
11. **Optional:** To delete the Strategy Builder configuration for this Strategy, click **Delete**.
    This does not remove the generated Strategy, only the ability to edit it in the Strategy Builder.
    Note: You can also use the Strategy Builder to edit existing Strategies. To access the Strategy Builder for an existing Strategy, find the Strategy on the Strategies landing page and click **Actions Open Strategy Builder** . If you make any manual changes to your Strategy, and then edit the Strategy using the Strategy Builder, your manual changes will be overwritten.

- Configuring the objective

Configuring the objective

An objective is a business goal that the Strategy should achieve. You must configure one objective for each Strategy.

Pega Customer Decision Hub

The following objectives are available:

- Calculate Customer Lifetime Value
- Calculate Customer Lifetime Value for retail banking
- Fixed or dynamic action bundle
- Action prioritization strategy
- Action targeting using Audiences and AI
- Action Targeting using paid optimization

- Configuring the Calculate Customer Lifetime Value objective
- Configuring the Calculate Customer Lifetime Value objective for retail banking
- Configuring the Fixed or Dynamic Action Bundle objective
- Configuring the Action Prioritization Strategy objective
- Configuring the action targeting using the Audiences and AI objective
- Configuring the Paid Optimization objective

Configuring the Calculate Customer Lifetime Value objective

The Calculate Customer Lifetime Value objective allows you to define how customer lifetime value (CLV) is calculated in Pega Customer Decision Hub.

Customer lifetime value (CLV)
: Customer lifetime value is an estimation of the net profit predicted for the entire future relationship with a customer.

Pega Customer Decision Hub
Before you begin: See Creating strategies.

1. In the **Method** section, click **Configure**.
2. In the **Configure Calculation Method for CLV** dialog box, select a calculation method by clicking **Add**. **Result:** Available calculation methods and expressions used by each calculation method are displayed. **Basic Calculation** is the default method.
3. Click **Apply**. **Result:** Additional configuration settings for the selected calculation method appear. For the default Basic Calculation method, the following settings are available:

   Acquisition cost
   : The cost of acquiring the customer.

   Annual profit
   : The yearly profit from the customer, excluding acquisition costs.

   Retention rate
   : The annual rate of retention of your customer base.

4. For the default Basic Calculation method, enter the following constant values: **For example:**
   - **Acquisition cost** = 50
   - **Annual profit** = 1400
   - **Retention rate** = 80

   The Basic Calculation method applies the following calculation expression for the Customer Lifetime Value:

   (Annual profit * YearsACustomer) – Acquisition cost

   where YearsACustomer = 1/(1 - Retention rate as a decimal value).

   So, the calculated Customer Lifetime Value in this example is:

   {1400 * [1/(1-0.8)]} - 50 = [1400 * (1/0.2)] - 50 = (1400 * 5) - 50 = 6950

5. Configure the values for the available fields. You can select the following types of values for each field:
   - **Constant** – Enter a number which represents the value.
   - **Property** – Select a property from the customer class that contains the value.
   - **Strategy** – Configure a Strategy that returns this value. After selecting a Strategy, select the strategy result property that contains the value.
6. Click **Save**.

Configuring the Calculate Customer Lifetime Value objective for retail banking

The Calculate Customer Lifetime Value objective allows you to define how customer lifetime value (CLV) is calculated in Pega Customer Decision Hub specifically for retail banking.

Customer lifetime value (CLV)

The CLV is the discounted value of the future profits that will be generated by an individual customer, discounted over a weighted average cost of capital. For example, a customer that will generate $120, $80, $30, $50, and $10 of profits in the next 5 years will have a CLV of $238, if the discounted factor is 10%.

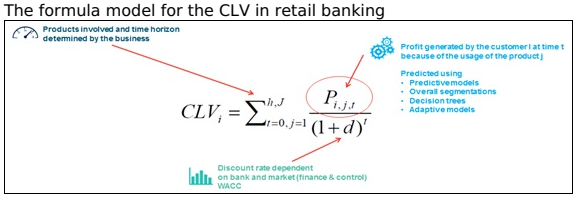
Pega Customer Decision Hub

In retail banking, CLV helps your organization reach strategic objectives, such as increasing share of wallet, improving customer retention, and increasing return on marketing investment.

By activating customer valuation models, you can place more emphasis on customer service and long-term retention, rather than on maximizing short-term sales. As these future profits are uncertain, models have to be developed to estimate the future value of customers and based on data analysis techniques instead of the traditional analysis of historical data.

To deliver a CLV in a retail banking context, use the Pega model for customer valuation based on research by Haenlein, Kaplan & Beeser, as shown in the following figure:

The formula model for the CLV in retail banking



where: *CLV* is a function of the profit *Pi,j,t.t* will be generated in the future through *j.j* is the product. *Pi,j,t* is unknown because the prediction model is built to derive this future profit based on overall segmentations for the next five years by using transition values.For the identified segment, use percentage *CLV* value 100 and calculate the change in *CLV* for the next 5 years using transition values across each segment. *WACC* is the Weighted Average Cost of Capital that is the discounted rate of returns expected by the bank.

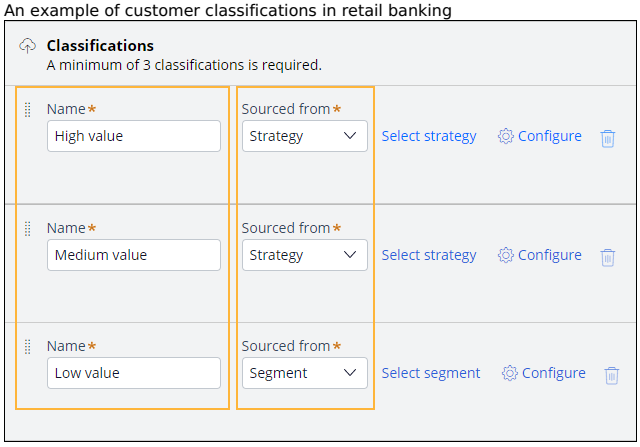
Calculate the CLV for retail banking by performing the following steps:

1. In the **Customer Lifetime Value method** section, click **Configure**.
2. In the **Configure Calculation Method** dialog box, select **Retail Banking** by clicking **Add**, and then **Apply**. **Result:** Additional configuration settings for the selected calculation method appear.
3. In the **Discount rate** field, enter, for example, 10.
   WACC is the recommended discount rate.
4. In the **Classifications** section, add a minimum of three classifications by clicking **Add classification**:

Classifications

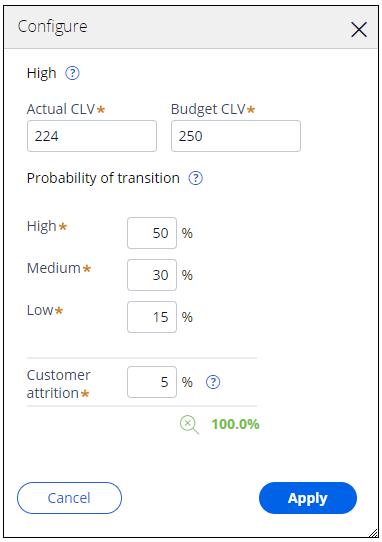
Classifications are groupings of your customers according to a criterion. For example, you may group your customers as high value, medium value, and low value clients. The CLV calculation for retail banking supports a maximum of 20 classifications. Each classification can be represented as either a (sub)strategy or a segment. Classifications require the appropriate strategy rules to be created in advance. For more information on creating a strategy rule, see Using the Strategy Builder. For information about creating a segment, see Defining customer audiences with customer segments.

An example of customer classifications in retail banking



   a. In the **Name** field, enter a name of your classification, for example, High Value.
   b. In the **Sourced from** drop-down list, choose if you want each classification to be represented as a strategy or a segment.
5. **Optional:** In the **Classifications** section, to choose a specific strategy or a segment for your classification, click **Select strategy** or **Select segment**:
   - If you click **Configure Strategy**, in the **Configure Strategy** dialog box, select your strategy by clicking **Add**, and then **Apply**.
   - If you click **Configure Segment**, in the **Configure Segment** dialog box, select your segment by clicking **Add**, and then **Apply**.

   You can also create a strategy or a segment by clicking **Create** in the **Configure Strategy** or **Configure Segment** dialog box. For more information, see Creating strategies or Creating a Criteria Segment.
6. In the **Classifications** section, configure each classification by clicking **Configure**.
7. In the **Configure** dialog box, fill in the required fields, as shown in the following example, and then click **Apply**. **For example:**



where:*Actual CLV* is the current CLV assigned to each classification.*Budget CLV* is the amount that you are willing to invest into the customer.*Probability of transition* is a chance that the customer from each classification transitions from their current classification to each of the other classifications.*Customer attrition* is the chance of losing the customer due to customer attrition.
Note: The total of the customer attrition rate with all the percentages of probability of transition should add up to 100%.

8. On the top of the **Create a Strategy** page, click **Save**.

Result: A strategy rule with your defined classifications and transition probabilities is generated. This strategy then calls the RetailCustomerLifeTimeValue strategy to run the final mathematical calculation before determining the CLV score of the customer.**What to do next:** You can modify your strategy rule only in the strategy rule canvas by searching for your strategy in the **Strategies** landing page of the Pega Customer Decision Hub portal, and then clicking **Actions Open Strategy Builder** .**Note:** Strategies that were not created through the Strategy Builder initially or have been directly edited on the canvas do not have the option to **Open Strategy Builder** available.

Configuring the Fixed or Dynamic Action Bundle objective

The Fixed or dynamic action bundle objective creates an action bundle with a Bundle Parent and one or more Bundle Members for each targeting approach. You can use audiences or analytics to determine the best mix of Bundle Members.

Pega Customer Decision Hub

1. Select the business issue that is addressed by this objective.
2. Select one or more business groups from the selected business issue. The business groups will be used in the generated strategy. After you select the business groups, the Targeting approach section appears.
3. In the **Targeting approach** section, click **+Add targeting approach**, and then select one of the following options:
   - **Audience driven** - Select this approach if you have an audience that you want to target with a specific action. An audience is made up of a starting segment, and can then be split into sub-segments.
   - **Analytics driven** - Select this approach if you want to use predictive or adaptive analytics to score and rank actions. Using analytics for targeting allows you to consider a larger set of actions.
4. Click **Apply**.
5. Configure the selected targeting approach.
   - For audience driven approaches, see Configuring Audience Driven Targeting.
   - For analytics driven approaches, see Configuring Analytics Driven Targeting.
6. Configure the Bundle Parent.
   a. In the **Bundle** section, click **Configure**.
   b. Select an action that you want to use as Bundle Parent. For more information about action bundles, see Grouping offerings into action bundles.
   Note: Only actions with the attribute Bundle Parent appear in this section. For more information, see Using the Details Tab on the Action Rule Form.
   c. Click **Apply**.
   d. **Optional:** Edit the name and type of the bundle. Any changes to the name and type of the bundle are only applied in the generated Strategy, not in the original action used as Bundle Parent.
7. **Optional:** Repeat steps 3-6 to configure more targeting approaches.
8. **Optional:** If you added multiple targeting approaches, specify how you want to arbitrate between them. For more information, see Arbitrating over Targeting Approaches.
9. **Optional:** If you added multiple targeting approaches, use the **Overall prioritization** section to specify how you want to prioritize across your targeting selections. This prioritization determines the order in which actions are made to a customer. For more information about configuring prioritization, see Configuring Overall Prioritization.
10. Click **Save**.

- Configuring audience driven targeting
- Configuring analytics driven targeting
- Arbitrating over targeting approaches
- Configuring overall prioritization

Configuring audience driven targeting

To configure audience driven targeting, select one or more Segments to represent the starting audience targets, then configure how each Segment should be used by the Strategy that you are creating.

Pega Customer Decision Hub

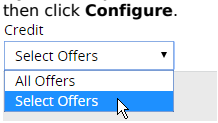
1. In the **Targeting approach Audience** section, click **Configure**.
2. In the Configure Audience window, click a Segment which you want to use as audience for this targeting approach, then click **Apply**. For more information about the window, see the Using a Configuration Modal Window appendix.
3. Click **Assign actions**.
4. In the Configure actions window, click the action or actions that you want to apply to customer entities in the selected Segment, and then click **Apply**. For more information about the window, see the Using a Configuration Modal Window appendix.
Note: The list only shows actions from the same business groups that you selected for the objective. For more information, see Configuring the Fixed or Dynamic Action Bundle objective or Configuring the Paid Optimization objective.
5. **Optional:** Assign different actions to customer entities from your starting Segment, based on other Segments that the customer entities also belong to. For example, you can start with a large Segment called Strategy Audience , split that Segment into Business Customers and Individuals , and then assign different actions to each under the same Strategy.
   a. Click **Actions Split by segment** by the name of a Segment that you selected in step 2. The **Configure Audience** pop-up window appears. For more information about the window, see the Using a Configuration Modal Window appendix.
   b. Select one or more Segments and then click **Apply** . The selected Segments appear as new rows under the Segment that you selected in step 2.
   c. For each of the selected Segments, click **Assign actions** , select the action or actions that you want to apply to customer entities in the selected Segment, and then click **Apply**.
6. **Optional:** To remove a Segment from the targeting approach, click **Actions Remove** by the name of a Segment.
7. **Optional:** Repeat steps 1-5 to add more Segments to the targeting approach.

Configuring analytics driven targeting

To configure analytics driven targeting, select the business groups to include in the approach, and then configure how each business group should be used by the Strategy you are creating.

Pega Customer Decision Hub

1. In the **Targeting approach Analytics driven** section, click **Configure**.
2. In the Add groups window, select one or more business groups, and then click **Apply**.
   Note: The window only shows the business groups that you selected for the objective. For more information, see Configuring the Fixed or Dynamic Action Bundle objective or Configuring the Paid Optimization objective.
3. **Optional:** By default, the Strategy uses all actions for the selected business groups. If you want to select specific actions instead, click **All actions Select actions** , and then click **Configure**.



4. **Optional:** To view all actions in a group, click **Actions View all actions** .
5. **Optional:** To remove a group from the targeting approach, click **Actions Remove** by the name of the group.
6. **Optional:** Specify eligibility criteria for the selected actions. For more information, see Configuring Elibilities.
7. Specify how the selected actions should be prioritized. For more information, see Configuring Prioritization.

- Configuring eligibilities
- Configuring prioritization

Configuring eligibilities

When configuring analytics-driven targeting approaches for Strategies, or business issue nodes in Next-Best-Action Designer, you can use Proposition Filter rules to specify eligibility criteria for the selected actions.

Pega Customer Decision Hub
The eligibility conditions you specify in Next-Best-Action Designer are applied to the targeting approach. They can be overridden on the Eligibility tab for individual actions.

In the **Eligibilities** section, select on of the following options:

- Click **Add criteria** to define new eligibility conditions by using a condition builder. After you save the configuration, Pega Customer Decision Hub generates When rules and Proposition Filters according to the specified criteria. For more information, see Configuring engagement policies for actions.
- Click **Actions Add action-specific eligibilities** to add an action to the list that does not currently have eligibility conditions. After you add an action, you can open it and add specific eligibilities. For more information, see Configuring engagement policies for actions.
- Click **Actions Use existing eligibilities** to add an existing Proposition Filter.

What to do next:

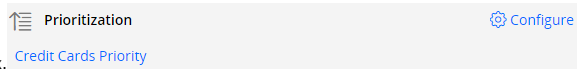
Refer to Pega Platform documentation for more details on Proposition Filter rules.

Configuring prioritization

When configuring analytics-driven targeting approaches for Strategies, or business issue nodes in Next-Best-Action Designer, you must specify how the selected actions should be prioritized. A prioritization Strategy determines the relative priority of actions in the group and ranks them so that the customer is always presented with the right action.

Pega Customer Decision Hub

1. In the **Prioritization** section, click **Configure**.
2. In the Configure Prioritization window, select the prioritization Strategy, and then click **Apply**.



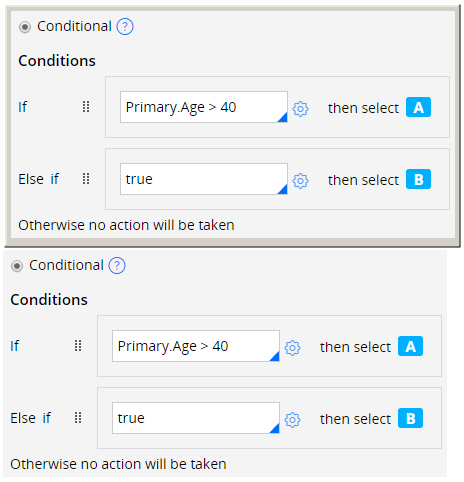
   For more information about the window, see the Using a Configuration Modal Window appendix.
   Note: The list only shows the Strategies that have the NBACategory custom field set to PRIORITIZATION. New rules created from this window automatically have this custom field set.

Arbitrating over targeting approaches

If you add multiple targeting approaches, specify how the system should arbitrate over these approaches. Each targeting approach is assigned a label (A, B, and so on). The following arbitration methods are available:

Pega Customer Decision Hub

- **A/B Testing** - Specify the percentage frequency when each targeting approach should be applied. A value must be specified for each approach and the values must add up to 100%.
- **Conditional** - Specify a conditional expression to arbitrate between targeting approaches. Drag and drop rows corresponding to targeting approaches to reorder them. The expression can use data from actions and customer entities. To access customer entities data, enter Primary.[customer property] , for example, Primary.Age .



Configuring overall prioritization

The overall prioritization configuration of a Strategy determines the order in which actions are made to a customer. You must specify the overall prioritization if you configured multiple targeting approaches or if an analytics-driven approach includes more than one business group.

Pega Customer Decision Hub

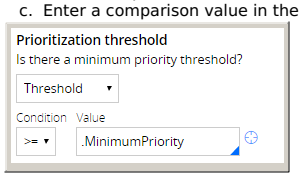
- You can configure overall prioritization in the same way as a group or issue-level prioritization. For more information, see Configuring Prioritization.

Configuring the Action Prioritization Strategy objective

The Action Prioritization Strategy objective creates a reusable Strategy that uses analytical models to determine the action priority. The Strategy uses an external input and can be applied to any set of actions.

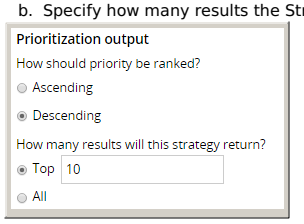
Pega Customer Decision Hub
Strategies generated for this objective have built-in propensity smoothing. This allows new actions to compete with existing actions. You can increase or decrease priority by assigning weights to results.

1. In the Prioritization section, specify how actions should be prioritized. Select one of the following options:
   - **Analytical model** - Select this option to use a predictive or adaptive model to determine action priority.
     1. Click **Configure** . The **Configure Analytical Model** pop-up window appears. The window lists the adaptive and predictive models in the system. For more information about the window, see the Using a Configuration Modal Window appendix.
     2. Select an analytical model, and then click **Apply**.
   - **Property** - Select this option to use a property as the driver for prioritization. Use the smart prompt to select the property.
2. **Optional:** In the **Prioritization calculation** section, assign weights to prioritization results. Use this option to increase or decrease priority of specific results.
   a. In the **Weighting** section, click **Configure**.
   b. In the Configure Weighting window, select the Decision Table that you want to use to assign weights to prioritization results, and then click **Apply**. For more information about the window, see the Using a Configuration Modal Window appendix.
3. **Optional:** In the **Prioritization calculation** section, customize the expression used to calculate the priority.
   a. In the **Priority** section, click **Configure** . The **Configure Priority** pop-up window appears. The window lists the Field Value rules for the **PriorityCalculation** field in the Embed-PegaMKT-NBAComponent-PriorityCalculation class. For more information about the window, see the Using a Configuration Modal Window appendix.
   b. Select the field value which you want to use to determine priority, and then click **Apply**.
4. **Optional:** In the **Prioritization threshold** section, specify a threshold to apply to the prioritization results. If a threshold is specified, results that do not meet the threshold criteria are ignored by the prioritization Strategy.
   a. Select **Threshold**.
   b. Select an operator from the **Condition** drop-down list.
   c. Enter a comparison value in the **Value** field. You can specify a static threshold value or a strategy result property as the comparison value.



Prioritization threshold
Is there a minimum priority threshold?
Threshold
Condition
Value
>=
.MinimumPriority

5. **Optional:** In the **Prioritization output** section, specify how the Strategy results should be ranked and how many results the Strategy should return.
   a. Select the order in which the results should be ranked by clicking either the **Ascending** or **Descending** radio button.
   b. Specify how many results the Strategy should return. You can choose to retrieve all results, or a specific number of the top results.



Prioritization output
How should priority be ranked?
Ascending
Descending
How many results will this strategy return?
Top
10
All

Configuring the action targeting using the Audiences and AI objective

The Audiences and AI objective creates a Strategy to rank one or more actions by priority, using various targeting approaches. You can use Segments, Eligibility rules, and Prioritization rules to achieve the desired results.

Pega Customer Decision Hub

1. Select the business issue which is addressed by this objective.
2. Select one or more business groups from the selected business issue. The business groups will be used in the generated strategy. After you select the business groups, the **Targeting approach** section appears.
3. In the **Targeting approach** section, click **+Add targeting approach**, and then select one of the following options:
   - **Audience driven** - Select this approach if you have an audience that you want to target with a specific action. An audience is made up of a starting segment, and can then be split into sub-segments.
   - **Analytics driven** - Select this approach if you want to use predictive or adaptive analytics to score and rank actions. Using analytics for targeting allows you to consider a larger set of actions.
4. Click **Apply**.
5. Configure the selected targeting approach.
   - For audience driven approaches, see Configuring Audience Driven Targeting.
   - For analytics driven approaches, see Configuring Analytics Driven Targeting.
6. **Optional:** Repeat steps 3-5 to configure more targeting approaches.
7. **Optional:** If you added multiple targeting approaches, specify how you want to arbitrate between them. For more information, see Arbitrating over Targeting Approaches.
8. **Optional:** If you added multiple targeting approaches, use the Overall prioritization section to specify how you want to prioritize across your targeting selections. This prioritization determines the order in which actions are made to a customer. For more information about configuring prioritization, see Configuring Overall Prioritization.
9. Click **Save**.

Configuring the Paid Optimization objective

The Paid Optimization objective creates a Strategy for paid channels, for example, Facebook Ads or Google Ads, to optimize bidding based on action priority. To use this option, Paid Media Manager must be enabled in your environment. For more information, see Extending next-best-action to digital advertising platforms with Paid Media Manager.

Pega Customer Decision Hub

Creating strategies manually

Manual Strategy creation allows you to create Strategies for all customer contexts in your application. To manually create a new Strategy, click **Create Start with new canvas** on the Strategies landing page.

Pega Customer Decision Hub

1. Enter a short description of the Strategy. The Strategy identifier is created automatically.
2. Select the Strategy Result class, or define a custom Strategy Result class instead. For more information about customizing Strategy Result classes, see the *Pega Customer Decision Hub Implementation Guide* on the Pega Customer Decision Hub product page.
3. Select the starting decision context, which will add an embedded strategy to the canvas. The embedded strategy will be configured with the data that is defined in the Context Dictionary. For more information about the Context Dictionary, see the *Pega Customer Decision Hub Implementation Guide* on the Pega Customer Decision Hub product page.
4. Click **Create and open**.
5. In the Strategy canvas, configure the Strategy.
   - For general information about Strategy configuration in Pega Platform, see About Strategy rules.
   - For information about Strategy configuration specific to Pega Customer Decision Hub, see Configuring Pega Customer Decision Hub Strategies.
6. Click **Save**.

Configuring Pega Customer Decision Hub strategies

Strategies in Pega Customer Decision Hub are configured in a similar way as in the Pega Platform. For more information, see About Strategy rules. However, Pega Customer Decision Hub provides some additional options.

Pega Customer Decision Hub

The following Strategy configuration options are specific to Pega Customer Decision Hub:

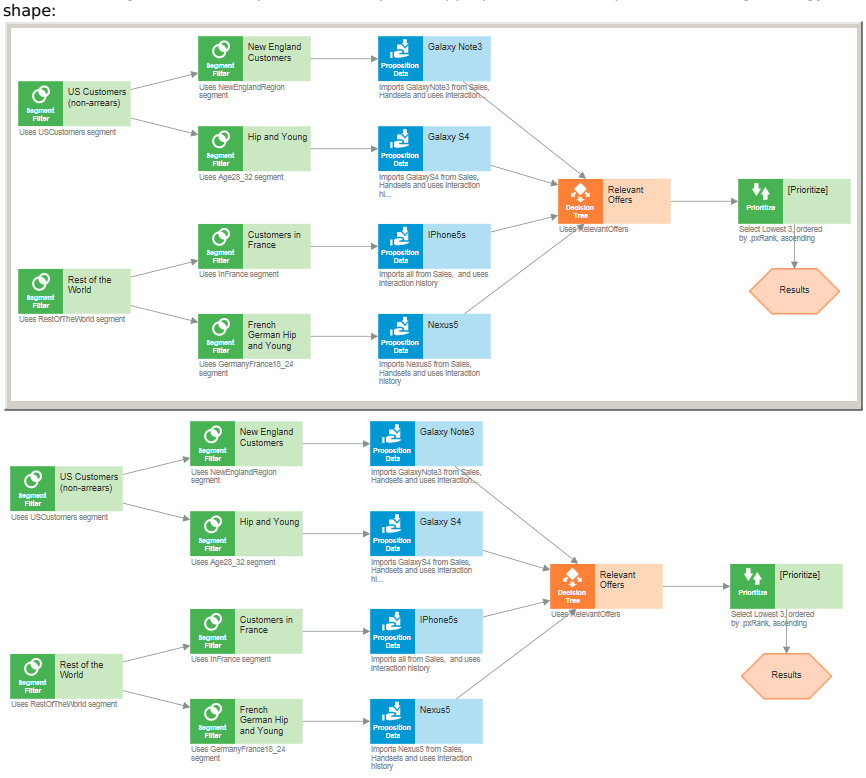
- Using segments in strategies
- Using contact policies in strategies
- Using Geofences in strategies

Using segments in strategies

Pega Customer Decision Hub Strategies can use Segments to filter results directly in the Strategy itself. To do this, add a Segment Filter shape to the Strategy canvas and configure its properties to reference a Segment you created. For more information about Segments, see Segments.

Pega Customer Decision Hub

1. To add a Segment Filter shape, in the Strategy canvas, right-click anywhere in the canvas, and then click **Arbitration Segment Filter** .
2. Right-click the Segment Filter shape, and then click **Properties**.
3. In the Segment Filter Properties window, in the **Filter Condition Segment rule** field, select a Segment which you want to use to filter Strategy results.
4. Click **Submit**.
5. Connect the Segment Filter shape to other shapes as appropriate. For example, the following Strategy uses Segment Filter shapes to provide data to a Decision Tree shape:



Using contact policies in strategies

Pega Customer Decision Hub Strategies can use Contact Policies to restrict the Strategy output based on channel limitations that are specified in the Contact Policy rule. To do this, add a Contact Policy shape to the Strategy canvas and configure its properties to reference a Contact Policy that you created. For more information about Contact Policies, see Contact Policies .

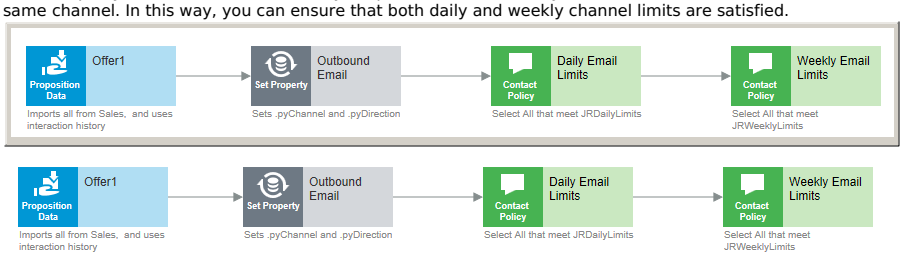
Pega Customer Decision Hub
Before you begin:

Before adding a Contact Policy shape to the Strategy, make sure that the following prerequisites are met:

- The Strategy must include a Proposition Data shape. The Proposition Data shape imports actions which the Contact Policy shape uses as input. **Note:** For Strategies that include an action bundle, only the action marked as Bundle Parent must be imported as input. For more information, see Contact policies and action bundles.
- Set the **pyDirection** property on each action where Contact Policies should apply to "Outbound" . Contact Policies only apply to actions on outbound channels.
- Set the **pyChannel** property on each action where Contact Policies should apply to one of the channel names in the referenced Contact Policy rule. Contact Policies only apply to actions on channels that are restricted in the referenced Contact Policy rule.
- Mark the referenced Contact Policy rule as available.
- Make sure that the Contact Policy is currently valid, that is, its start date is not in the future and its end date is not in the past.

If the prerequisites are not met, Contact Policy limits are ignored, except for settings in the **Contact Policy Output** section. The Contact Policy will always limit the number of actions to the value specified in the Output section, even if other limits are ignored. For more information, see step 4 below.

1. To add a Contact Policy shape, in the Strategy canvas, right-click anywhere in the canvas, and then click **Arbitration Contact Policy** .
2. Right-click the Contact Policy shape, and then click **Properties**.
3. In the Contact Policy Properties window, in the **Contact Policy Context** field, select a Contact Policy that you want to use to restrict Strategy results.
4. **Optional:** In the **Contact Policy Output** section, specify the number of actions that are allowed by the Contact Policy shape. Select on of the following options:
   - First – This option restricts the output of the Contact Policy shape to a specific number of actions. For example, if you enter 5 as the number of allowed actions, the Contact Policy shape filters out all actions after the first five that satisfy the Contact Policy criteria. The default number of allowed actions is 1. **Note:** Ensure that actions are prioritized before the Contact Policy shape so that Contact Policy can restrict the output to the highest-priority actions.
   - All – This option does not filter out any actions that satisfy the Contact Policy criteria.
5. Click **Submit**.
6. **Optional:** Add more Contact Policy shapes to the canvas.
   For example, you can link a Contact Policy shape which sets the daily limit of actions on the Email channel to a Contact Policy shape which sets the weekly limit on the same channel. In this way, you can ensure that both daily and weekly channel limits are satisfied.



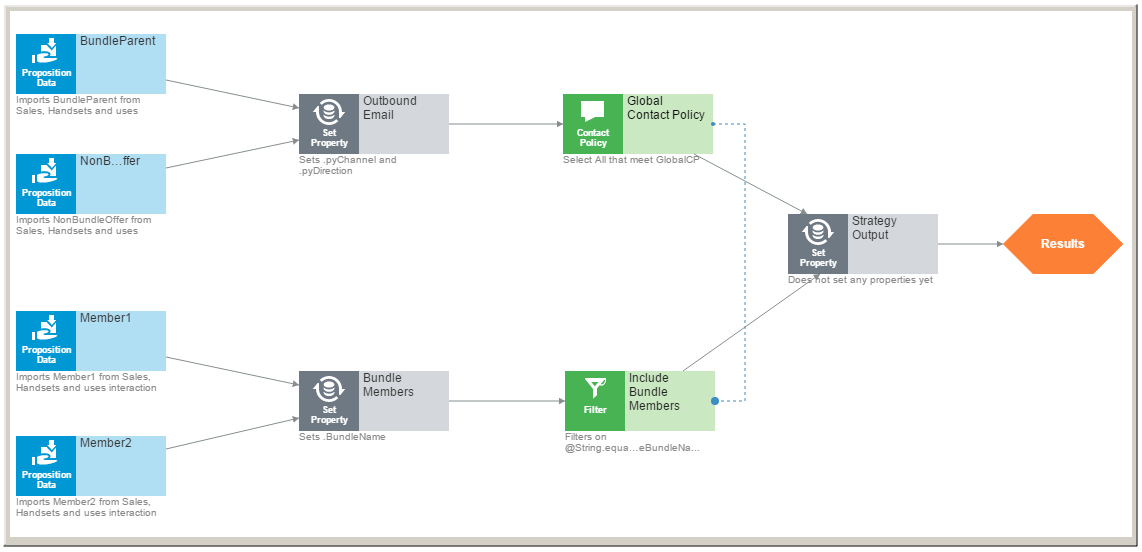
- Contact policies and action bundles

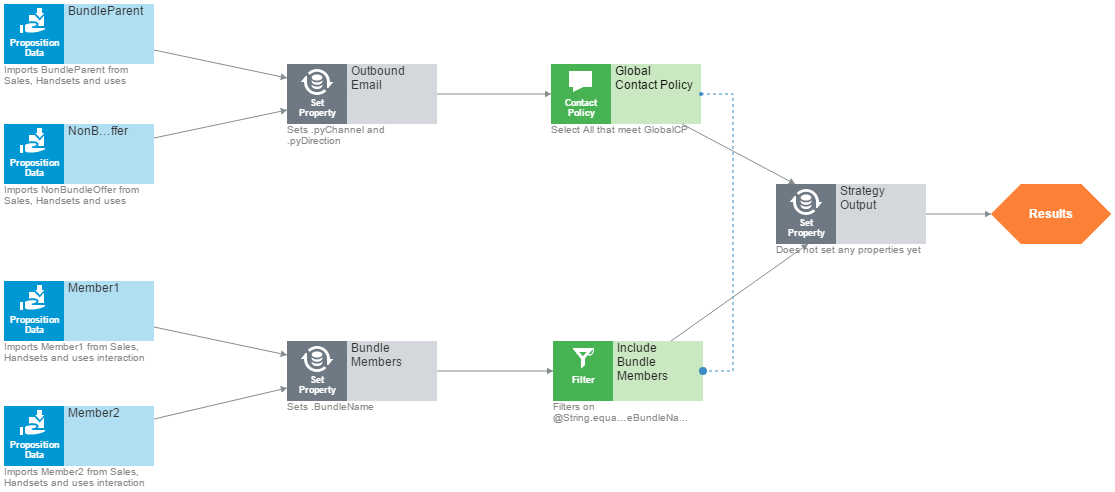
Contact policies and action bundles

For Strategies that include an action Bundle, only the action marked as Bundle Parent must be imported as input. If the Contact Policy shape allows the Bundle Parent to be included in the Strategy results, actions marked as Bundle Members should be added to the Strategy output.

Pega Customer Decision Hub

For example, the following Strategy uses the **Include Bundle Members** Filter shape to add the Bundle Member actions only if the Contact Policy shape allows the Bundle Parent action as output. To achieve this, set the Filter condition to @String.equals(GlobalContactPolicy.BundleName,"SampleBundleName").



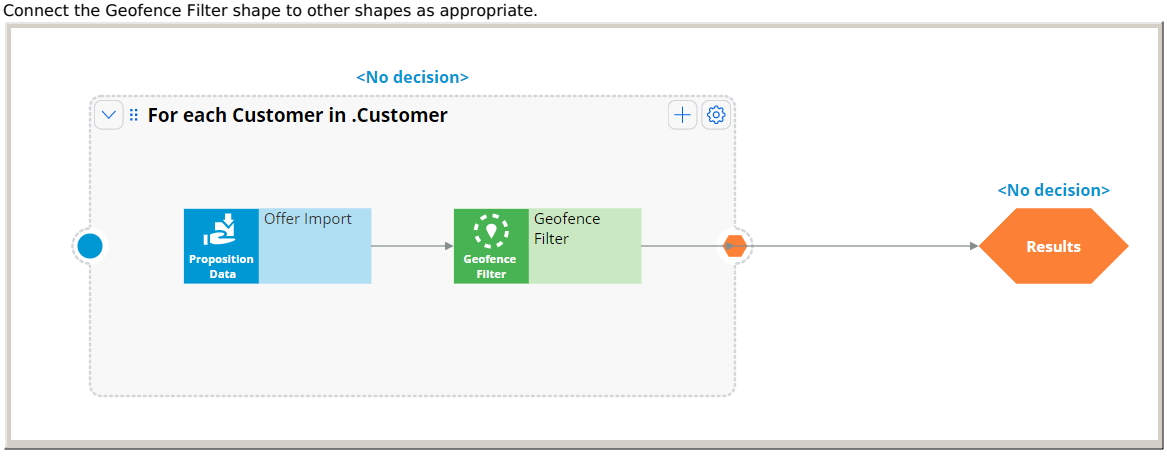


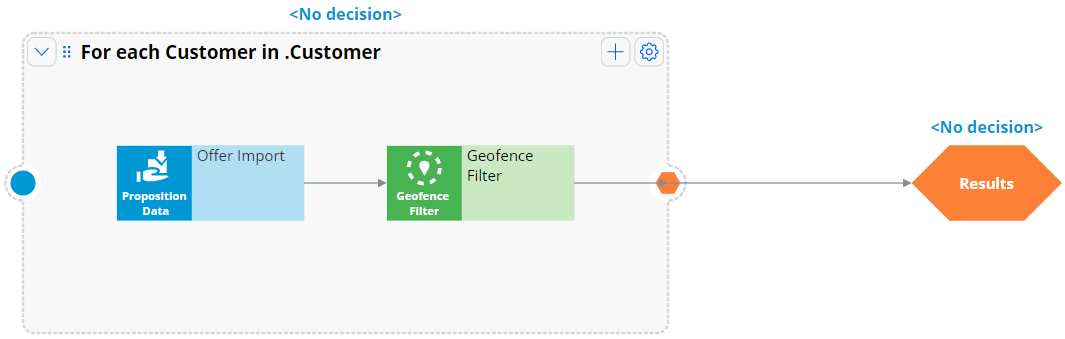
Using Geofences in strategies

Pega Customer Decision Hub Strategies can use Geofences to filter action inputs, or as a driver preceding proposition import. To do this, add a Geofence Filter shape to the Strategy canvas and configure its properties to reference a Geofence you created. For more information about Geofences, see Geofences .

Pega Customer Decision Hub

1. To add a Geofence Filter shape, in the Strategy canvas, right-click anywhere in the canvas, and then click **Arbitration Geofence Filter** .
2. Right-click the Geofence Filter shape, and then click **Properties**.
3. In the Geofence Filter Properties dialog box, in the **Geofences Select Geofences to check against customer location** section, specify whether you want to add Geofences one at a time, or in bulk. The following options are available:
    - **Select Geofences one at a time**
    - **Select multiple Geofences**
4. Select one or more Geofences that you want to use to filter action inputs. When the Strategy runs, each Geofence rule is evaluated until a Geofence is satisfied, that is, the Customer location is within the Geofence. The Geofence rules are evaluated from the top to the bottom of the list. Reorder the list if you want to give priority to specific Geofences.
5. In the **Options Select the shape behavior when customer location data is unavailable** section, specify how the Geofence shape should behave when the Customer location is not available. Select one of the following options:
    - **Exclude source component results**
    - **Pass through source component results**
6. **Optional:** In the **Options Select Customer location properties** section, configure the properties that are used to determine the latitude and longitude of the Customer location. The default location properties are set by the application administrator. To change the location properties, select one of the following options:
    - Location data from the Customer profile – Prefix the property with `Primary.` , for example, `Primary.Latitude`
    - Location data from the action data – Prefix the property with a period, for example, `.Latitude`
    - Calculated fields – Prefix the property with the name of a Strategy shape
7. **Optional:** In the **Options Select Customer location properties** section, select the **Use Customer location from Real-Time Event if available** check box to use location data from a triggered Event. If location data is not available for the Event, the customer location fields specified in step 6 will be used. For more information about Events, see Real-Time Events.
8. Click **Submit**.
9. Connect the Geofence Filter shape to other shapes as appropriate.





Configuring ethical bias checks

You can test your strategies for unwanted bias by creating an ethical bias policy. For example, you can test whether your strategies generate biased results by sending more actions to male rather than female customers, as well as check how big the bias is. Perform the following procedure to configure the fields that you want to use to measure bias, and to define the thresholds for detecting bias.

Pega Customer Decision Hub
Before you begin: Assign the *pzBiasPolicyConfiguration* privilege to your access group. For more information, see "Optional: Granting permission to configure ethical bias policies" in the *Pega Customer Decision Hub Implementation Guide* on the Pega Customer Decision Hub product page.

1. In the Pega Customer Decision Hub portal, click **Configuration Ethical Bias Policy**
2. On the **Bias fields** tab, click **Add bias field**.
   Tip: Any properties that are related to your main customer class, such as gender, age, and ethnicity-related properties, are good candidates for bias testing, even if your strategies or AI models do not directly reference these properties. This is because indirect bias can still occur.
3. If the property that you select has a number value, in the **Add bias field** window, specify whether you want Pega Platform to interpret the value as a category (for example, gender or ethnicity), or as an ordinal number (for example, for age).
   Note: If there are many categorical values, the system only checks the 20 most frequent values for bias. In most cases, numerical values should not be classified as a category. However, if the property is not truly an ordinal number, you may want to treat it as a category, for example, you can use a postal code to test for location-related bias.
4. On the **Bias fields** tab, review and configure the bias threshold settings for each issue in your business structure.
   The bias threshold measurement depends on the type of field that you select:
   - *Rate ratio* - Rate ratio is used to calculate bias for categorical fields by comparing the number of customers who were selected for an action to those not selected for an action, and then correlating the result to the selected bias field. For example, the rate ratio represented in the following table indicates that actions are sent more often to male rather than female customers:

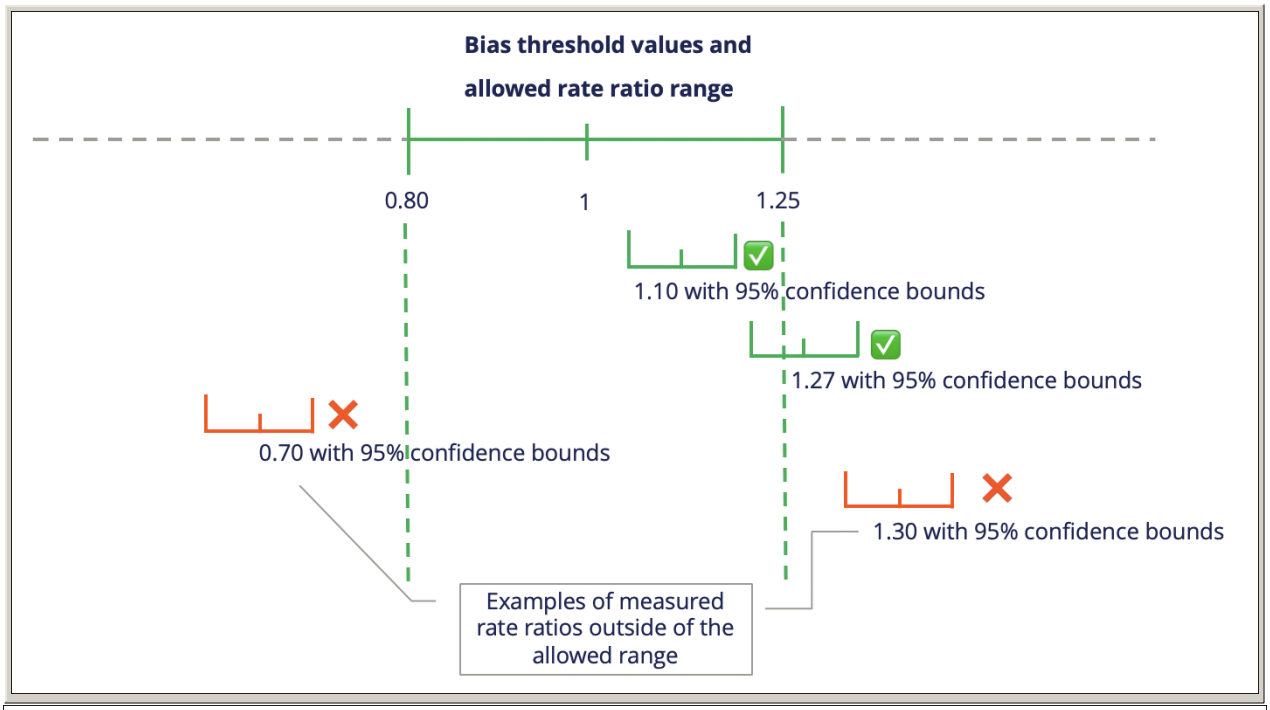
	Female customers	Male customers
selected for the action	500	1000
not selected for the action	20000	18000
rate ratio	0.46	2.16

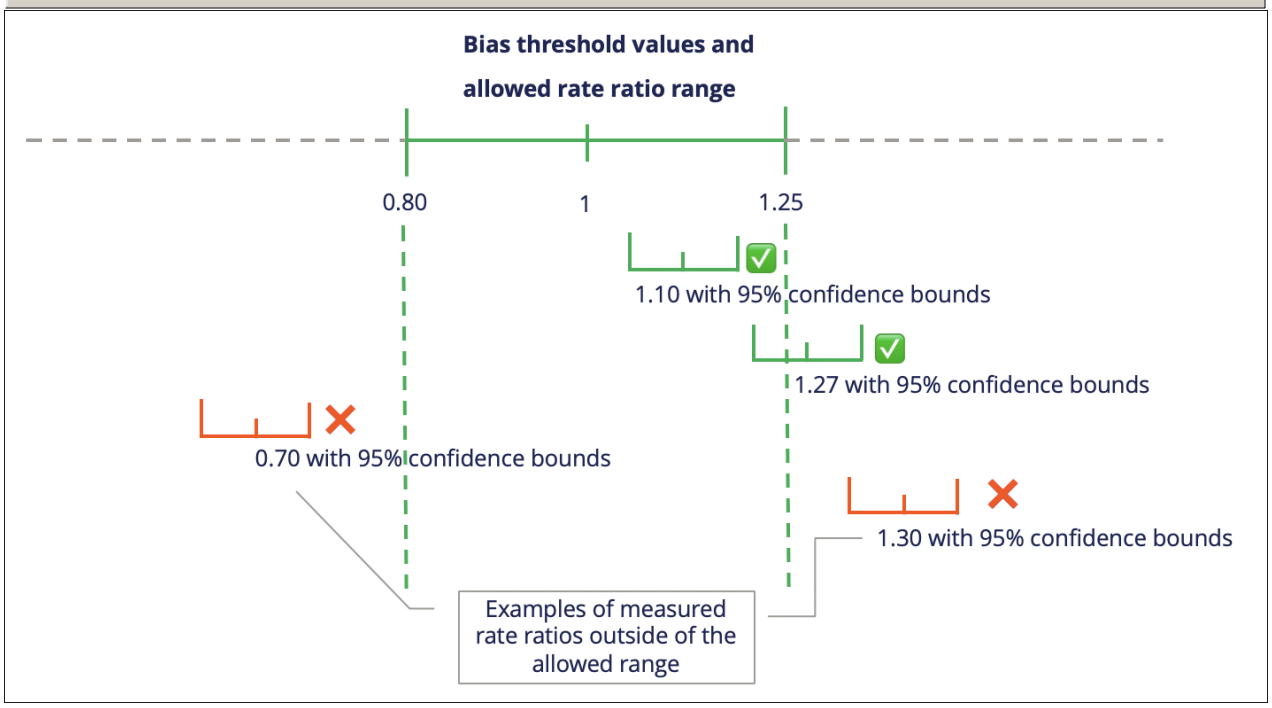
A rate ratio of 1 represents no shifts in the distribution. You can select a warning range with a threshold between **0** (send a warning if any bias is detected), and **0.7** (send a warning only if very high bias is detected). You can also choose to ignore this bias field for a particular issue in your business structure. For each bias threshold setting, there is no difference made between a positive shift (towards > 1), and a negative shift (towards < 1)

Bias threshold	Allowed rate ratio range
no bias allowed	any bias with a 95% confidence for the rate ratio being greater than or less than 1 will be detected
very light	0.9-1.11
light	0.8-1.25
heavy	0.66-1.50
very heavy	0.5-2.0
all bias allowed	no bias detection

For the rate ratio, the confidence interval is calculated using the approximation on the error on log on.

Bias threshold values and allowed rate ratio range





- *Gini coefficient* - The Gini inequality coefficient is used to calculate bias for numerical fields. For example, it detects whether the distribution of age is different for customers who receive an action. A Gini coefficient of 0 represents no shifts in the distribution. You can select a warning threshold between **0** (send a warning if any bias is detected) and **0.50 - 2.00** (send a warning only if very high bias is detected). If the measured Gini coefficient is outside of the allowed range with a confidence interval of 95%, it will signify a significant bias. You can also choose to ignore this bias field for a particular issue in your business structure.**Tip:** For issues such as credit risk, set the value of the threshold lower than for marketing issues such as upsell, to better detect the variations.

Bias threshold	Allowed Gini range
very light	< 0.1
very light	< 0.2
heavy	< 0.5
very heavy	< 0.7
no bias allowed	any bias with a 95% confidence for the Gini coefficient to be greater than 0 will be detected
all bias allowed	no bias detection

The confidence interval for the Gini coefficient is calculated using Delong's method.

Note: The confidence level for detecting bias above threshold can be set through a Dynamic System Setting (DSS). The default value is 0.9999 (or 99.99%). Lowering this level increases the probability of false alerts on bias detected.

Dynamic System Setting for the confidence level used to determine the interval width

Pega-DecisionEngine	*decision/simulation/ethicalbias/confidenceinterval*	0.9999

5. Click **Save**.
6. To use the bias policy to test the behavior of your strategies:
    a. In the Pega Customer Decision Hub portal, click **Simulation Testing**.
    b. Create a new simulation test with the purpose **Ethical bias**.
    For more information, see Running a decision strategy simulation on sampled production data.

Supplementing next-best-action with traditional outbound campaigns

In Pega Customer Decision Hub, campaigns provide an additional mechanism to deliver actions to customers and monitor the action performance.

Pega Customer Decision Hub
Important:

Before the introduction of Next-Best-Action Designer and the Next-Best-Action strategy framework, campaigns were the main mechanism by which actions were delivered to customers. In newer versions of the product, this functionality has been superseded by the practices of 1:1 Customer Engagement, embodied within the product through the use of Next-Best-Action Designer. As a best practice, you should configure your customer engagement strategy in Next-Best-Action Designer, as described in Improving customer experience with next-best-action on owned channels.

The following links are provided for reference for users who have already configured campaigns in an earlier version of their Pega Marketing application. If you are a new user of Pega Customer Decision Hub, campaigns are disabled in your application by default and can only be enabled with a separate license. If you have purchased the license, see "Enabling campaign editing" in the *Pega Customer Decision Hub Implementation Guide* on the Pega Customer Decision Hub product page.

- Self-Optimized Campaigns
- Multi-Channel Campaigns
- Outbound Campaigns
- Defining outbound limits for traditional marketing campaigns

Self-Optimized Campaigns

Self-Optimizing Campaigns provide an intuitive user experience that automates the process of splitting a campaign into a series of waves and leveraging the power of adaptive analytics to automatically learn which actions appeal to your customers the most.

Pega Customer Decision Hub

By continuously monitoring and applying the learning from previous waves to the next wave, Self-Optimizing Campaigns deliver incremental lift and improved outcomes across the entire campaign.

This chapter includes the following sections that explain various aspects of Self-Optimizing Campaigns:

- Creating a Self-Optimizing Campaign
- Self-Optimizing Campaign configuration

Creating a Self-Optimizing Campaign

To create a Self-Optimizing Campaign, click the **Create** button on the Campaigns landing page and select **Create Self-Optimizing Campaign**.

Pega Customer Decision Hub

A new Campaign launches with the Campaign display in edit mode. You can start designing your Campaign right from this display.

The Campaign's name must be unique across the system. Unlike some of the other rules (such as action), the same Campaign name cannot be used at different hierarchical levels (Top/Issue/Group). The name must also begin with a letter and should not contain any special characters (except spaces and underscores).

Self-Optimizing Campaign configuration

Marketers can start configuring the Campaign right from the **Create a campaign** screen. Once the Campaign has been created, they can use the **Edit** button to go into edit mode and make further updates to the Campaign configurations.

Pega Customer Decision Hub

As the user updates the Campaign's configuration, the updated values are displayed in the overview pane on the right. This pane provides an overview of the Campaign's various configurations. The overview pane is displayed at all times and includes links to open relevant artifacts.

The following sections provide details on the steps involved in configuring a Campaign:

- Configuring Self-Optimizing Campaign Details
- Configuring the Audience for Self-Optimizing Campaigns
- Configuring the Actions for Self-Optimizing Campaigns
- Configuring the Outcome Optimization for Self-Optimizing Campaigns

- Configuring Self-Optimizing Campaign Details
- Configuring the Audience for Self-Optimizing Campaigns
- Configuring the Actions for Self-Optimizing Campaigns
- Configuring the Outcome Optimization for Self-Optimizing Campaigns

Configuring Self-Optimizing Campaign Details

The Campaign details section for Self-Optimizing Campaigns allows you to specify general information about the campaign, including its objective, start and end dates, financial details, and others.

Pega Customer Decision Hub

1. In the **Campaign Details** section, click **Configure**.
2. In the **Configure Campaign Details** pane, fill out the required information.
3. After you specify the start and end date for the campaign, the **Campaign run schedule** section appears. You can use this section to review information about the planned campaign waves.
   The number of planned waves is automatically calculated based on the length of the campaign in days divided by the duration of each wave. To change the duration of each wave, and the overall number of planned waves, edit the **Each wave** field.

Configuring the Audience for Self-Optimizing Campaigns

The Audience section for Self-Optimizing Campaigns allows you to define the audience for the campaign. The audience can be based on an existing Segment, or randomly generated based on a specified audience size. If you have multiple customer contexts defined, self-optimizing campaigns target the primary customer context in your application.

Pega Customer Decision Hub

1. In the **Audience** section, click **Configure**.
2. In the **Configure Audience** pane, specify whether you want to use an existing Segment, or automatically generate a Segment populated with randomly selected customers.
    - To automatically generate a random Segment, select **Specify an audience size** and enter the number of customers that should be included in the audience. To ensure that the analytics results are statistically significant, the audience size should be larger than 2,500.
    - To use an existing Segment, select **Select a segment**, and then click Configure to launch the Configure Segment modal window. See Using a configuration modal

window for details on using this window.
3. In the **Set control group** field, specify the approximate size of the control group used for this audience. The control group is a percentage of customers who receive an offer that is randomly selected rather than based on analytics. Control groups allow you to measure the effectiveness of your analytics.

Configuring the Actions for Self-Optimizing Campaigns

The Actions section for Self-Optimizing Campaigns allows you to select the outbound channel and actions to be used in this campaign.

Pega Customer Decision Hub

1. In the **Actions** section, click **Configure**.
2. In the **Configure actions** pane, select the outbound channel. Actions will be filtered based on the selected channel. Currently Self-Optimizing campaigns only allow a single channel.
3. Click **Configure** to select the actions. See Using a configuration modal window for details on using the Configure actions window.

Configuring the Outcome Optimization for Self-Optimizing Campaigns

The Outcome optimization section for Self-Optimizing Campaigns allows you to select a metric that should be used to monitor and improve campaign performance. For example, select Conversion Rate to prioritize actions by the ratio of actions sent to actions accepted. This optimizes the campaign to prioritize actions that will lead to higher conversions. Only rate metrics can be used to optimize outcomes.

Pega Customer Decision Hub

1. In the **Outcome optimization** section, click **Configure**.
2. In the **Configure Outcome optimization** pane, click **Configure** to select a rate metric. See Using a configuration modal window for details on using the Configure Metrics window.

Multi-Channel Campaigns

Multi-Channel Campaigns (commonly referred to as Campaigns) enable marketers to use the power of intelligent segmentation and analytics to deliver the right offering to the right customer at the right time.

Pega Customer Decision Hub

This chapter includes the following sections that explain various aspects of Multi-Channel Campaigns:

- Campaign Management
- Campaign Creation
- Configuring an Outbound Campaign
- Campaign Approval
- Campaign Validation
- Campaign Testing
- Campaign Execution
- Campaign Monitoring
- Campaign Completion

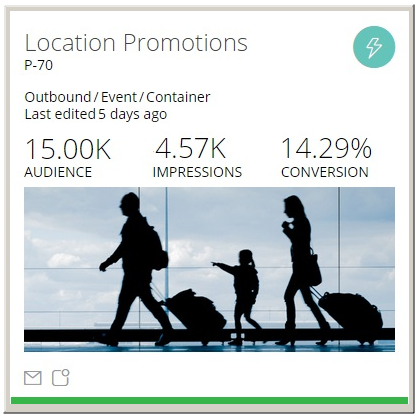
Campaign Management

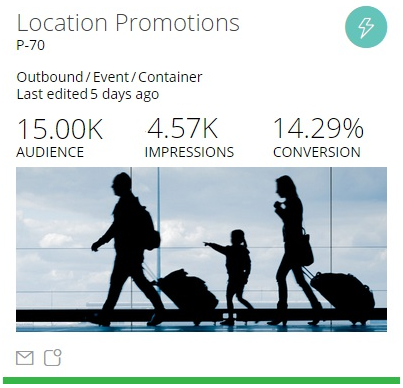
Campaigns are listed on the Campaigns landing page which is accessed in the Pega Customer Decision Hub portal via the **Campaigns** link in the left navigation.

Pega Customer Decision Hub

This landing page lists the twenty most recently updated Multi-Channel Campaigns and Outbound Campaigns available in the system. The **View more results** link (at the bottom) enables users to view the next twenty results. Users can utilize the filters at the top of the page to filter the results by Status, Issue / Group, Name, and Created by.**Note:** A Campaign is deemed available if its status is not Archived or Withdrawn.

Each Campaign is represented by a card in the landing page.





Users can click anywhere on the card to open the Campaign. The card displays the following salient information about the Campaign:

- Name
- Work id
- Status icon - icon's tooltip displays status text
- Enabled engagement types - available types are Outbound, Event, and Container
- When the Campaign was last updated
- Configured audience size
- Total number of impressions
- Overall conversion rate
- Image associated with the Campaign
- Channel icon array, containing icon for each channel utilized within the Campaign - icon's tooltip displays channel name
- Conversion goal achievement indicator - This is the color bar at the bottom of the card and indicates performance against the Outbound Conversion Rate goal. It displays one of the following colors:
  - Gray - If Outbound Conversion Rate is not a configured goal for the Campaign or when the Campaign is in design mode.
  - Green - If the conversion rate goal has been met
  - Red - If the conversion rate goal is not met

Campaign Creation

Users can utilize either of the following mechanisms to create Multi-Channel Campaigns:

Pega Customer Decision Hub

- Click the **Create** button on the Campaigns landing page
- Click the **Create a new campaign** shortcut widget in the Pega Customer Decision Hub portal Home page

A new Campaign launches with the Campaign display in edit mode. You can start designing your Campaign right from this display.

The Campaign's name must be unique across the system. Unlike some of the other rules (such as action), the same Campaign name cannot be used at different hierarchical levels (Top/Issue/Group). The name must also begin with a letter and should not contain any special characters (except spaces and underscores).

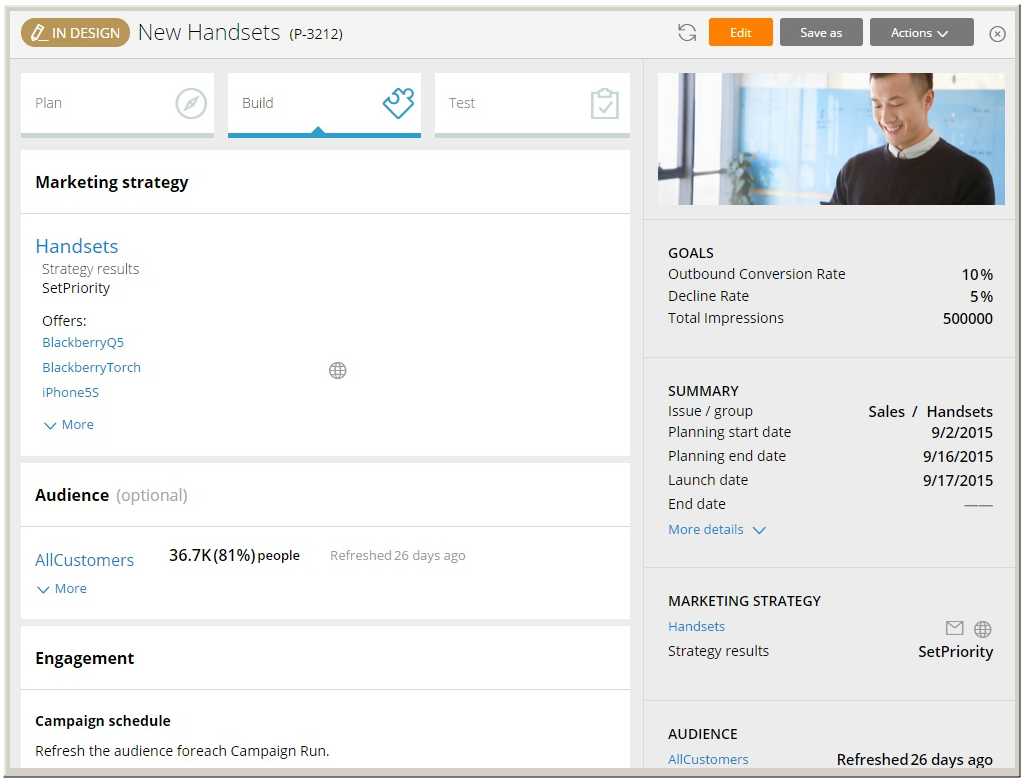
Campaign Configuration

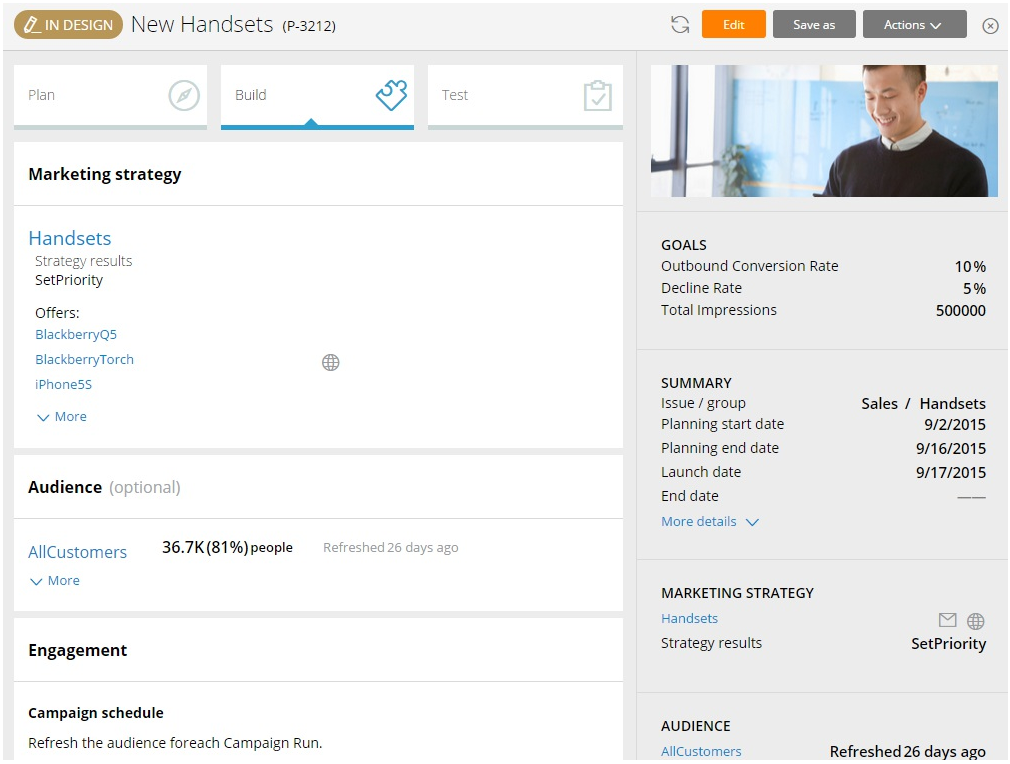
Marketers can start configuring the Campaign right from the **Create a campaign** screen. Once the Campaign has been created, they can use the **Edit** button to go into edit mode and make further updates to the Campaign's configurations.

Pega Customer Decision Hub

As the user updates the Campaign's configuration, the updated values are displayed in the overview pane on the right. This pane provides an overview of the Campaign's various configurations. The overview pane is displayed at all times and includes links to open relevant artifacts.

A sample configured Campaign is shown below:





The following sections provide details on the steps involved in configuring a Campaign:

- Specifying Details
- Specifying Goals
- Selecting a Strategy
- Selecting an Audience
- Configuring Campaign Engagement
- Selecting a Constraint

- Specifying Multi-Channel Campaign Details
- Specifying Multi-Channel Campaign Goals
- Selecting a Strategy for Multi-Channel Campaigns
- Selecting an Audience for Multi-Channel Campaigns
- Configuring Campaign Engagement
- Setting volume constraints for a campaign

Specifying Multi-Channel Campaign Details

Users can utilize the **Plan** tab to specify various details about the Campaign. These details are organized into the following sections:

Pega Customer Decision Hub

- Campaign details
- Financials

Campaign details

This section includes basic details about the Campaign and includes the following:

- **Name**
- **Issue**
- **Group**
- **Business objective**
- **Context** - Type of audience to target via this Campaign. The value selected for this field determines the items presented in the Audience and Marketing strategy sections in the Build tab. By default, the available contexts are **Customer** or **Prospect**. You can define additional Customer populations, for example, Accounts, to better target your campaigns. For more information, see "Setting the context for your customer class" in the *Pega Customer Decision Hub Implementation Guide* on the Pega Customer Decision Hub product page.
- **Campaign type**
- **Key code**
- **Campaign image** - Image associated with the Campaign. This image is displayed in the right overview pane and in the card for the Campaign on the Campaigns landing page.
- **Priority**
- Anticipated planning and launch dates
- **Ignore global exclusion list** - Authorized users can enable this option to have outbound processing of actions for this Campaign ignore the global exclusion list. Refer to the Exclusion List chapter for details.

Note: The Campaign's Name, Issue, and Group values cannot be changed once it is created.

Financials

This section includes fields to specify financial metrics for the Campaign. These values typically play an important role in the approval of the Campaign.

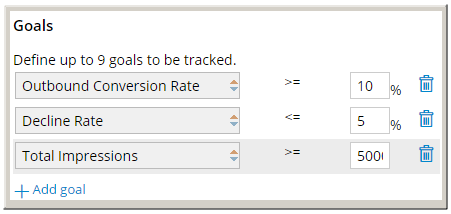
Specifying Multi-Channel Campaign Goals

The **Goals** section in the **Plan** tab enables the user to specify goals for their Campaign. Users can add up to nine goals. Each available goal is a Key Performance Indicator (KPI) configured on the Visual Business Director landing page.

Pega Customer Decision Hub

The operator displayed between the goal field and the target value automatically toggles (between >= and <=) depending on whether higher or lower performance of the goal metric is the desirable result. For example, the Outbound Conversion Rate goal utilizes the >= operator while the Decline Rate goal utilizes the <= operator.

Similarly, a percentage symbol is displayed alongside appropriate goals.



Goals
Define up to 9 goals to be tracked.
Outbound Conversion Rate
>=
10
%
Decline Rate
<=
5
%
Total Impressions
>=
500
Add goal

Selecting a Strategy for Multi-Channel Campaigns

The **Marketing strategy** section in the **Build** tab enables users to designate a Strategy rule as the backing strategy for their Campaign.

Pega Customer Decision Hub

Clicking the **Configure** button launches the **Configure Marketing Strategy** modal window which lists available Strategies in the system. For more information about using this window, see the Using a Configuration Modal Window appendix.

A sample list entry for a Strategy rule is shown below.

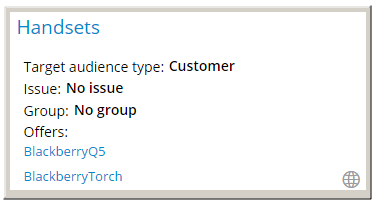


Handsets
Edited 6 days ago
SalesHandsets

This view includes the following information about the Strategy:

- Friendly name / label
- Rule name
- When the Strategy was last updated

Selecting this Strategy displays further information about it in the details panel.

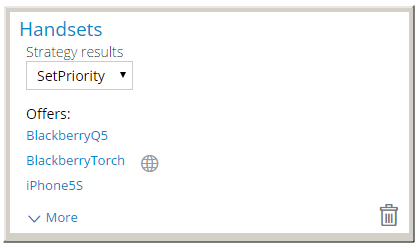


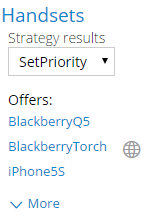
Handsets
Target audience type: Customer
Issue: No issue
Group: No group
Offers:
BlackberryQ5
BlackberryTorch

This view displays the following information about the Strategy:

- Friendly name - Click to open the Strategy rule.
- Description
- Target audience type
- Issue, Group categorization
- List of actions output by the Strategy. Along with the action, icons are displayed for each channel that the action utilizes. Clicking an action name opens the action rule.

After selecting a Strategy, the user must also select the desired result component to use from the Strategy. This is done via the **Strategy results** field in the Marketing strategy section. Once a results component has been selected, the system displays the actions that are utilized by the selected component along with their channel icons. The delete icon can be used to dissociate the selected Strategy from the Campaign.





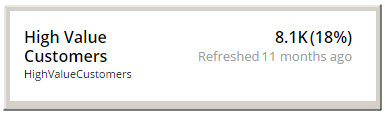
Selecting an Audience for Multi-Channel Campaigns

The **Audience** section in the **Build** tab enables the selection of the desired audience for the Campaign. Audience selection is optional for certain engagement types. It is required when the Campaign has scheduled executions. It is also required if customer validation is enabled in an event-based engagement.

Pega Customer Decision Hub

Clicking the **Configure** button launches the **Configure Audience** modal window which lists available Segments in the system. For more information about using this window, see the Using a Configuration Modal Window appendix.

A sample list entry for a Segment rule is shown below.



High Value Customers
HighValueCustomers
8.1K (18%)
Refreshed 11 months ago

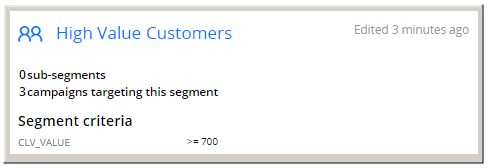
This view includes the following information about the Segment:

- Friendly name / label
- Rule name
- Total number of customers in the Segment
- Percentage of the total customer population included in this Segment
- When the Segment population was last refreshed

Selecting this Segment displays further information about it in the details panel. This panel displays the following information for all Segment types:

- Segment image
- Friendly name - Click to open the Segment rule.
- When the Segment rule was last edited
- Number of Campaigns utilizing this Segment

Criteria Segments

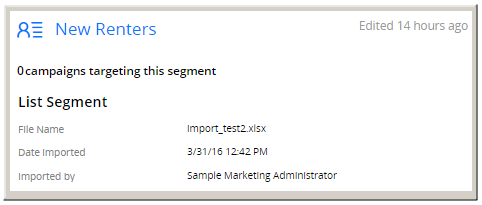


High Value Customers
Edited 3 minutes ago
0 sub-segments
3 campaigns targeting this segment
Segment criteria
CLV_VALUE
>= 700

The following information is only shown for criteria Segments:

- Number of Segments referenced within this Segment
- List of criteria utilized by the Segment

List Segments



New Renters
Edited 14 hours ago
0 campaigns targeting this segment
List Segment
File Name
Import_test2.xlsx
Date Imported
3/31/16 12:42 PM
Imported by
Sample Marketing Administrator

The following information is only shown for list Segments:

- Name of the last imported customer file
- Date and time when this file was imported

- Name of the user who imported this file

Configuring Campaign Engagement

The **Engagement** section in the **Build** tab enables the configuration of the modes in which the Campaign engages with the customers.

Pega Customer Decision Hub

Clicking the **Configure** button launches the **Configure Engagement** modal window which presents the following three engagement types:

- Campaign schedule
- Real-Time events
- Real-Time containers

Users can enable multiple engagement types on the same Campaign.

- Campaign schedule
- Configuring event-driven campaign execution
- Exposing the results of a marketing campaign to real-time channels
- Configuring a Fallback Strategy

Campaign schedule

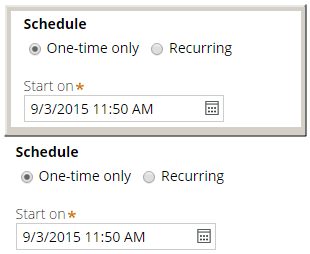
This engagement type represents scheduled Campaign execution. Users can configure desired scheduling options and the system will automatically run the Campaign based on the configured schedule.

Pega Customer Decision Hub

To begin with, the user must select between single or recurring execution.

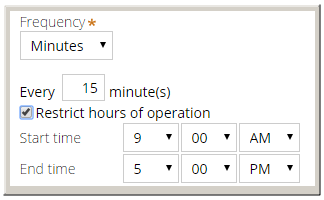
The following additional configuration is available for the One-time only option:

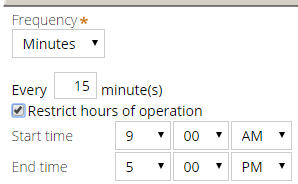
- **Start on** – Specify the start date and time for the Campaign execution.



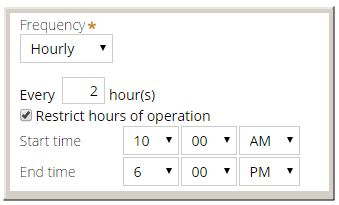
The following additional configurations are available for the Recurring option:

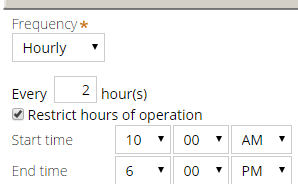
- **Frequency** – Select the recurrence frequency. Options available are:
  - **Minutes** - Specify the number of minutes in which to recur. Additionally, the user can also choose to specify the hours of operation which are applicable to this mode of recurrence.





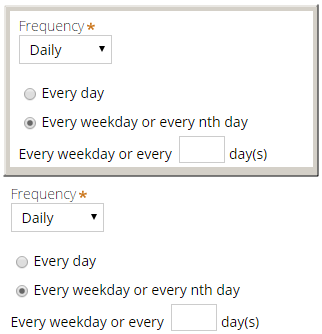
  - **Hourly** - Specify the number of hours in which to recur. Additionally, the user can also choose to specify the hours of operation which are applicable to this mode of recurrence.



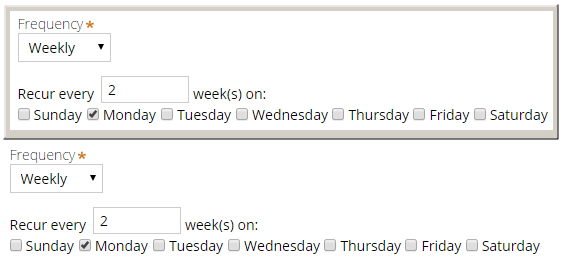


  - **Daily** - The user can choose between the following three options:
    - Every day
    - Every weekday - leave days field empty (shown below)

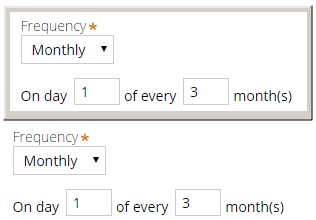
- Every X days – specify X in days field



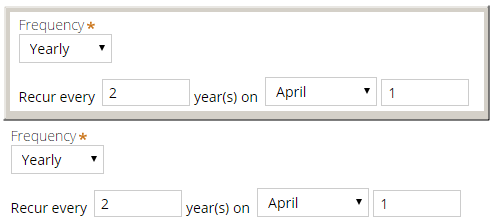
- **Weekly** - The following modes are supported:
  - Recur every X weeks based off a start date – specify X in weeks field, do not select any week days
  - Recur every X weeks on selected days – specify X in weeks field, select week days (as shown below)



- **Monthly** - Specify the day of the month and the number of months in which to recur. The following shows a quarterly configuration:



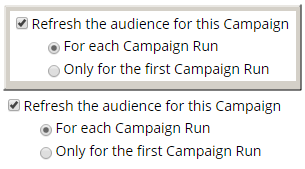
- **Yearly** - Specify the day of the year and the number of years in which to recur.



- **Start campaign** – Select the start date and time for Campaign execution.
- **End campaign** – Determines when scheduled Campaign execution should end. Select from the following three options:
  - **No end date**
  - **Set number of occurrences** - Specify the number of occurrences.
  - **End by a date** - Specify the end date and time.

Refreshing Campaign audience

In addition to configuring scheduling options, the user can also determine whether to refresh the audience for the Campaign as part of the scheduled execution. The user can further opt between refreshing the population for each run or just for the first run.



Note: The refresh audience option is only available when the Campaign audience is a criteria Segment.

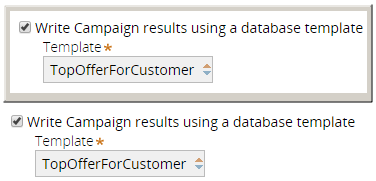
Direct output to template

For Multi-Channel Campaigns, Pega Customer Decision Hub provides support for directly outputting the results of a scheduled run to a configured Database Template rule. In this mode, action processing is bypassed.

Note: New deployments of Pega Customer Decision Hub on Pega Cloud services are configured to use the Decision Data Store for action state storage. Option to write Campaign results using a database template is disabled in this mode. If you want to enable the option to write Campaign results using a database template, see Disabling the Decision Data Store as action state storage on Pega Cloud.

To utilize this feature, the marketer can enable the **Write Campaign results using a database template** check box and specify the desired Database Template rule in the

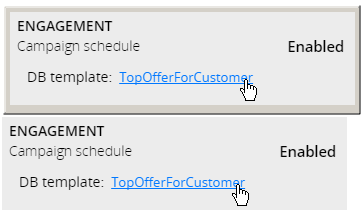
Template field.



Write Campaign results using a database template
Template *
TopOfferForCustomer

Note: This feature requires the Database Template rule to utilize the Append writing mode. The smart-prompt for the Template field only lists templates that utilize this mode.

After configuring the schedule, marketers can utilize the template link in the Engagement section or the right overview pane to open the Database Template rule.



ENGAGEMENT
Campaign schedule
Enabled
DB template: TopOfferForCustomer

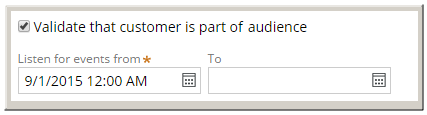
Configuring event-driven campaign execution

This engagement type represents event triggered execution of the Campaign. Refer to the Real-Time Events chapter for details on creating and triggering Events.

Pega Customer Decision Hub

The following configuration options are available for this engagement type:

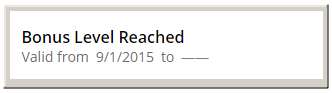
- **Validate that customer is part of audience** – If this is enabled, when a Campaign is triggered by an Event, the customer associated with the Event is validated against the configured audience on the Campaign. Only if the customer is present in the Campaign's audience, Campaign execution is triggered. **Note:** If the Campaign is referencing a Strategy rule that contains Segment references within the Strategy, the above setting is enforced even if it is not enabled. The customer must be a part of the Campaign audience in this scenario.
- **Listen for events from/to** – Specify the start and, if desired, end date and time values during which the Campaign will listen for the Events associated to it. Any Event that is received outside of the specified time frame will be ignored.



Validate that customer is part of audience
Listen for events from *
To
9/1/2015 12:00 AM

Users should use the **Add events** link to associate Real-Time Events with this Campaign . Clicking this link launches the **Configure Engagement** modal window which lists available Real-Time Event rules in the system. For more information about using this window, see the Using a Configuration Modal Window appendix.

A sample list entry for a Real-Time Event rule is shown below.

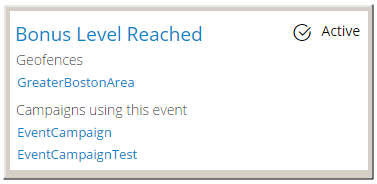


Bonus Level Reached
Valid from 9/1/2015 to ——

This view includes the following information about the Event rule:

- Friendly name / label
- Validity time period

Selecting this Event displays further information about it in the details panel.

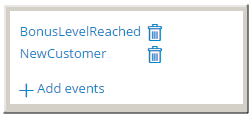


Bonus Level Reached
Active
Geofences
GreaterBostonArea
Campaigns using this event
EventCampaign
EventCampaignTest

This view displays the following information about the Event:

- Friendly name - Clicking the name opens the Real-Time Event rule.
- Whether the Event is active, i.e. available to respond to requests
- List of Geofences that the Event triggers. Clicking the name opens the Geofence rule.
- List of Campaigns that utilize this Event. Clicking the name opens the Campaign.

Clicking **Add Event** in the modal adds the selected Event to the list of Events associated with the Campaign.



BonusLevelReached
NewCustomer
Add events

Users can continue to add more Events to the Campaign following the process prescribed above. Clicking the delete icon next to an Event removes it from the Campaign .

Note: Volume Constraints and any Contact Policy limitations specified in the Strategy are not applicable during Event triggered Campaign execution.

The Configuring a Fallback Strategy section provides details on using a fallback Strategy with Real-Time Events.

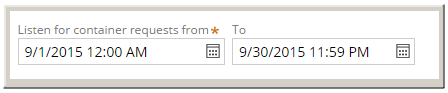
Exposing the results of a marketing campaign to real-time channels

This engagement type allows marketers to expose the results of a marketing Campaign to real-time channels such as web and mobile. Refer to the Real-Time Containers chapter for more details on creating and triggering Containers.

Pega Customer Decision Hub

The following configuration options are available for this engagement type:

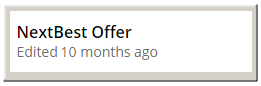
- **Listen for container requests from/to** – Specify the start and, if desired, end date and time values during which the Campaign will be valid for the specified Container. If a Container request is made outside of the specified time frame, this Campaign will not be considered for execution. Multiple Campaigns can reference a single Container as long as the active listening dates do not overlap.



Listen for container requests from*
To
9/1/2015 12:00 AM
9/30/2015 11:59 PM

Users should use the **Add containers** link to associate Real-Time Containers with this Campaign. Clicking this link launches the **Configure Engagement** modal window which lists available Real-Time Container rules in the system. For more information about using this window, see the Using a Configuration Modal Window appendix.

A sample list entry for a Real-Time Container rule is shown below.

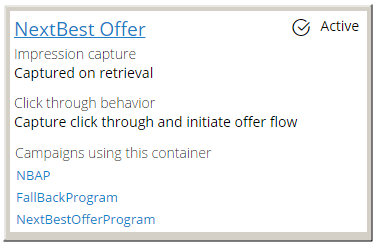


NextBest Offer
Edited 10 months ago

This view includes the following information about the Real-Time Container rule:

- Friendly name / label
- When the Container was last updated

Selecting this Container displays further information about it in the details panel.

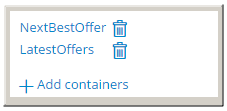


NextBest Offer
Active
Impression capture
Captured on retrieval
Click through behavior
Capture click through and initiate offer flow
Campaigns using this container
NBAP
FallBackProgram
NextBestOfferProgram

This view displays the following information about the Container:

- Friendly name - Clicking the name opens the Real-Time Container rule.
- Whether the Container is active, i.e. available to respond to requests
- Impression capture setting for the Container
- Click-through behavior setting for the Container
- List of Campaigns that utilize this Container. Clicking the name opens the Campaign.

Clicking **Add Container** in the modal adds the selected Container to the list of Containers associated with the Campaign.



NextBestOffer
LatestOffers

+ Add containers

Users can continue to add more Containers to the Campaign following the process prescribed above. Clicking the delete icon next to a Container removes it from the Campaign.

The Configuring a Fallback Strategy section below provides details on using a fallback Strategy with Real-Time Containers.

Configuring a Fallback Strategy

For Real-Time Events and Containers, define a fallback Strategy which gets executed if the Strategy defined on the Campaign fails. Save the FallbackStrategy Strategy rule into your implementation layer and replace the placeholder Set Property component in this Strategy with the action(s) that should be rendered if a Strategy fails.

Pega Customer Decision Hub

Setting volume constraints for a campaign

The **Constraint** section in the **Build** tab enables users to designate a Volume Constraint rule as the backing constraint for the Campaign.

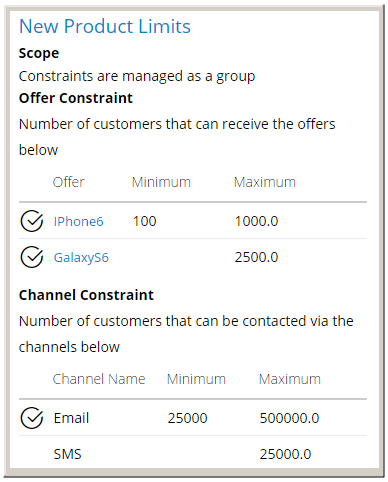
Pega Customer Decision Hub

Clicking the **Configure** button launches the **Configure** modal window which lists available Volume Constraints in the system. For more information about using this window, see the Using a Configuration Modal Window appendix.

In the list of entries, the following information is shown for each Volume Constraint:

- Friendly name / label
- Rule identifier
- When the Volume Constraint was last updated

Selecting a Volume Constraint displays further information about it in the details panel.



New Product Limits
Scope
Constraints are managed as a group
Offer Constraint
Number of customers that can receive the offers below
Offer
Minimum
Maximum
IPhone6
100
1000.0
GalaxyS6
2500.0
Channel Constraint
Number of customers that can be contacted via the channels below
Channel Name
Minimum
Maximum
Email
25000
500000.0
SMS
25000.0

This view displays the following information about the Volume Constraint:

- Friendly name - Click to open the Volume Constraint rule.
- Scope - Whether constraints defined in the rule are reset individually or as a group
- Grid of action constraints - Includes action name, minimum, and maximum volumes. Click the action name to open the action rule. A check mark next to a row in the grid indicates that the corresponding constraint is enabled.
- Grid of channel constraints - Includes channel name, minimum, and maximum volumes. A check mark next to a row in the grid indicates that the corresponding constraint is enabled.

Campaign Approval

Pega Customer Decision Hub provides supplementary support for incorporating an approval process into the Campaign lifecycle. This functionality is included in the Marketing Operations module along with other operational features.

Pega Customer Decision Hub

As part of this functionality, Campaigns must be approved before they can be launched. Application implementers can customize the criteria for Campaigns to require approval. Advanced implementations may even consider customizing the approval flow to better suit their needs.

In the out-of-the-box process, users can perform the following actions:

- Submit Campaign for approval
- Recall submitted Campaign

The manager can perform the following action:

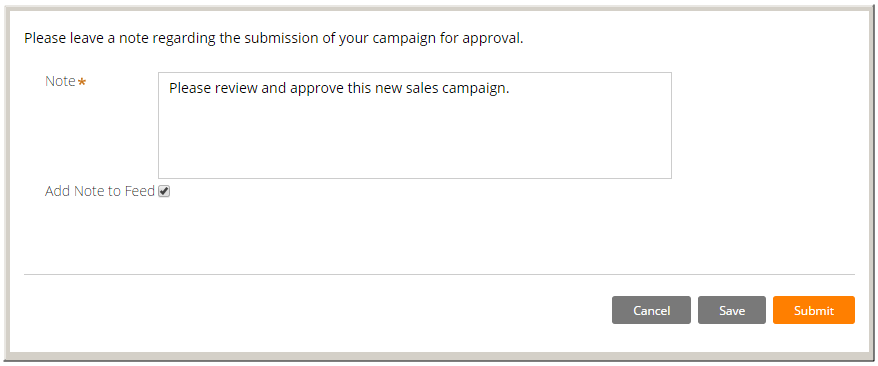
- Approve or Reject a Campaign

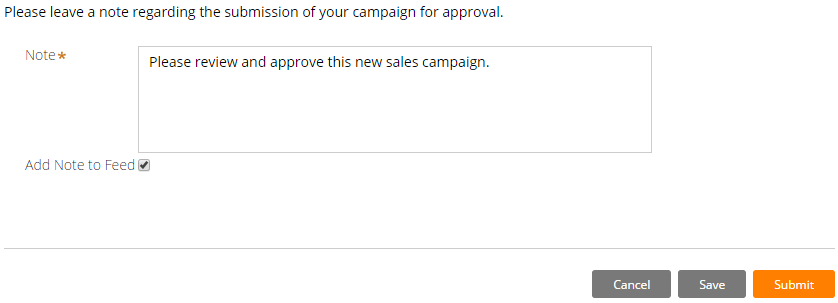
- Submitting a Campaign for Approval
- Recalling a Submitted Campaign
- Approving and Rejecting Campaigns

Submitting a Campaign for Approval

The **Submit for Approval** action enables the user to submit their Campaigns for approval. Most Campaigns require approval before they can be executed. Users can enter a note along with their approval request.

Pega Customer Decision Hub





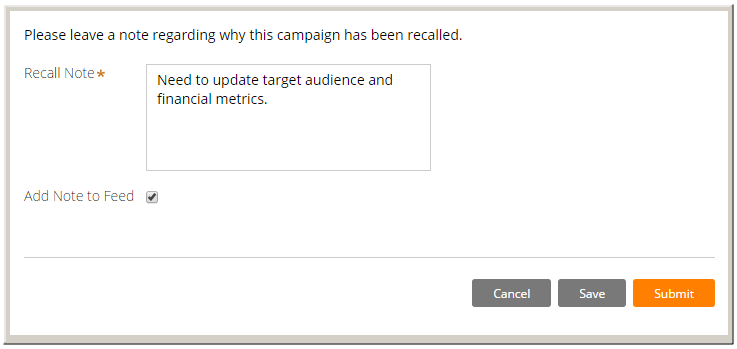
The system automatically routes approval requests to the manager of the user's organizational unit. In addition to adding the Campaign to the manager's Work List, the system also notifies the manager about the approval request via email. This email contains salient Campaign details, the marketer's note, and a link to directly open the Campaign.

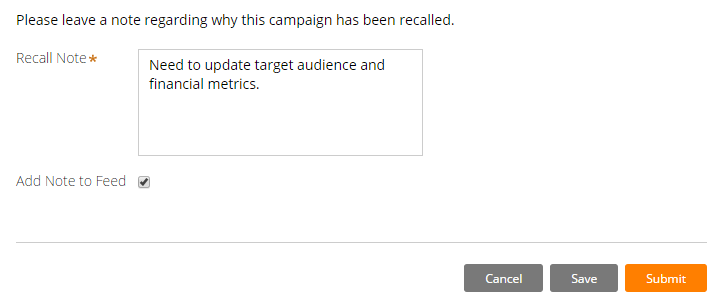
Recalling a Submitted Campaign

After a Campaign has been submitted for approval, no further changes are allowed to the Campaign. If a user wishes to make further updates to the Campaign, they must first recall the Campaign from the manager via the **Recall** action.

Pega Customer Decision Hub

Users can enter a note explaining the reason for recalling the Campaign from the manager.





As before, the manager is notified via email when a Campaign is recalled from their Work List. This email contains the recall note and a link to directly open the Campaign.

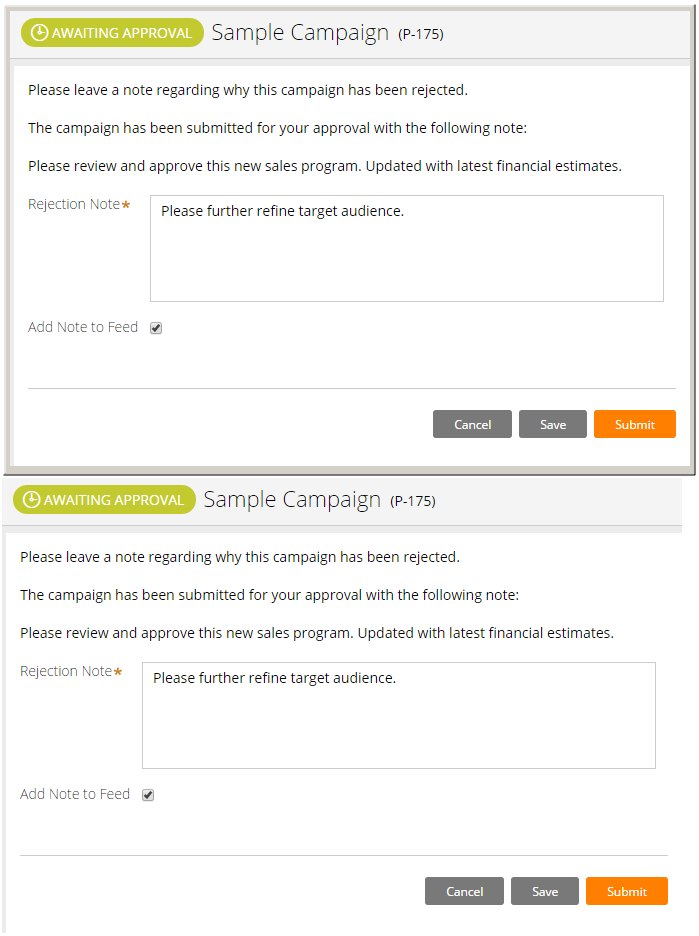
Once a Campaign has been successfully recalled, it can be updated and re-submitted for approval when ready.

Approving and Rejecting Campaigns

To approve or reject a Campaign, a manager can open the Campaign either directly from the Campaign link in an email or from their Work List. The **Approve** and **Reject** actions enable the manager to take the corresponding approval action once they have reviewed the Campaign details.

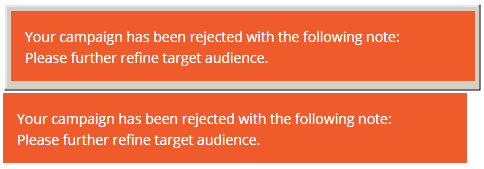
Pega Customer Decision Hub

When approving the Campaign, the manager can also enter an approval note. A note is required if the manager is rejecting the Campaign.



After the manager performs an approval action on the Campaign, the system notifies the user that submitted the Campaign for approval. This email contains the approval status, manager's note, and a link to directly open the Campaign.

The approval status and the manager's note are also displayed in the Campaign.

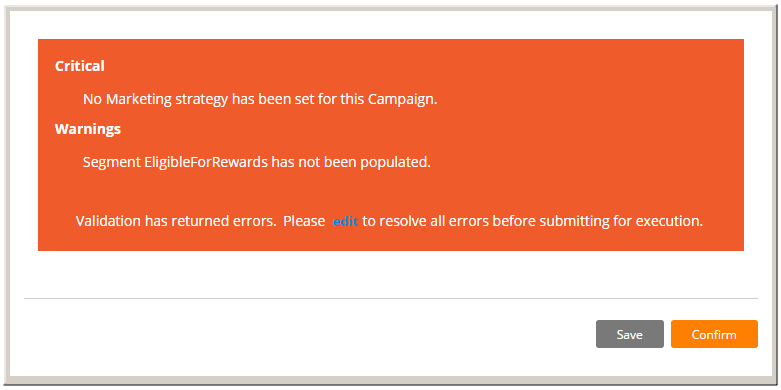


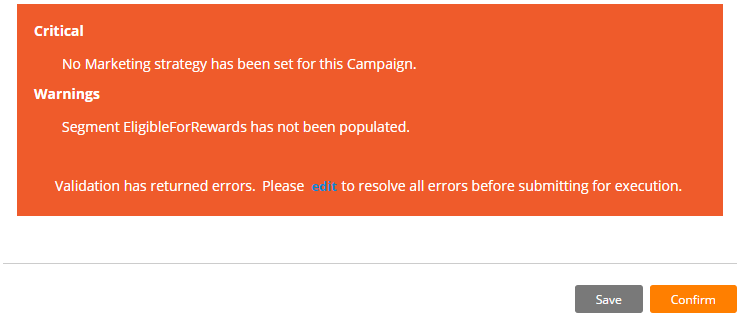
Note: When a Campaign is rejected, the marketer must make updates to its configurations (via the **Edit** action) before other actions can be performed on the Campaign.

Campaign Validation

While the Campaign is in the design stage, click **Actions Validate** to validate your Campaign configuration. Reported violations are categorized into critical items and warnings. Critical items will prevent the Campaign from being launched. Warnings represent best practice violations and other scenarios which may impact Campaign execution.

Pega Customer Decision Hub





The user can use the **edit** link to make any necessary updates to the Campaign.

The system automatically invokes this same validation routine when the Campaign is tested or run. If the validation returns any critical items, the user must correct these in order to perform the desired action. Warnings do not prevent these actions from being taken but should still be individually heeded and addressed, where necessary.

Campaign Testing

After configuring the Campaign, use the **Test** tab on the Campaign to test the configuration and review test results.

Pega Customer Decision Hub

Pega Customer Decision Hub supports two types of Campaign testing:

- Distribution Test
- Seed Test

Note: The above test modes are disabled on Campaigns that utilize the direct output to template feature.

- Distribution Test
- Seed Test

Distribution Test

A distribution test enables the user to test the execution of a Campaign in terms of assessing the actions made and the customers targeted. The result of a distribution test is a series of reports outlining various distribution metrics. No actions are actually made to customers in this form of Campaign testing.

Pega Customer Decision Hub

The **Distribution test** link in the **Tests** section enables the user to initiate a distribution test. This link is enabled when the following conditions are met:

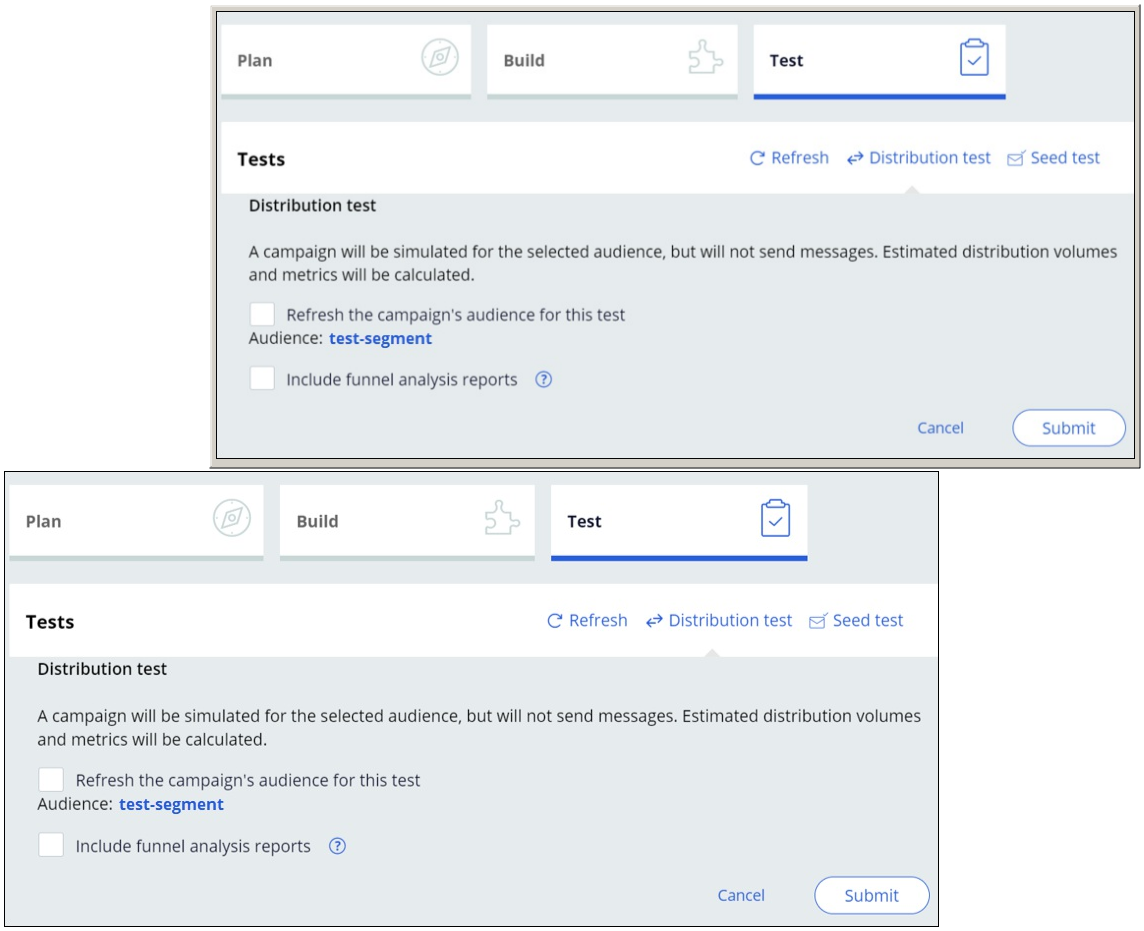
- Strategy and Strategy result component is selected
- At least one engagement is configured
- Audience is configured
- Campaign is not utilizing direct output of results using a database template

Clicking this link performs Campaign validation and presents the user with validation violations and configuration options. The user must resolve all critical violations before they can initiate the distribution test.

Additionally, the user can enable the following options for the distribution test:

- Refresh the Campaign audience before executing the test
- Apply the Campaign Constraint as part of executing the test

Campaign distribution test configuration



Upon clicking **Submit**, the system initiates a distribution test. An entry for this test is added to the test runs grid. Users can refresh the Campaign display to view the current status of the test.

After the test run finishes, click a test row to display various details about the distribution test.

The delete icon in the row enables users to remove the corresponding distribution test from the Campaign.

Funnel analysis

The **Include funnel analysis reports** option is deprecated. To assess how certain components and expressions influence the overall outcome, go to the Simulation Testing landing page and create a decision funnel test. This type of simulation test provides drill-down reports of proposition counts by Champion Challenger, Proposition Filter, Prioritization and Switch components, and optionally, by Geofence, Segment Filter, and Contact Policy components.

Seed Test

Use a seed test to configure and initiate a simulation of Campaign execution.

Pega Customer Decision Hub

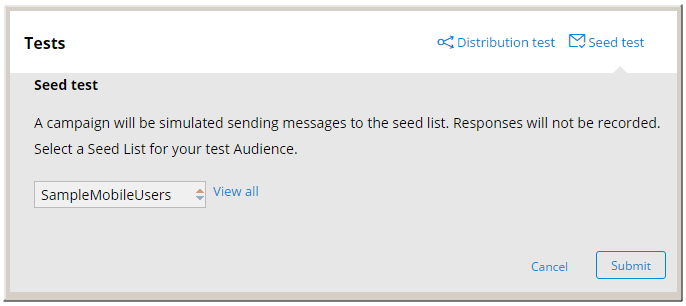
This test is run against customers specified in a Seed List. The test executes the referenced Strategy to determine the actions that should be made to the customers (seeds), and then makes these actions. Customers are able to respond to these actions. These responses, however, are not captured in Interaction History.

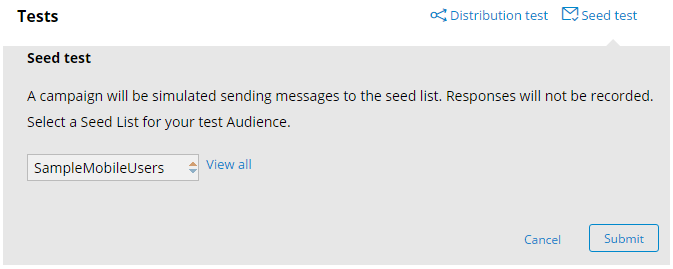
The **Seed test** link in the **Tests** section enables the user to initiate a seed test. This link is enabled when the following conditions are met:

- Strategy and Strategy result component is selected
- Campaign isn't utilizing direct output of results using a database template

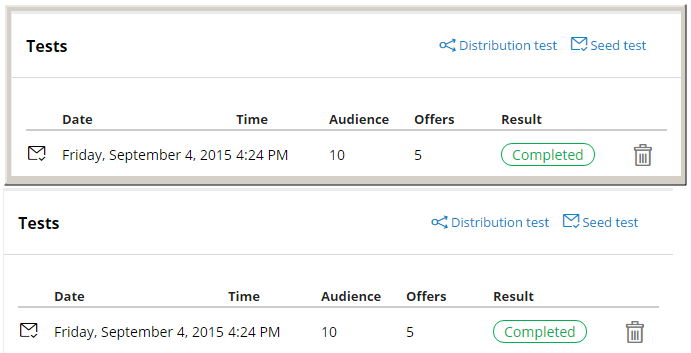
Clicking this link performs Campaign validation and presents the user with validation violations and configuration options. The user must resolve all critical violations before they can initiate the seed test.

For a seed test, the user must simply select a Seed List to represent the audience for the Campaign. Refer to the Seed Lists chapter for information on creating and configuring Seed Lists.





Upon clicking **Submit**, the system initiates a seed test. An entry for this test is added to the test runs grid. Users can refresh the Campaign display to view the current status of the test.



Clicking the test row displays various details about the seed test. Refer to Viewing Run Details for information on using this view.

The delete icon in the row enables users to remove the corresponding seed test from the Campaign.

Campaign Execution

This section explains the following areas pertaining to Campaign execution:

Pega Customer Decision Hub

- Running a Campaign
- Rescheduling a Campaign
- Suspending a Campaign
- Resuming a Campaign
- Stopping all runs
- Stopping a run
- Restarting a run
- Skipping a run
- Campaign execution cycle
- Managing Concurrent Campaign Data Flow Runs

- Running an Outbound Campaign
- Rescheduling an Outbound Campaign
- Suspending an Outbound Campaign
- Resuming a Campaign
- Stopping All Runs
- Stopping a Run
- Restarting a Run
- Skipping a Run
- Campaign Execution Cycle
- Managing Concurrent Campaign Data Flow Runs

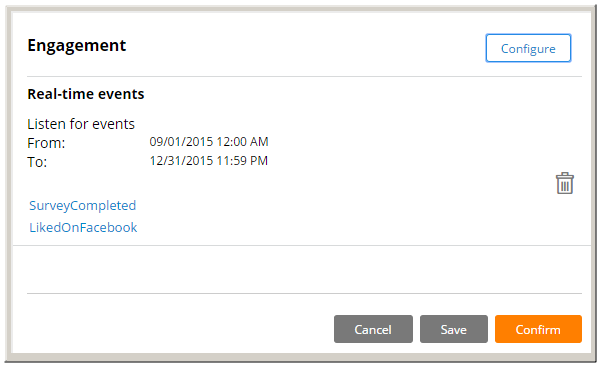
Running a Campaign

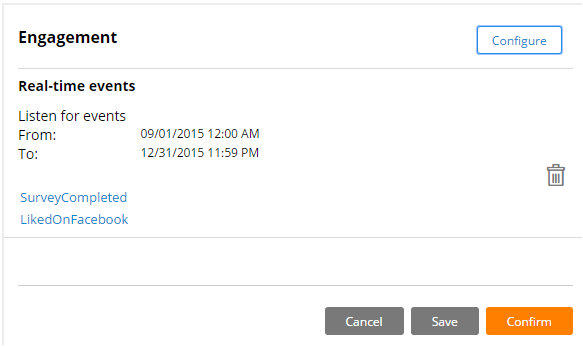
After the Campaign has been configured and approved, it is ready to be launched. This readiness is indicated by the availability of the **Run** action.

Pega Customer Decision Hub

Clicking the **Run** button or selecting the **Submit for Execution** action from the **Actions** menu performs Campaign validation and presents the user with any reported validation violations and run configuration options. The user must resolve all critical violations before the Campaign can be launched.

From this screen, the user can configure the desired engagement types for the Campaign . The options available in this screen are explained in the Configuring Campaign Engagement section. The user must select at least one engagement type in order to launch the Campaign.





After configuring the desired engagement type(s) for the Campaign, users can click **Confirm** to launch the Campaign. Upon doing so, the system generates artifacts necessary for Campaign execution and transitions the Campaign to the running stage.

After submitting a Campaign for execution (and while it is still running), its referenced components (such as Segments, Strategies, and Volume Constraints) should not be deleted. In addition, certain structural changes should only be made after suspending the Campaign. Examples of these changes include modifying the Strategy result component and adding Segment usage within Strategies. After making the desired structural change, the Campaign must be updated and resubmitted for execution.

Rescheduling a Campaign

Use the **Reschedule** action (in the **Actions** menu) to modify the schedule of a running Campaign. The new schedule applies only to upcoming runs and does not impact any runs that are currently running. All upcoming runs for the Campaign are suspended while this action is in progress.

Pega Customer Decision Hub

The configuration options available for the Reschedule action are explained in the Configuring Campaign Engagement section. However, during this action the user can only make updates to schedule-related configurations. They can not enable or disable engagement types, or modify associated Events or Containers. To make such changes, the user must first suspend the Campaign.

Once the user completes the Reschedule action, the previous set of upcoming runs is removed and a new set of upcoming runs is generated based on the newly specified schedule.

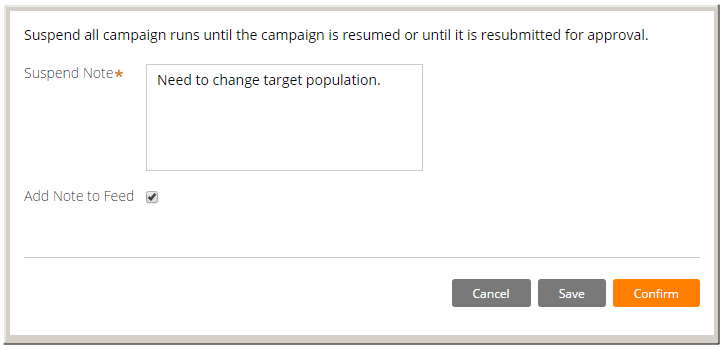
The system also notifies the user who submitted this Campaign for execution. The notification email contains a link to the Campaign and relevant schedule information, such as the next scheduled run time.

Suspending a Campaign

Use the **Suspend** action (from the **Actions** menu) if they need to suspend a running Campaign, for example, to make modifications to the Campaign configuration, such as its audience or Strategy.

Pega Customer Decision Hub

Suspending a Campaign prevents any upcoming runs from executing. When performing this action, the marketer must enter a note.



Suspend all campaign runs until the campaign is resumed or until it is resubmitted for approval.
Suspend Note*
Need to change target population.
Add Note to Feed
Cancel
Save
Confirm

Upon completion of this action, the system transitions the Campaign back to the design stage and notifies the user who launched the Campaign that it has been suspended. The notification email contains the marketer's note and a link to open the Campaign.

Resuming a Campaign

Use the **Resume** action (from the **Actions** menu) to resume a suspended Campaign. This action is only available if the marketer hasn't modified the Campaign after it was suspended.

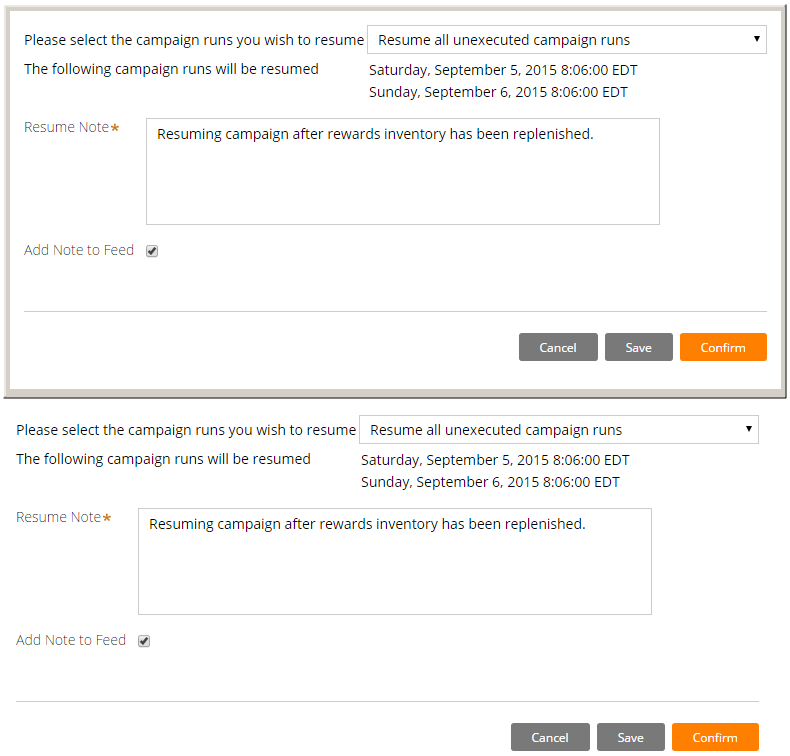
Pega Customer Decision Hub

When performing this action, the marketer can determine what the system should do to scheduled runs.

- **Resume all unexecuted campaign runs** - In this option, all runs that were not executed will be resumed. This will also include all runs that are now in the past. Campaign execution will start with the earliest run (that was not executed) and continue until it has caught up with the current time. This implies that if there are multiple runs in the past that were not executed, they will execute one after the other.

- **Remove all past unexecuted campaign runs** - In this option, any run that is not executed but in the past will be removed. This option basically provides a mechanism for ignoring (skipping over) runs that are in the past and resuming Campaign execution with the next run, i.e. one that is scheduled to execute on or after today.
- **Remove all past unexecuted campaign runs except the last** - In this option, all past runs that were not executed will be removed with the exception of the last one. This option will execute the last run immediately on submission and then follow the rest of the upcoming Campaign schedule.

Regardless of the option selected above, the system displays the list of runs that will be resumed so that the marketer can see the impact the various resumption options will have on Campaign execution. The marketer can enter a note when they resume the Campaign.



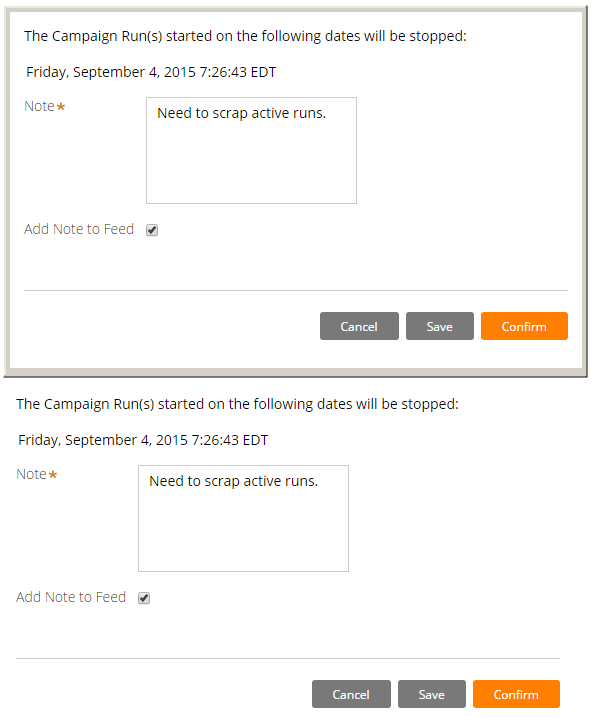
Upon submission of this action, the system transitions the Campaign back to the running stage and notifies the user who launches this Campaign for execution that it has been resumed. The notification email contains a link to open the Campaign.

Stopping All Runs

The **Stop all runs** action (in the **Actions** menu) enables marketers to stop all active runs of the Campaign.

Pega Customer Decision Hub

Upon initiating this action, the system presents the user with a list of all runs that will be stopped. The user must enter a note detailing why the stop action was selected.



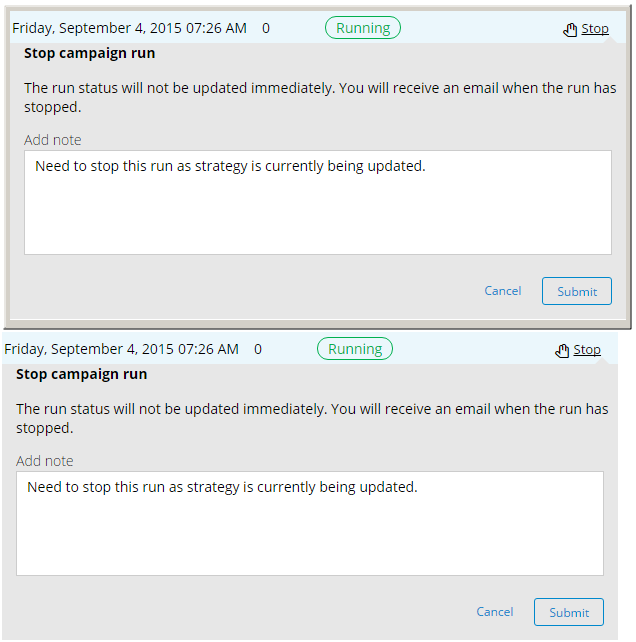
Upon submission of this action, the system issues stop requests for each of the listed runs. The system will halt run processing (if possible) at the earliest possible instance. When the system successfully stops a run, a notification email is sent to the user who submitted the Campaign for execution.

Stopped runs can then be individually restarted or skipped by selecting the appropriate action from the runs grid in the Campaign display.

Stopping a Run

At times, marketers may wish to stop an individual Campaign run. To do so, they can utilize the **Stop** link in the row for the desired run in the Run schedule section. The marketer can also enter a note along with the stop request.

Pega Customer Decision Hub



Upon submission of this action, the system will attempt to halt run processing at the earliest possible instance. Once the system successfully stops the run, a notification email is sent to the user who submitted the run's Campaign for execution. The system also assigns the run to this user by adding it to their work list. The user must act on this stopped run, i.e. perform either restart or skip, in order for the Campaign to be eligible for wrap-up and completion.

Restarting a Run

If a campaign run stops or fails, you can restart it after you solve the cause of the previous run failure.

Note: New deployments of Pega Customer Decision Hub on Pega Cloud services are configured to use the Decision Data Store for action state storage. Campaign Run restarts are disabled in this mode. If you want to enable the campaign run restarts feature, see Disabling the Decision Data Store as action state storage on Pega Cloud.
Pega Customer Decision Hub

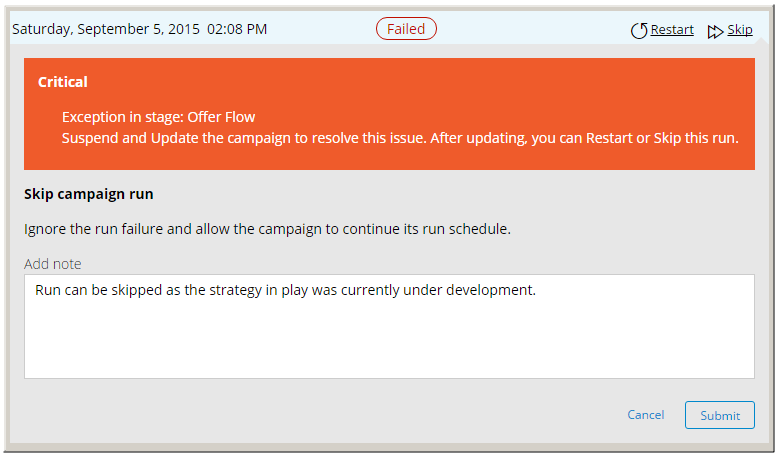
1. Review and address any errors reported for the previous campaign run.
2. In the **Run schedule** section, click **Restart** for the run which you want to restart.
3. **Optional:** Enter an explanation note along with the restart request.

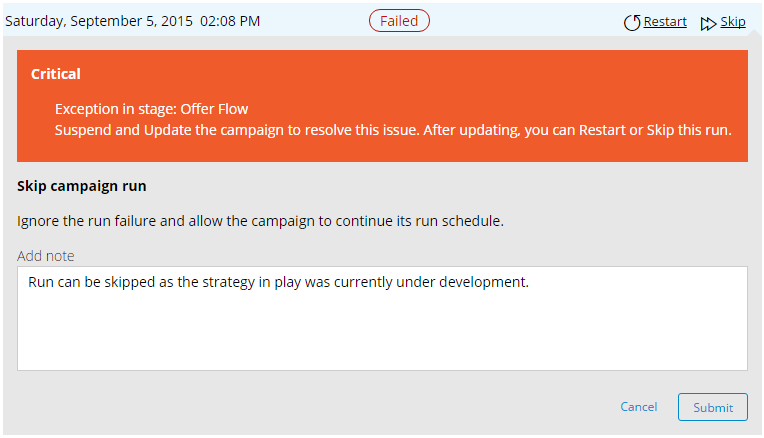
Result: The failed run is skipped and a new run is initiated. The new run only sends actions to customers whose actions were not initiated during the failed run, to avoid restarting the campaign from the beginning.

Skipping a Run

The Skip action enables marketers to ignore a failed or stopped run. To perform this action, marketers can utilize the **Skip** link in the row for the desired run in the **Run schedule** section. The marketer can also enter a note along with the skip request.

Pega Customer Decision Hub





Upon submission of this action, the run is initially displayed in the run schedule with a skipped status. When the user refreshes (or reopens) the Campaign, the skipped run will no longer be displayed in the run schedule section.

As part of this action, the system also notifies the user via email that the run has been skipped.

Campaign Execution Cycle

The life cycle of the Campaign execution (i.e. running) stage depends on the individual execution cycles for each of its configured engagement types. Once each of these types have completed their execution cycles, the Campaign moves to the wrap-up stage.

Pega Customer Decision Hub

Scheduled execution

The system generates the execution schedule for the Campaign and then waits for the execution time of the first scheduled run. As a run completes, the system schedules the next run and repeats this process until all runs are complete.

For Campaigns where the end date is not specified, the system generates the upcoming schedule for the first ten runs. It will add the next set of runs to the upcoming schedule when there are fewer than five runs remaining. To configure the number of runs the system generates at a time, click **Settings** in the Pega Express navigation bar, then click **Application Artifacts** .

Once all scheduled runs of a Campaign are complete, the scheduled execution cycle is also considered to be complete.

Event-enabled execution

The system listens for configured Events during the configured time frame. If an end time is configured for the window, the Campaign stops listening for the Event once the end time is reached. At this point, the event execution cycle of the Campaign is considered to be complete.

Container-enabled execution

The system responds to Container requests during the configured time frame. If an end time is configured, the Container will no longer consider this Campaign once the end time is reached. At this point, the container execution cycle of the Campaign is considered to be complete.

Managing Concurrent Campaign Data Flow Runs

To optimize performance and stability of your Pega Customer Decision Hub application, the number of campaign data flows which run concurrently is limited. You can adjust the number of concurrent data flows and the retry interval by editing the related Dynamic System Settings.

Pega Customer Decision Hub
Note: The Dynamic System Settings listed below apply to all outbound, multi-channel, and self-optimized campaigns defined in your application.

- **MKTMaxAllowedBatchDFs** - Edit this setting to change how many campaign data flows can run concurrently. By default, ten runs can be active at the same time, and any additional data flows are paused for the duration defined in the setting **MKTDFRetryInterval**.
- **MKTDFRetryInterval** - Edit this setting to change how soon Pega Customer Decision Hub will attempt to execute a data flow which was paused because it exceeded the number of data flows defined in **MKTMaxAllowedBatchDFs**. By default, the retry interval is 30 minutes. After the inverval passes, Pega Customer Decision Hub checks the data flow availability again, for the number of times defined in the setting **MKTDFNumRetries**.
- **MKTDFNumRetries** - Edit this setting to change how many times Pega Customer Decision Hub will attempt to execute a data flow which was paused because it exceeded the number of data flows defined in **MKTMaxAllowedBatchDFs**. By default, the number of retries is 5. After the specified number of retries, the campaign run fails and an email notification is sent to the operator who triggered the run.

Campaign Monitoring

This section includes the following topics pertaining to Campaign monitoring:

Pega Customer Decision Hub

- Monitoring Campaign Performance
- Viewing Run Details

- Monitoring Campaign Performance
- Viewing Run Details

Monitoring Campaign Performance

Once a Campaign is in its running stage, the Campaign display switches to the monitoring view. This view includes the following sections which aid marketers in monitoring various aspects of their Campaigns.

Pega Customer Decision Hub

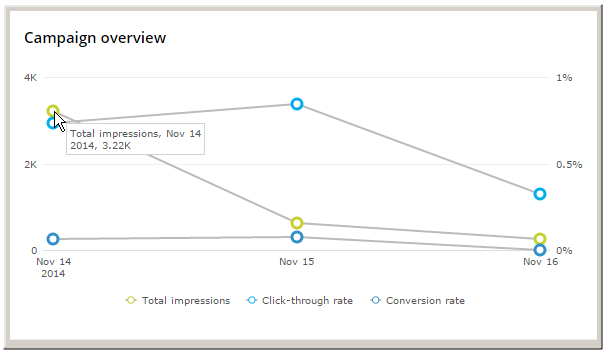
You can easily monitor the status of your running and scheduled campaigns with the Campaign Runs widget. Add the widget to the Dashboard and use it to quickly identify Campaigns that need attention. For more information about Dashboard widgets, see Customizing the Pega Customer Decision Hub dashboard.

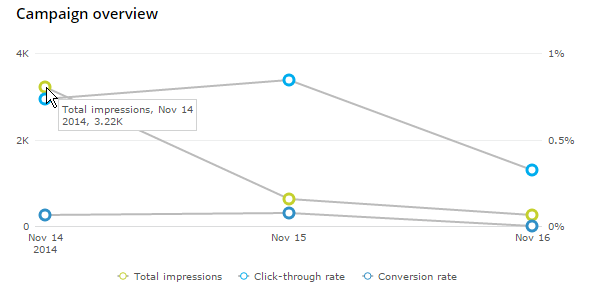
- Campaign overview
- Performance
- Run schedule
- Goals

- Campaign overview

- Performance
- Run schedule
- Goals

Campaign overview





Pega Customer Decision Hub

This section provides a visual representation of how the Campaign is performing since the start of its execution across the following salient metrics:

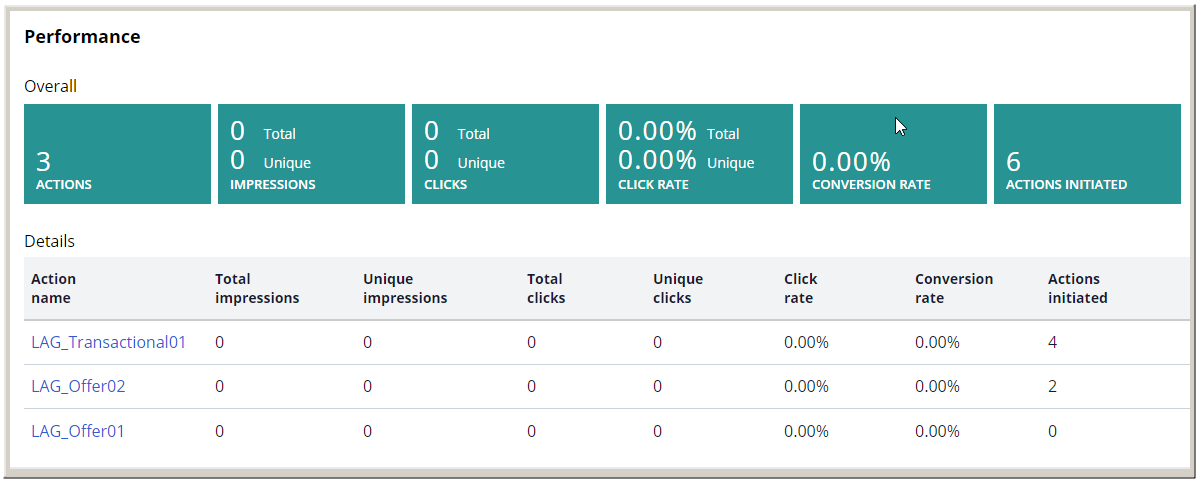
- Total impressions
- Click-through rate
- Conversion rate

The key usage aspects of the chart in this section are:

- The x-axis represents time from the start of Campaign execution till its end, or till the current day if the Campaign is still running.
- The left y-axis corresponds to the impressions metric, while the right y-axis corresponds to the two rate metrics.
- Hovering over a data point in the chart displays details about the corresponding metric value.
- Clicking the metric name in the legend toggles the display of the metric's line chart.

Performance

Pega Customer Decision Hub



Action name	Total impressions	Unique impressions	Total clicks	Unique clicks	Click rate	Conversion rate	Actions initiated
LAG_Transactional01	0	0	0	0	0.00%	0.00%	4
LAG_Offer02	0	0	0	0	0.00%	0.00%	2
LAG_Offer01	0	0	0	0	0.00%	0.00%	0

Performance

Overall

3 ACTIONS	0 Total 0 Unique IMPRESSIONS	0 Total 0 Unique CLICKS	0.00% Total 0.00% Unique CLICK RATE	0.00% CONVERSION RATE	6 ACTIONS INITIATED

Details

Action name	Total impressions	Unique impressions	Total clicks	Unique clicks	Click rate	Conversion rate	Actions initiated
LAG_Transactional01	0	0	0	0	0.00%	0.00%	4
LAG_Offer02	0	0	0	0	0.00%	0.00%	2
LAG_Offer01	0	0	0	0	0.00%	0.00%	0

This section lists both cumulative and individual metrics pertaining to the Campaign's overall performance.

The metrics displayed in the boxes are cumulative across the execution cycle of the Campaign. The following metrics are displayed:

- **Actions** - Number of actions used during the execution of this Campaign.
- **Total impressions** - Total number of impressions for this Campaign. The total impressions count can include multiple impressions from the same individual. For example, if an individual opens the same email treatment three times, three impressions are added to the total count.
- **Unique impressions** - Number of unique impressions. The unique impressions count only includes one impression per individual. For example, if an individual opens the same email treatment three times, only one impression is added to the unique impressions count.
- **Total clicks** - Overall click-through rate of Total number of clicks for this Campaign. The total clicks count can include multiple clicks from the same individual. For example, if an individual opens the same link three times, three clicks are added to the total count.
- **Unique clicks** - Number of unique clicks. The unique clicks count only includes one impression per individual. For example, if an individual opens the same link three times, only one click is added to the unique click count.
- **Total click rate** - Overall click-through rate of this Campaign.
- **Unique click rate** - Click-through rate for unique clicks only.
- **Conversion rate** - Overall conversion rate of this Campaign.
- **Actions initiated** - Total number of actions initiated during Campaign execution.

The **Details** sub-section below the boxes presents a drill-down view of the above metrics for each action that was utilized by the Campaign. Clicking the action name opens the corresponding action rule. The following details are available:

- **Action name** - Name of the action.
- **Total impressions** - Total number of impressions for this action. The total impressions count can include multiple impressions from the same individual. For example, if an individual opens the same email treatment three times, three impressions are added to the total count.
- **Unique impressions** - Number of unique impressions. The unique impressions count only includes one impression per individual. For example, if an individual opens the same email treatment three times, only one impression is added to the unique impressions count.
- **Total clicks** - Overall click-through rate of Total number of clicks for this action. The total clicks count can include multiple clicks from the same individual. For example, if an individual opens the same link three times, three clicks are added to the total count.
- **Unique clicks** - Number of unique clicks. The unique clicks count only includes one impression per individual. For example, if an individual opens the same link three times, only one click is added to the unique click count.
- **Click rate** - Overall click-through rate of this action.
- **Conversion rate** - Overall conversion rate of this action.
- **Actions initiated** - Total number of actions initiated from this action.

Run schedule

In the monitoring view, the **Run schedule** section displays two grids - one for initiated runs and the other for upcoming runs.

Pega Customer Decision Hub

Initiated runs

This grid lists each initiated Campaign run with the most recent run at the top. Each run row in the grid contains the following:

- Status of the run - typical values are Scheduled, Running, Completed, Failed
- Date and time when the run started its execution
- Number of actions initiated by the run
- Links for available actions
  - When running, available action is Stop
  - If failed, available actions are Restart and Skip

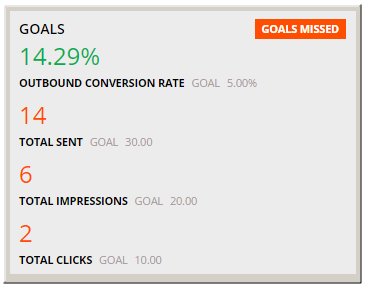
Additionally, users can click the run's row to see a detailed view of the run. Refer to Viewing Run Details for information on using this view.

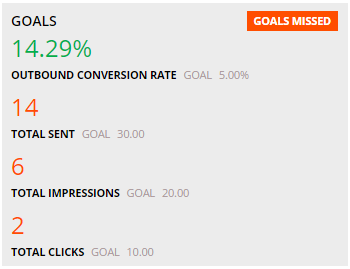
Upcoming runs

The **Upcoming** grid displays the date and time of upcoming runs for the Campaign.

Goals

Pega Customer Decision Hub





This section displays the performance of the Campaign against the goals configured by the marketer.

At a high level, the overall goal achievement status for the Campaign can have one of the following values:

- Goals Met - When at least half of all configured goals have been met
- Goals Missed - When less than half of configured goals are met (as shown above)
- Goals Not Set - When no goals have been configured for the Campaign

For each configured Campaign goal, the system shows current value and the target goal value. The current value displays in green if the corresponding goal has been met.

Viewing Run Details

Clicking a row for an initiated Campaign run, distribution test, or seed test presents an expanded view which contains details about the specific run. Users can click the row again or click **Close** to collapse this view and return to the grid view.

Pega Customer Decision Hub

To begin with, the section provides an overview of the execution stages and the progress of the run through these stages. This view contains the following four boxes:

- **Audience** - This section displays the number of customers that make up the run's audience.
- **Strategy** - This section displays the number of actions that were identified by the Campaign's Strategy.
- **Initiated** - This section displays the number of actions that were actually initiated by the system.
- **Result** - This section shows the overall result of the run (or test).

The Audience, Strategy, and Initiated sections also include the date and time when the corresponding stage was completed.

Additionally, the color of the status badge visually indicates the status of the corresponding stage.

- Green - Indicates that the run is progressing without any problems, or that it was completed successfully.
- Orange - Indicates a potential issue. For example, the run has completed, but with action failures, or the campaign is stopped, but the run is still active.
- Red - Indicates that the run has failed.

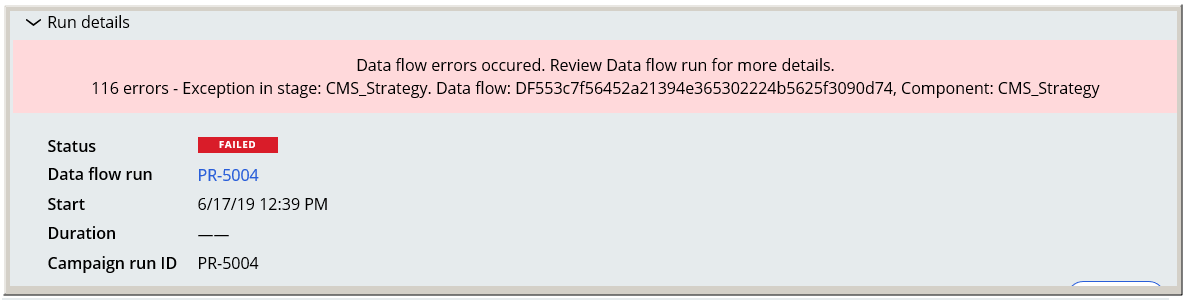
The Actions grid displays the list of distinct actions initiated as part of the run (or test) along with their instantiated counts. Clicking an action name in the grid opens the corresponding action rule.

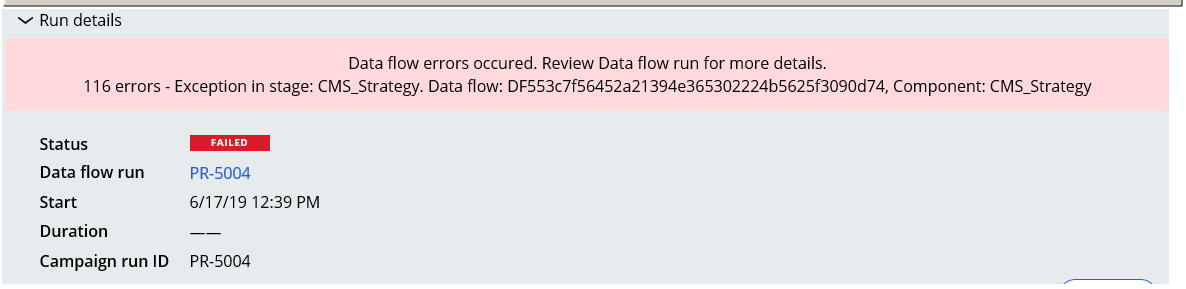
When the run includes action bundles, there are additional rows in the grid for bundled members and the **Bundle member count** column displays the count of each initiated member.

Clicking the **Reports** link at the bottom displays the distribution reports that were generated as part of the run (or test).

Tip: You can add a Campaign Runs widget to your Pega Customer Decision Hub portal home page. The widget displays all current run details. Use it to monitor and plan campaign execution.

Clicking the **Run details** link at the bottom displays details about the campaign data flow. If any errors occurred in the data flow, they are displayed on top of the **Run details** section. For easier troubleshooting, the **Run details** section also contains a link to the data flow work object. To open the data flow, click the flow ID in the **Data flow run** field, and then open the data flow from the data flow work object harness.





Campaign Completion

This section includes the following topics pertaining to Campaign completion:

Pega Customer Decision Hub

- Wrapping up a Campaign
- Archiving a Campaign
- Restoring a Campaign
- Reopening a Campaign

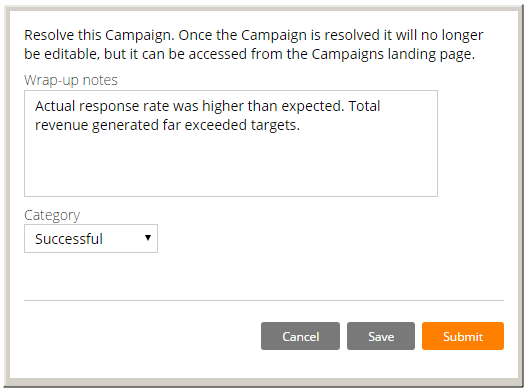
- Wrapping up an Outbound Campaign
- Archiving a Campaign
- Restoring a Campaign
- Reopening a Campaign

Wrapping up a Campaign

When a Campaign completes its execution cycle, it moves to the wrap-up stage. In this stage, the marketers can utilize the **Wrap-up** button or the **Wrap-up** action (in the **Actions** menu) to complete the Campaign.

Pega Customer Decision Hub

As part of this action, the marketer can enter any additional remarks on the Campaign and classify it based on its outcome into one of four categories: Successful, Not successful, Neutral, or Not specified.



Resolve this Campaign. Once the Campaign is resolved it will no longer be editable, but it can be accessed from the Campaigns landing page.
Wrap-up notes
Actual response rate was higher than expected. Total revenue generated far exceeded targets.
Category
Successful
Cancel
Save
Submit

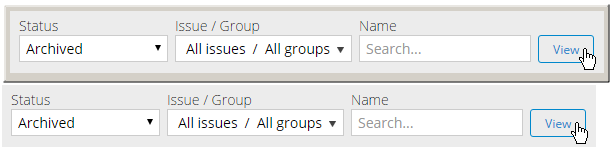
Upon submission of this action, the Campaign transitions to the completed stage. Users can no longer make any edits to this Campaign.

Archiving a Campaign

Once a Campaign is complete, the marketer can use the **Archive** button or the **Archive** action (in the **Actions** menu) to archive the Campaign. Once the Campaign is archived, it will not appear (by default) in the list of Campaigns on the Campaigns landing page.

Pega Customer Decision Hub

To view archived Campaigns in the Campaigns landing page, the marketer should set the **Status** filter to **Archived** and then click **View**.



Restoring a Campaign

The **Restore** action (in the **Actions** menu) enables the marketer to move an archived Campaign back to the stage it was in before being archived, i.e. either the completed or the wrap-up stage. Once the Campaign is restored, it will be visible in the list of Campaigns on the Campaigns landing page.

Pega Customer Decision Hub

Reopening a Campaign

The **Reopen** action (in the **Actions** menu) enables the marketer to revert a completed Campaign back to the wrap-up stage.

Pega Customer Decision Hub

Outbound Campaigns

For advanced marketing initiatives, Multi-Channel Campaigns allow the use of adaptive and predictive analytics with business rules to optimize across multiple actions, resulting in the selection of the best action for each individual customer. However, marketers often need a simpler mechanism to promote a single offering and monitor its performance.

Pega Customer Decision Hub

The Outbound Campaign construct provides just this functionality using a visual card paradigm that highlights all aspects of the initiative on a single canvas. This allows marketers to focus on key aspects, such as the audience, the action, and the flow.

High-level features of the Outbound Campaign functionality include the following:

- Simplified canvas, which includes all details in single view
- Visual ‘card’ paradigm for easier building

- Traceable goals and the ability to monitor their performance

Outbound Campaigns have a lifecycle of their own and go through various stages. These stages are outlined in the Outbound Campaign Lifecycle section. The remaining sections in this chapter outline the various interactions available to the user. These interactions include the following:

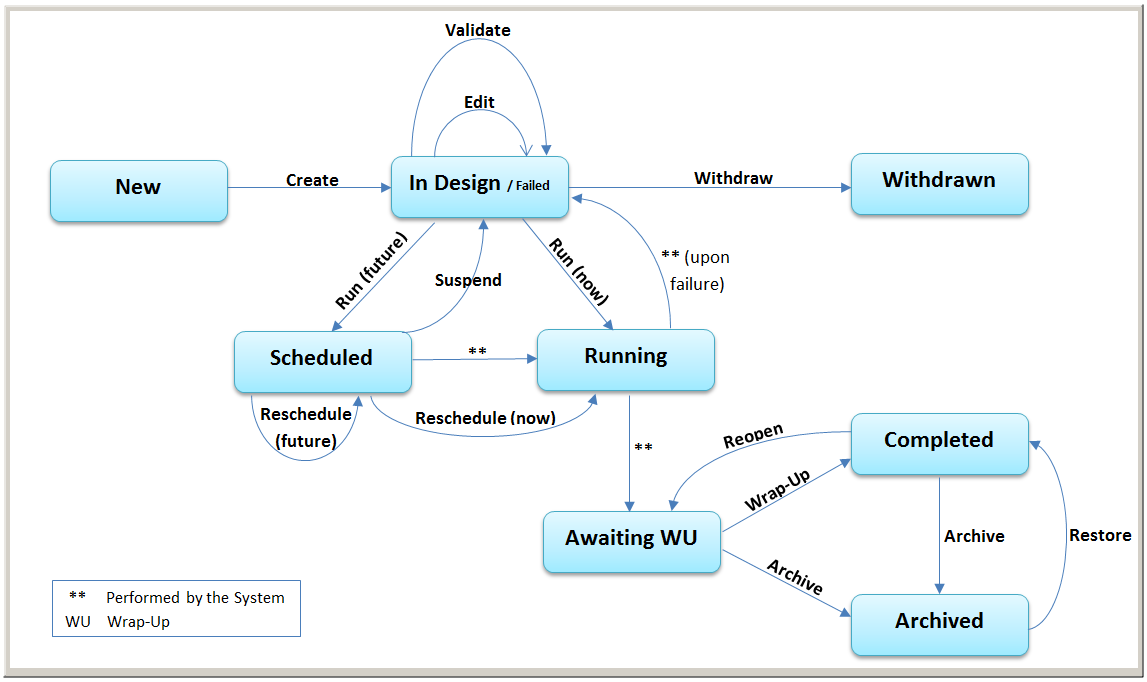
- Outbound Campaign Lifecycle
- Creating a traditional outbound marketing campaign
- Campaign Configuration
- Validating an Outbound Campaign
- Running a Campaign
- Monitoring Outbound Campaign Goals
- Making changes to a running campaign

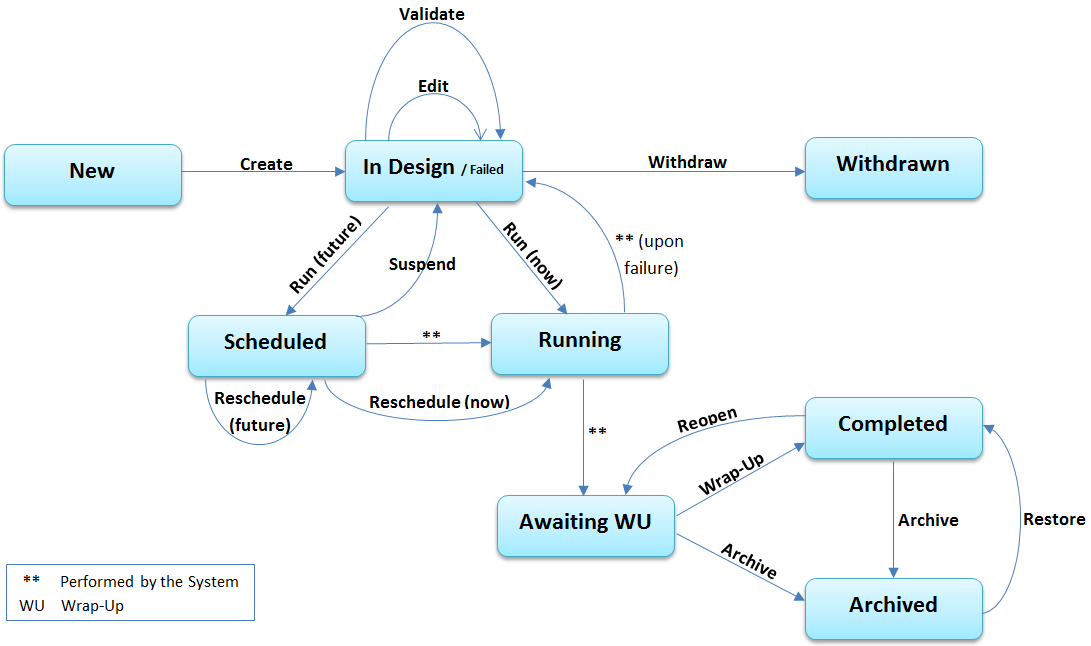
Outbound Campaign Lifecycle

The user can perform appropriate actions on the Outbound Campaign at some of these stages. Performing an action on the Outbound Campaign may move to it a different stage.

Pega Customer Decision Hub

The following graphic illustrates the stages in the Outbound Campaign lifecycle and the actions that can be performed by the user at each stage.





The stages in the Outbound Campaign lifecycle are:

- New – Initial stage, Outbound Campaign is yet to be saved
- In Design – Outbound Campaign has been created and is in design
- Scheduled – Outbound Campaign is pending the scheduled run time
- Running – Outbound Campaign is currently running
- Awaiting Wrap-Up – Outbound Campaign has completed running and is available for wrap-up
- Failed – Outbound Campaign has failed. Actions in this stage are similar to those in the design stage.
- Completed – Outbound Campaign has been wrapped-up
- Archived – Outbound Campaign has been archived

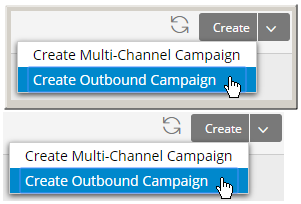
- Withdrawn – Outbound Campaign has been withdrawn

Creating a traditional outbound marketing campaign

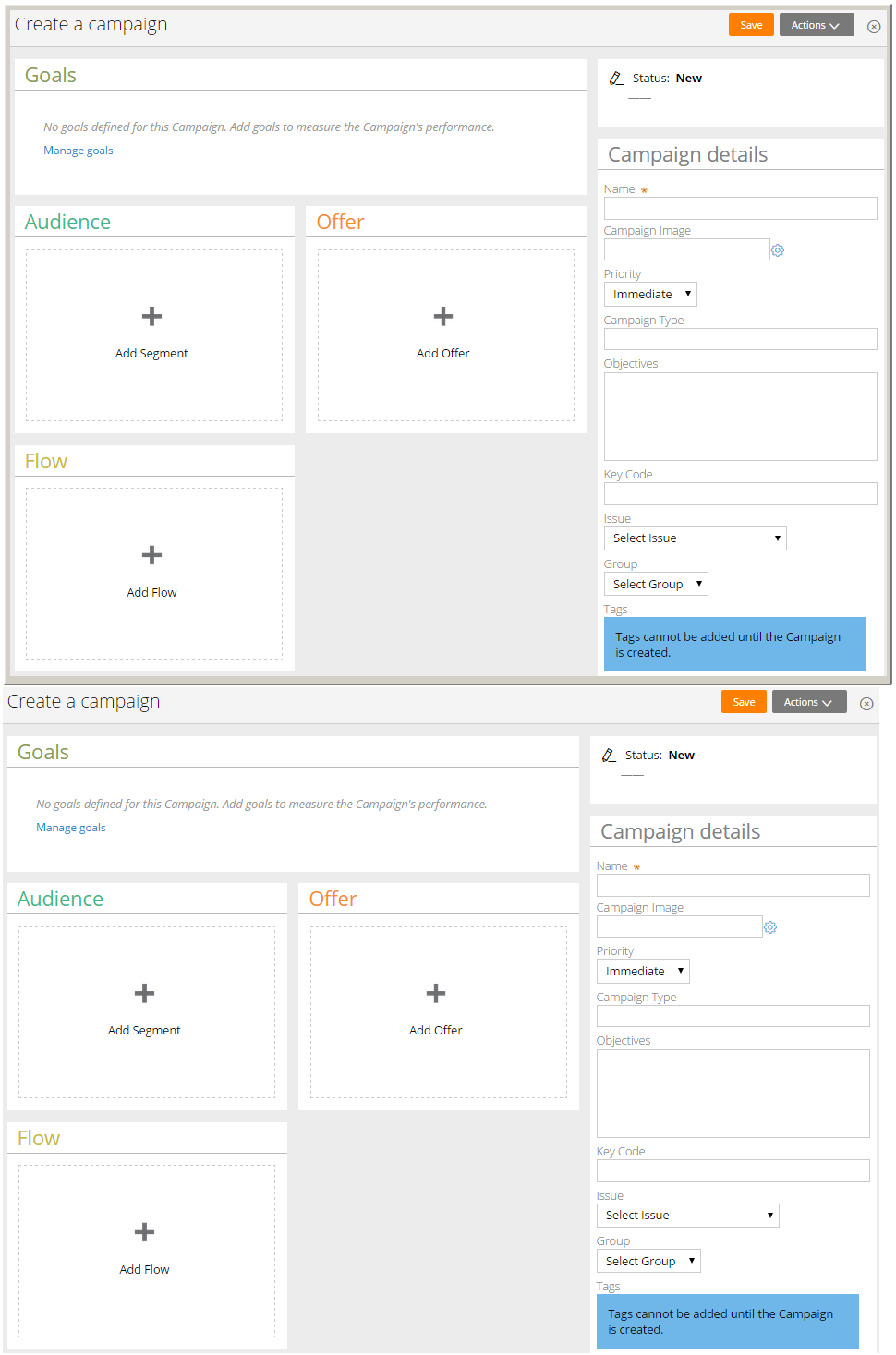
Outbound Campaigns are listed alongside Multi-Channel Campaigns on the Campaigns landing page. This landing page is accessed in the Pega Customer Decision Hub portal via the **Campaigns** link in the left navigation.

Pega Customer Decision Hub

To create a new Outbound Campaign, users should select the Create Outbound Campaign item in the **Create** button menu on the Campaigns landing page.



Upon creation, a new Outbound Campaign resembles the following:



Create a campaign
Save
Actions
Goals
No goals defined for this Campaign. Add goals to measure the Campaign's performance.
Manage goals
Status: New
Campaign details
Name *
Campaign Image
Priority
Immediate
Campaign Type
Objectives
Key Code
Issue
Select Issue
Group
Select Group
Tags
Tags cannot be added until the Campaign is created.
Audience
Add Segment
Offer
Add Offer
Flow
Add Flow

Configuring an Outbound Campaign

Marketers can start configuring the Outbound Campaign right from the **Create a Campaign** screen. Once the Outbound Campaign has been created, they can use the **Edit** button to go into edit mode and make further updates to the Outbound Campaign configuration.

Pega Customer Decision Hub

This section provides details on the various aspects of configuring the Outbound Campaign. This includes the following:

- Specifying Campaign Details
- Selecting the Audience
- Selecting the Action
- Selecting the Flow
- Specifying Goals

- Specifying Campaign Details
- Selecting the Audience
- Selecting the Action
- Selecting the Flow
- Specifying Goals

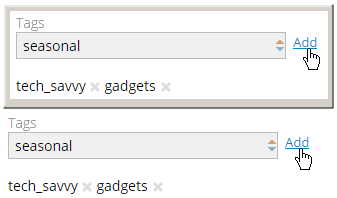
Specifying Campaign Details

The **Campaign details** section contains various details about the Outbound Campaign, such as its name, issue/group information, objectives, planned start date, and basic financial information. The Name, Issue, and Group can't be modified after the Outbound Campaign has been created.

Pega Customer Decision Hub
Note: The Outbound Campaign name must be unique across the system. The name must also begin with a letter and shouldn't contain any special characters (except spaces and underscores).

The user can utilize the **Campaign Image** field to associate an image with the Outbound Campaign. This image will be displayed in the card for the Outbound Campaign in the Campaigns landing page.

In addition to basic information, marketers can also add tags to their Outbound Campaigns. Tagging is available once the Outbound Campaign has been created.



Currently, tagging allows the marketer to further classify the Outbound Campaign as desired. In future iterations of the product, the tagging capabilities may be further enhanced to allow for tag-based searching.

Authorized users can enable the **Ignore global exclusion list** option to have processing of actions for this Outbound Campaign ignore the global exclusion list. Refer to the Exclusion List chapter for details.

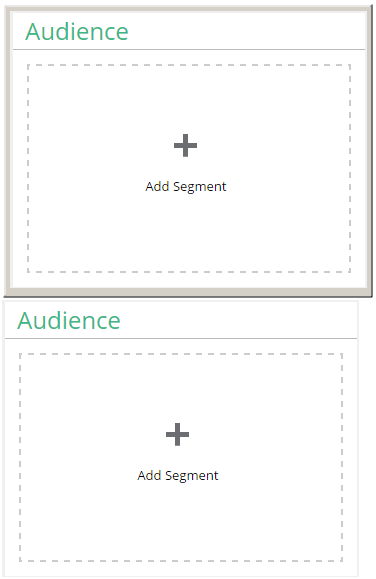
Selecting the Audience

A key element of Outbound Campaigns is the audience, which is the set of individuals the user plans to target with their marketing message. Audience selection in Outbound Campaigns utilizes the Segment rule, which can be used by the marketer to hone in on their desired audience.

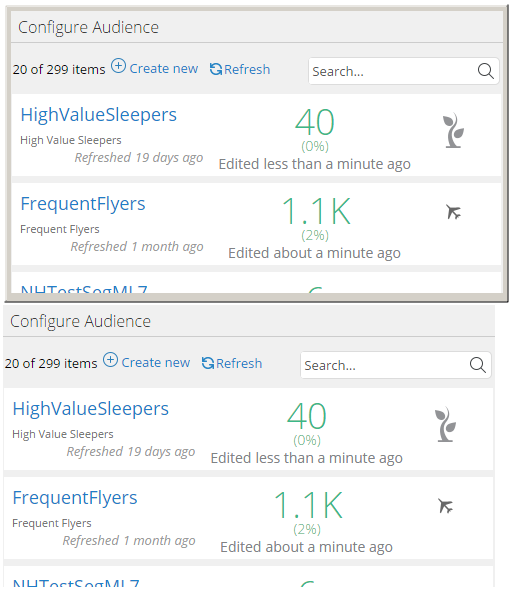
Pega Customer Decision Hub

When configuring an Outbound Campaign, the marketer can review their existing Segments (along with pertinent details) right from the Outbound Campaign canvas. The marketer can also create a new Segment and reference it in the Outbound Campaign.

The Audience card in the Outbound Campaign canvas displays the currently selected Segment and its details. Initially, this card is empty.

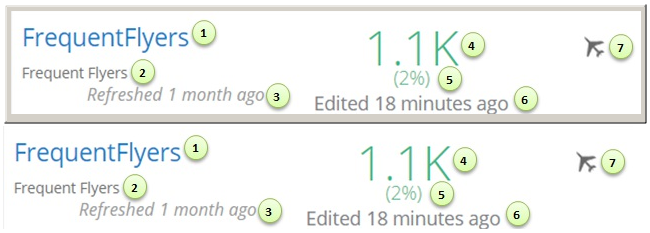


Marketers can click anywhere in the card to launch the **Configure Audience** dialog.



This dialog presents the twenty most recently updated Segments in the system. The **View more results** link at the bottom of the list enables users to load the next twenty Segments. Users can also utilize the search box to search for Segments by name or description. The **Refresh** link refreshes the list of Segments, retaining any search criteria that was previously entered.

Each Segment is represented by a card in the list.



This card contains the following details about the Segment:

- Segment name. Users can click the name to open the Segment rule.
- Segment description
- When the Segment's population was last refreshed
- Number of customers in the Segment
- Percentage of the total population represented by the Segment
- When the Segment was last edited
- Image associated with the Segment (if any)

If the desired audience is not represented by an existing Segment, users can also choose to create a new Segment via the **Create** link. After the Segment has been created, the user can use the **Refresh** link to reload the list of Segments. The user's new Segment should now show at the top of the list since it was most recently edited.

Clicking a Segment card selects the Segment. The right pane in the Configure Audience modal displays further details about the selected Segment. This includes the following common information for all Segment types:

- Segment image
- Friendly name - Click to open the Segment rule.
- When the Segment rule was last edited
- Number of Campaigns (both multi-channel and outbound) that are using this Segment as their audience

The following information is only shown for criteria Segments:

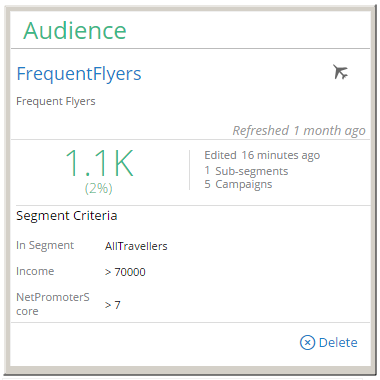
- Number of Segments that the selected Segment references
- List of criteria utilized by the Segment

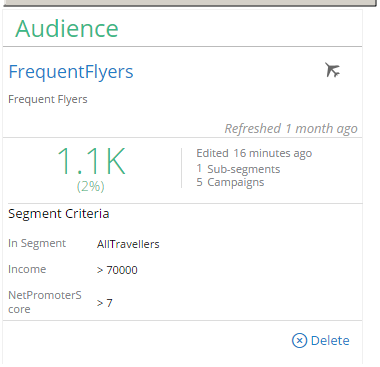
The following information is only shown for list Segments:

- Name of the last imported customer file
- Date and time when this file was imported
- Name of the user who imported this file

To complete the selection process, click the **Add** button to denote the selected Segment as the Outbound Campaign audience. The Audience card is updated to reflect the selected Segment. This card shows pertinent details about the Segment. This provides the user with a comprehensive summary of the selected Segment. The user can still open the referenced Segment by clicking the name link.

The **Delete** link in the bottom right corner of the Audience card (in edit mode) allows the user to remove the selected Segment. The user can then select a different Segment using the process outlined above.





Selecting the Action

Next, the marketer can select the action they want to use in their Outbound Campaign. The action portion of the Outbound Campaign focuses on the promotional content that the marketer wishes to communicate and the channels they wish to utilize.

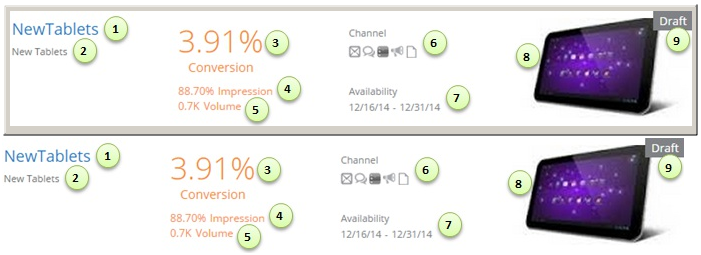
Pega Customer Decision Hub

The marketer can review their existing actions (along with pertinent details) right from the Outbound Campaign canvas. They can also create a new action and reference it in the Outbound Campaign.

The Action card in the Outbound Campaign canvas displays the currently selected action and its details. Initially, this card is empty.

Marketers can click anywhere in the card to launch the **Configure action** dialog. This dialog presents the twenty most recently updated actions in the system. The **View more results** link at the bottom of the list enables users to load the next twenty actions. Users can also utilize the search box to search for actions by name or description. The **Refresh** link refreshes the list of actions, retaining any search criteria that was previously entered.

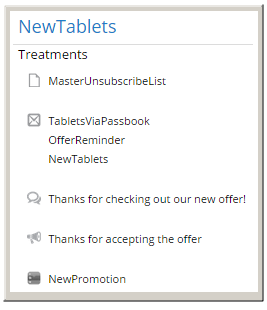
Each action is represented by a card in the list.

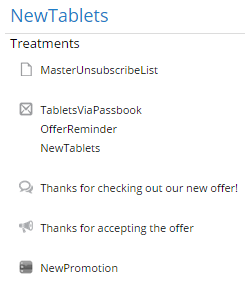


This card contains the following details about the action:

1. Action name. Users can click the name to open the action rule.
2. Action description
3. Conversion rate of the action
4. Impression rate of the action
5. Volume of the action – number of times this action has been made
6. Channels utilized by this action. Each channel is represented by its icon. Supported channels include Email, SMS, Passbook, Push Notification, and File/Database.
7. Availability of the action – duration when this action is available, if start and end dates are set on the action
8. Image associated with the action (if any)
9. Draft status of the action – **Draft** icon appears if the action is in draft mode

Clicking an Action card selects the action. The right pane in the Configure action modal displays further information about the engagement content that the action utilizes.





This information is grouped by communication channel and includes every channel that the action uses. In the example action above, all supported channels are being used. For each engagement, relevant information - such as the Email Treatment name or the Push Notification message - is displayed. Users can click the treatment name to open the associated treatment rule.

To complete the selection process, click the **Add** button to denote the selected action as the Outbound Campaign action. The Action card is updated to reflect the selected action. This card shows pertinent details about the action. This provides the user with a comprehensive summary of the selected action. The user can still open the referenced action by clicking the name link.

The **Delete** link in the bottom right corner of the action card (in edit mode) allows the user to remove the selected action. They can then select a different action using the process outlined above.

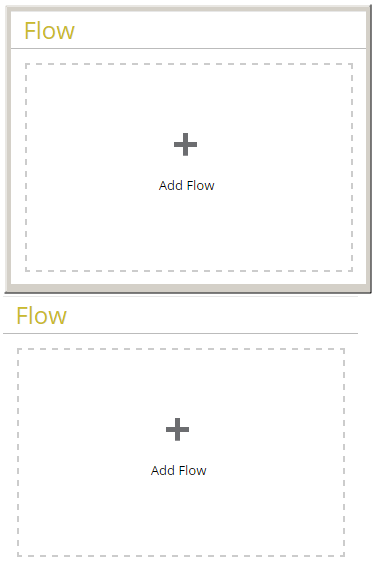
Note: In the current release of Pega Customer Decision Hub , the Action and Flow cards are inter-related as the metadata for both these cards uses the same Action rule. As such, selecting one of these cards auto-selects the other card as well. Similarly, removing either the selected action or Flow also removes the other.

Selecting the Flow

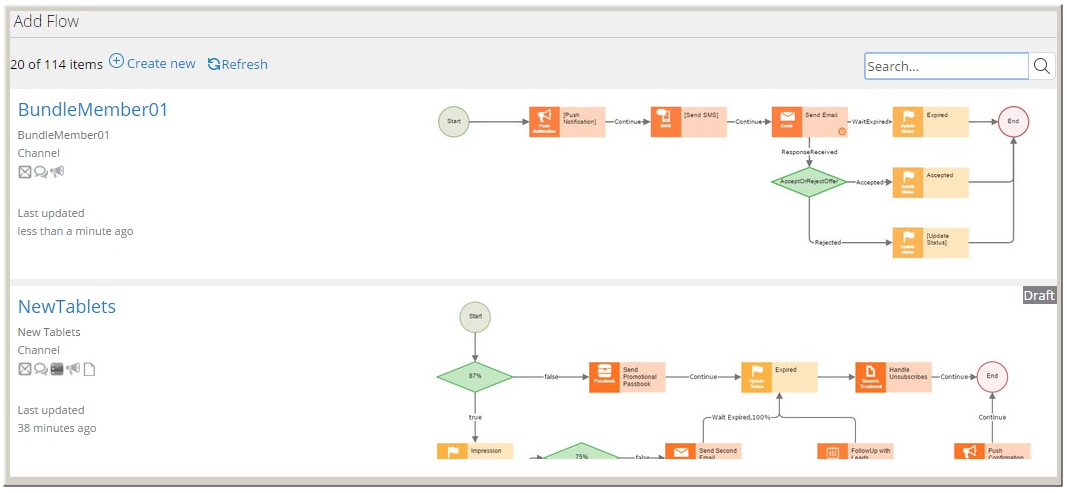
Finally, the marketer can select the Flow they want to use in their Outbound Campaign. The Flow determines the means in which the Outbound Campaign content (i.e. action) will be sent to the customers.

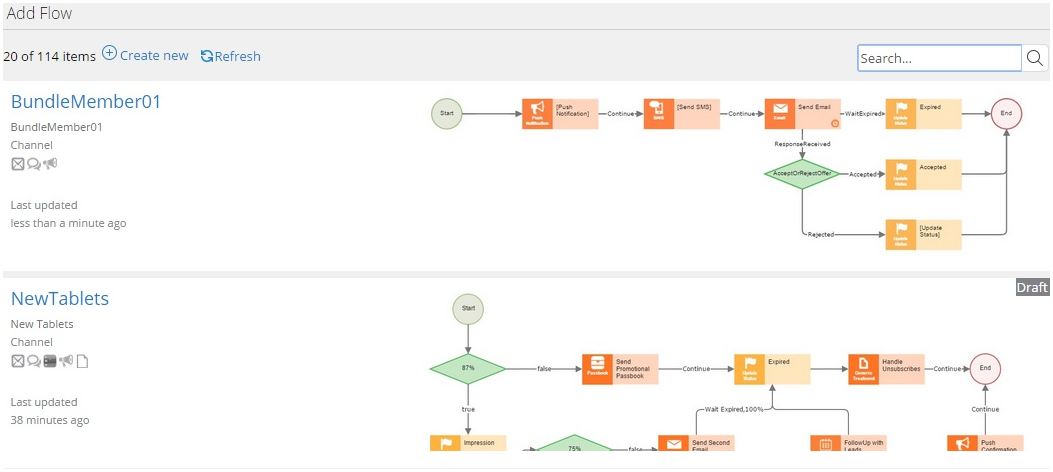
Pega Customer Decision Hub

In the current release, the Flow is also represented by the Action rule and is, therefore, coupled to the Action card. However, the Flow card displays flow-centric information and provides a different view to the marketer. The Flow card in the Outbound Campaign canvas displays the currently selected flow and its details. Initially, this card is empty as shown below:

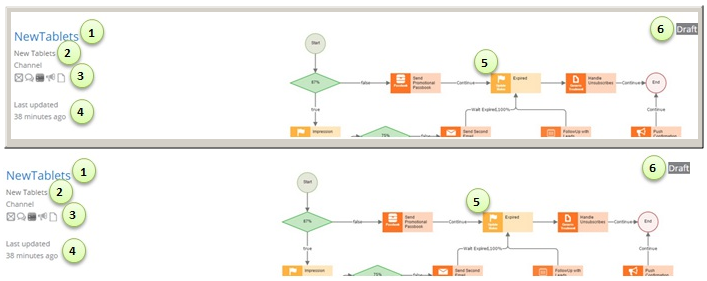


Marketers can click anywhere in the card to launch the **Configure Flow** dialog. This dialog presents the twenty most recently updated actions in the system. The **View more results** link at the bottom of the list enables users to load the next twenty actions. Users can also utilize the search box to search for actions by name or description. The **Refresh** link refreshes the list of actions, retaining any search criteria that was previously entered.





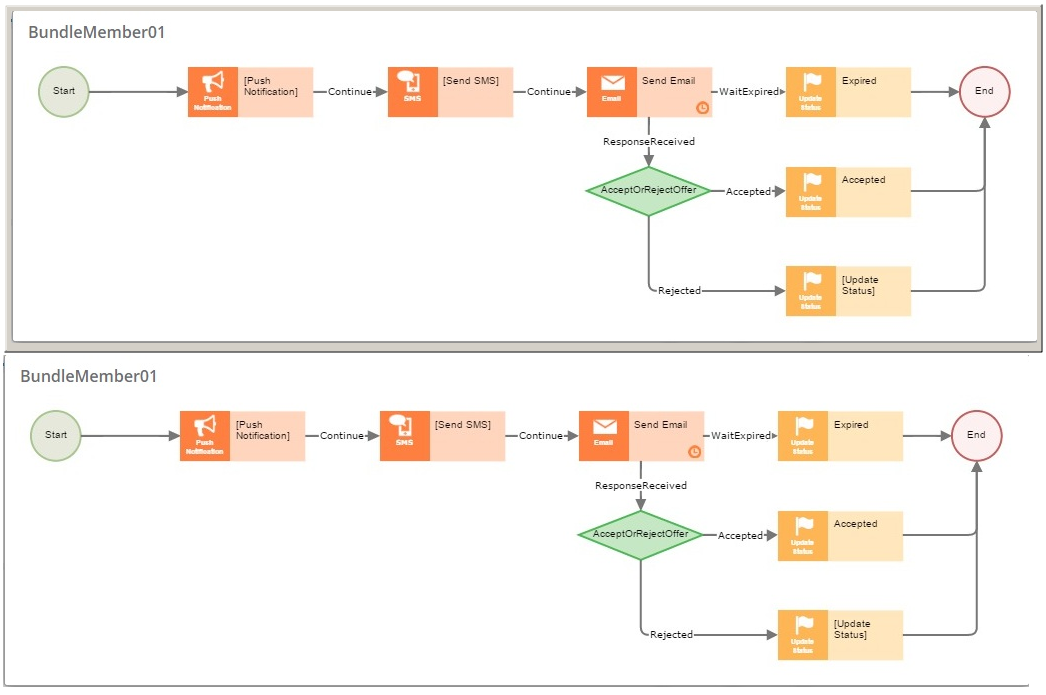
Each flow is represented by a card in the list.



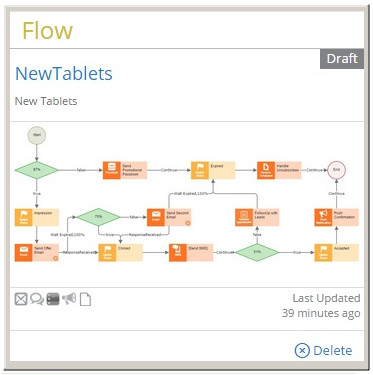
This card contains the following details about the action and its flow:

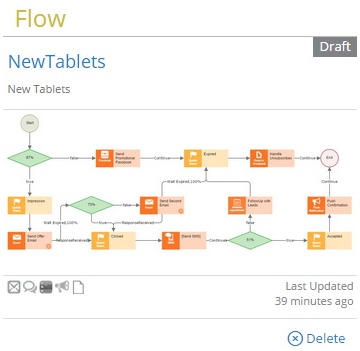
1. Action name. Users can click the name to open the Action rule.
2. Action description
3. Channels utilized by this action. Each channel is represented by its icon. Supported channels include Email, SMS, Passbook, Push Notification, and File/Database.
4. When the action was last updated
5. Flow
6. Draft status of the action – “Draft” icon appears if the action is in draft mode

Users can view the flow in more detail by clicking on the flow. This opens the flow in a modal window (as shown below). Clicking anywhere outside the modal closes the flow view.



Users can select a particular flow by clicking on the corresponding card. To complete the selection process, click the **Add** button to denote the selected Flow as the Outbound Campaign Flow. The Flow and Action cards are updated to reflect the selected action. The **Delete** link in the bottom right corner of the Flow card (in edit mode) allows the user to remove the selected action or flow. They can then select a different action or flow using the process outlined above.





Specifying Goals

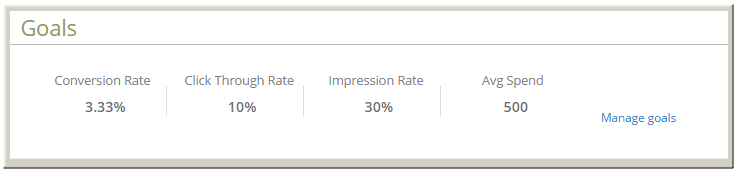
The **Goals** section of the Outbound Campaign allows marketers to define goals for their Outbound Campaigns and to monitor the performance of these goals once the Outbound Campaign is running.

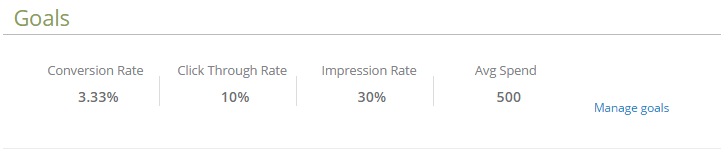
Pega Customer Decision Hub

When the Outbound Campaign is in edit mode, the **Manage goals** link is available in this section. Clicking this link launches a modal dialog that marketers can use to specify various goals for their Outbound Campaign. Initially, this modal contains a single row for the Outbound Conversion Rate metric.

Users can specify up to nine goal metrics. Each of these metrics is a Key Performance Indicator (KPI) configured on the Visual Business Director landing page. Clicking **Apply** adds all goals with target values to the Outbound Campaign.

The **Goals** section displays all goals that have been configured for the Outbound Campaign. Users can continue to make changes to these goals and their target values while the Outbound Campaign is in design mode. Once the Outbound Campaign is launched, these goals can no longer be edited.

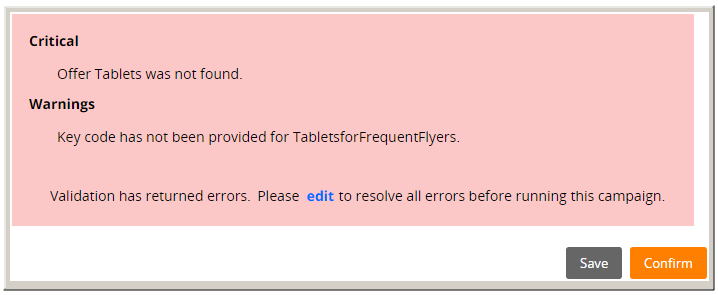


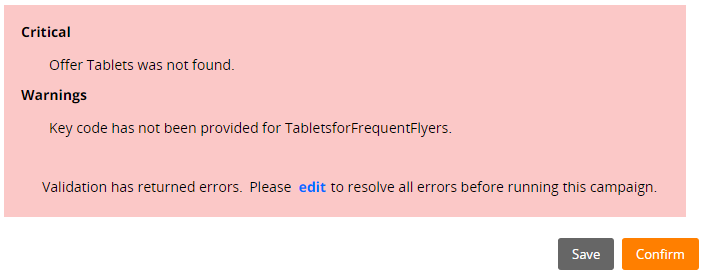


Validating an Outbound Campaign

Click **Actions Validate** to validate your Outbound Campaign configuration. Reported violations are categorized into critical items and warnings. Critical items will prevent the Outbound Campaign from being launched. Warnings represent best practice violations and other scenarios which may impact Outbound Campaign execution.

Pega Customer Decision Hub





The user can use the **edit** link to make any necessary updates to the Outbound Campaign.

Running an Outbound Campaign

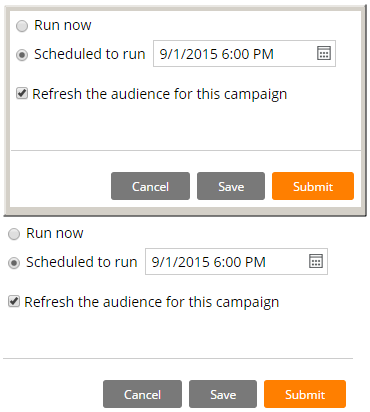
Once the user is satisfied with their Outbound Campaign configurations, they can launch the Outbound Campaign via the **Run** button or the **Run** action in the **Actions** menu.

Pega Customer Decision Hub

When this action is initiated, the system automatically performs Outbound Campaign validation and reports the validation results to the user. The user must address any validation errors before the Outbound Campaign can be launched. Warnings do not prevent the Outbound Campaign from being launched but should still be individually heeded and addressed, where necessary.

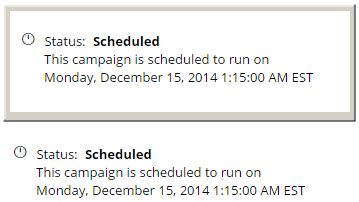
After addressing validation errors and warnings, the user can launch this Outbound Campaign. The user can choose to run the Outbound Campaign right away or schedule it for a time in the future. The system pre-populates the scheduled run time with the Outbound Campaign's Planned Start Date, if it is set.

The user can also choose to refresh the audience for the Outbound Campaign before it is run.



Note: The refresh audience option is only available when the audience of the Outbound Campaign is a criteria Segment.

After the user completes this action, the Outbound Campaign status switches to either Running or Scheduled based on the run option that was selected. The Outbound Campaign Status card displays this information along with the Outbound Campaign run duration (if running) or the scheduled Outbound Campaign start time (if scheduled).

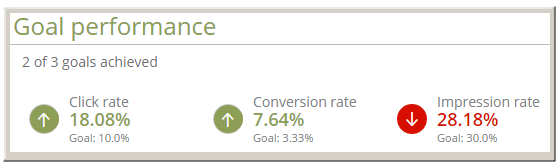


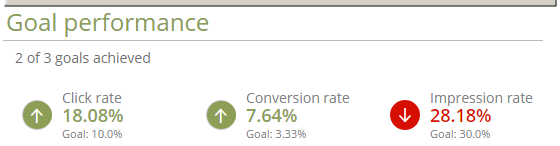
Monitoring Outbound Campaign Goals

Once the Outbound Campaign enters the running stage, the system tracks its performance and stores relevant information in Interaction History (IH). If the user specified performance goals on the Outbound Campaign, they can monitor these goals as the Outbound Campaign progresses.

Pega Customer Decision Hub

The **Goal performance** section displays relevant metrics both for the Outbound Campaign as a whole and for each individual goal.

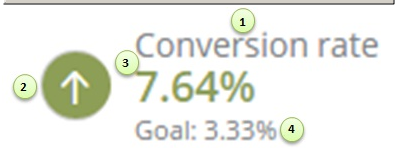




First, the number of goals achieved and the total number of goals are displayed.

Then, each goal is represented with pertinent metric data.





The following information displays for each goal:

1. Goal name
2. Goal status indicator – Up arrow when the goal has been achieved, down arrow otherwise
3. Current value for goal – Green font when achieved, red font otherwise
4. Target goal value – Set by the user as part of designing the Outbound Campaign

Goal metrics are fetched each time the Outbound Campaign is refreshed or re-opened. To get the latest metric values, the user can simply refresh the Outbound Campaign.

Making changes to a running campaign

This section provides details on the following additional actions that a user can perform on their Outbound Campaign:

Pega Customer Decision Hub

- Rescheduling an Outbound Campaign
- Suspending an Outbound Campaign
- Withdrawing a campaign
- Archiving an Outbound Campaign
- Restoring an archived Outbound Campaign
- Reopening an Outbound Campaign
- Managing Concurrent Campaign Data Flow Runs

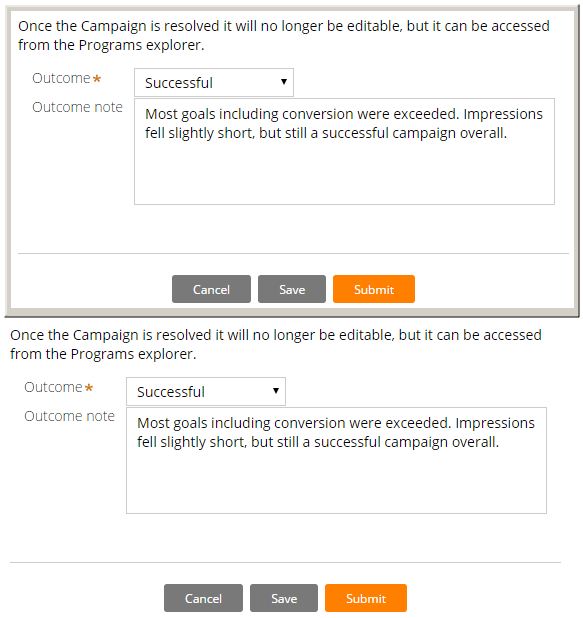
- Wrapping up a Campaign
- Rescheduling a Campaign
- Suspending a Campaign
- Withdrawing a campaign
- Archiving an Outbound Campaign
- Restoring an archived Outbound Campaign
- Reopening an Outbound Campaign
- Troubleshooting an Outbound Campaign
- Managing Concurrent Campaign Data Flow Runs

Wrapping up an Outbound Campaign

Once an Outbound Campaign has finished running, the user can perform one of the following actions:

Pega Customer Decision Hub

- **Edit** – This action allows the user to update Outbound Campaign details and enter actual values for data collected and other pertinent information about the Outbound Campaign and its outcome.
- **Wrap-Up** – This action allows the user to complete the Outbound Campaign and is available in the Actions menu. The user must select a value for the Outbound Campaign ’s outcome and can enter an outcome note as well.



Once the Campaign is resolved it will no longer be editable, but it can be accessed from the Programs explorer.
Outcome*
Successful
Outcome note
Most goals including conversion were exceeded. Impressions fell slightly short, but still a successful campaign overall.
Cancel
Save
Submit

Upon completion of this action, the Outbound Campaign moves to the Completed stage.

Rescheduling an Outbound Campaign

When the Outbound Campaign is in the scheduled stage, the user can update its run time (if desired) by utilizing the **Reschedule** action from the **Actions** menu.

Pega Customer Decision Hub

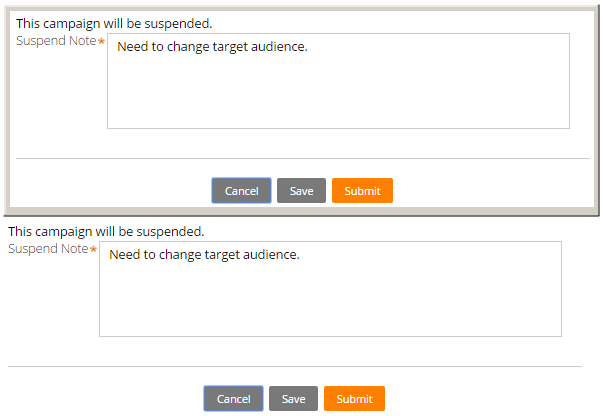
The Outbound Campaign is once again validated and the user is presented with the same configuration options as the Run action. Upon completing this action, the Outbound Campaign moves to either the running or the scheduled stage based on the run option that was selected.

Suspending an Outbound Campaign

If the user desires to make changes to a scheduled Outbound Campaign, they must first suspend it. To do so, the user can utilize the **Suspend** button or the **Suspend** action from the **Actions** menu. The user must also enter a note along with their suspension request.

Pega Customer Decision Hub

As part of suspending the Outbound Campaign, the system sends an email to the user that launched the Outbound Campaign notifying them of the Outbound Campaign's suspension. This email also includes the note that was entered.

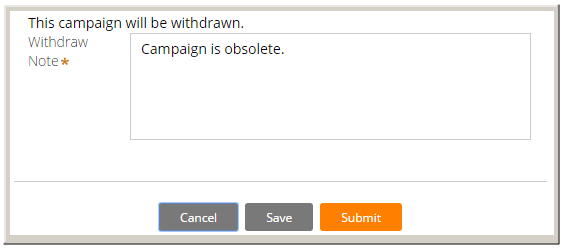


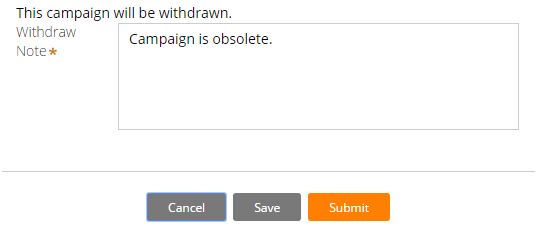
Suspending an Outbound Campaign returns it back to the In Design stage. The user can make further modifications to the Outbound Campaign before running it.

Withdrawing a campaign

While the Outbound Campaign is in either the design or the failed stage, the user can also withdraw it if it is no longer needed. To do so, the user can utilize the **Withdraw** action from the **Actions** menu and enter a note for the withdrawal.

Pega Customer Decision Hub





Withdrawing an Outbound Campaign moves it to the withdrawn stage. No further actions are available on this Outbound Campaign. The user can still utilize the **Save as** button to create a copy of this Outbound Campaign. The Outbound Campaign is also no longer listed (by default) in the Campaigns landing page. To view withdrawn Outbound Campaigns, users can set the **Status** filter on this landing page to **Withdrawn**.

Archiving an Outbound Campaign

Once the Outbound Campaign has finished running, the user can archive it via the **Archive** button or the **Archive** option in the **Actions** button menu.

Pega Customer Decision Hub

Archiving an Outbound Campaign moves it to the completed stage. The Outbound Campaign is also no longer listed (by default) in the Campaigns landing page. To view archived Outbound Campaigns, users can set the **Status** filter on this landing page to **Archived**.

Restoring an archived Outbound Campaign

To restore an archived Outbound Campaign, the user can utilize the **Restore** button or the **Restore** option in the **Actions** button menu. Restoring an Outbound Campaign moves it back to the stage it was in before being archived.

Pega Customer Decision Hub

Reopening an Outbound Campaign

To reopen a completed - either wrapped-up or restored - Outbound Campaign, the user can utilize the **Reopen** button or **Reopen** option in the **Actions** button menu. Re-opening an Outbound Campaign places it back in the awaiting wrap-up stage.

Pega Customer Decision Hub

Troubleshooting an Outbound Campaign

If any errors occurred in the data flow, they are displayed on top of the **Run details** section at the bottom of the **Status** section.

Pega Customer Decision Hub

Click the **Run details** link at the bottom of the **Status** section to view details about the campaign data flow. For easier troubleshooting, the **Run details** section also contains a link to the data flow. To open the data flow, click the flow ID in the **Data flow run** field.

Managing Concurrent Campaign Data Flow Runs

To optimize performance and stability of your Pega Customer Decision Hub application, the number of campaign data flows which run concurrently is limited. You can adjust the number of concurrent data flows and the retry interval by editing the related Dynamic System Settings.

Pega Customer Decision Hub
Note: The Dynamic System Settings listed below apply to all outbound, multi-channel, and self-optimized campaigns defined in your application.

- **MKTMaxAllowedBatchDFs** - Edit this setting to change how many campaign data flows can run concurrently. By default, ten runs can be active at the same time, and any additional data flows are paused for the duration defined in the setting **MKTDFRetryInterval**.
- **MKTDFRetryInterval** - Edit this setting to change how soon Pega Customer Decision Hub will attempt to execute a data flow which was paused because it exceeded the number of data flows defined in **MKTMaxAllowedBatchDFs**. By default, the retry interval is 30 minutes. After the inverval passes, Pega Customer Decision Hub checks the data flow availability again, for the number of times defined in the setting **MKTDFNumRetries**.
- **MKTDFNumRetries** - Edit this setting to change how many times Pega Customer Decision Hub will attempt to execute a data flow which was paused because it exceeded the number of data flows defined in **MKTMaxAllowedBatchDFs**. By default, the number of retries is 5. After the specified number of retries, the campaign run fails and an email notification is sent to the operator who triggered the run.

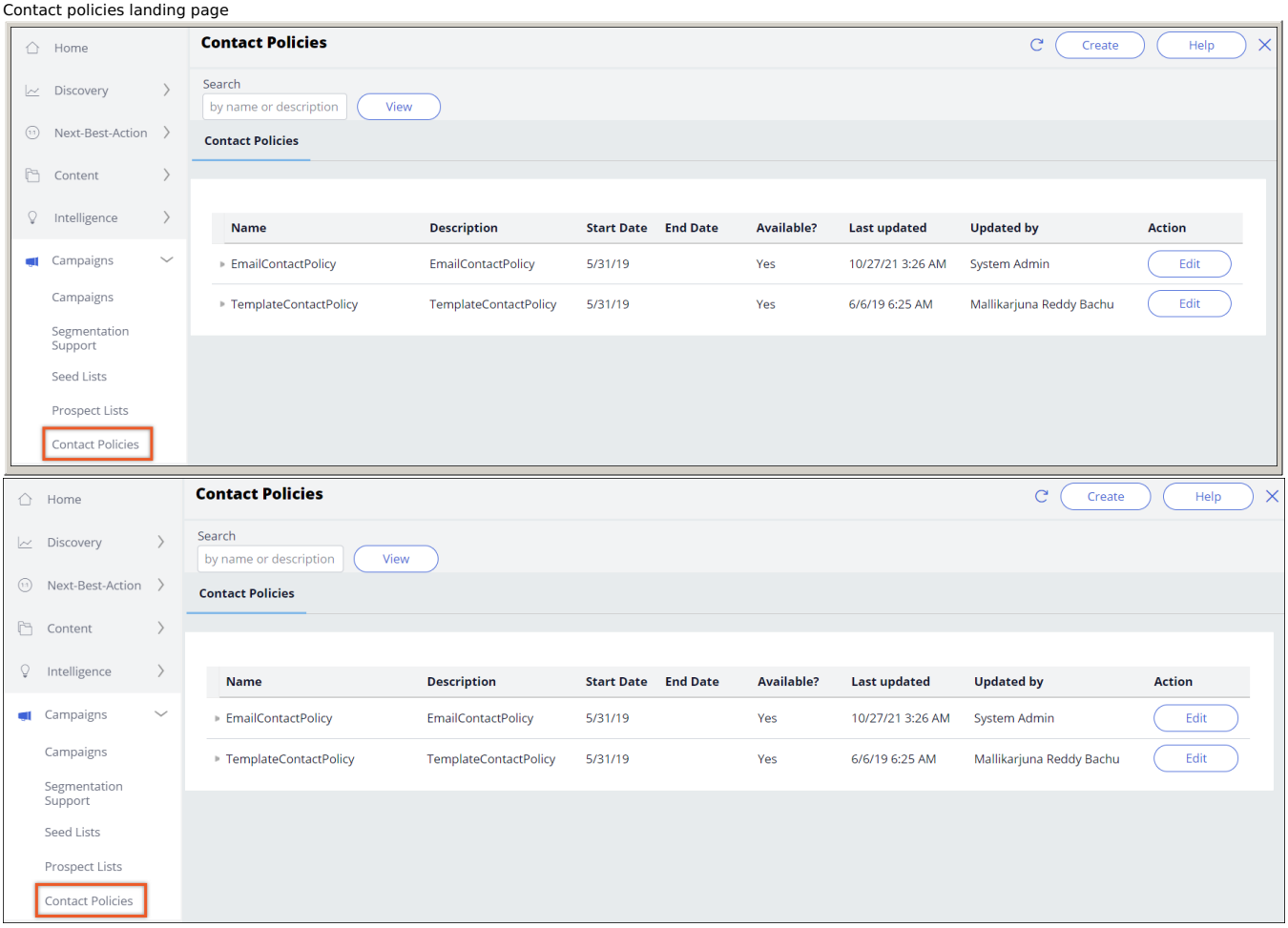
Defining outbound limits for traditional marketing campaigns

To define outbound limits for traditional outbound marketing campaigns, create a new contact policy.

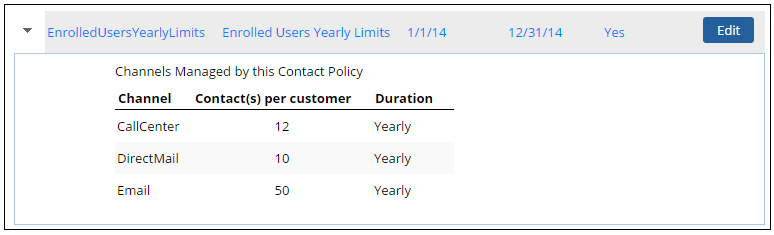
Pega Customer Decision Hub
Attention: This method should only be used for defining outbound limits for traditional outbound marketing campaigns. For outbound customer engagement Pega recommends to use the outbound schedule option available within the Next-Best-Action Designer, referred to as the next-best-action approach.

To create the Contact policy, you need access to the Campaigns menu, which is available only when the Marketing Campaign Edition is enabled in your system.

Contact policies landing page



The **Contact Policies** landing page provides a quick overview of the various Contact policies configured in the application. The landing page displays the start and end date of each policy along with its availability. Details about the limits specified for a particular policy are visible by expanding the row for the individual policy (via the arrow on the left side), as shown below:



1. In the Pega Customer Decision Hub portal, click **Campaigns Contact Policies** .
2. Click **Create**.

3. Enter a unique Contact Policy name in the **Name** field.
4. Choose the desired outcome to be tracked and click **Save**.

Using customer journeys to influence next-best-actions

Customer journeys are a sum of the experiences that your customers go through when interacting with your organization. Rather than looking at just part of an experience or transaction, the customer journey documents the full spectrum of the customer experience.

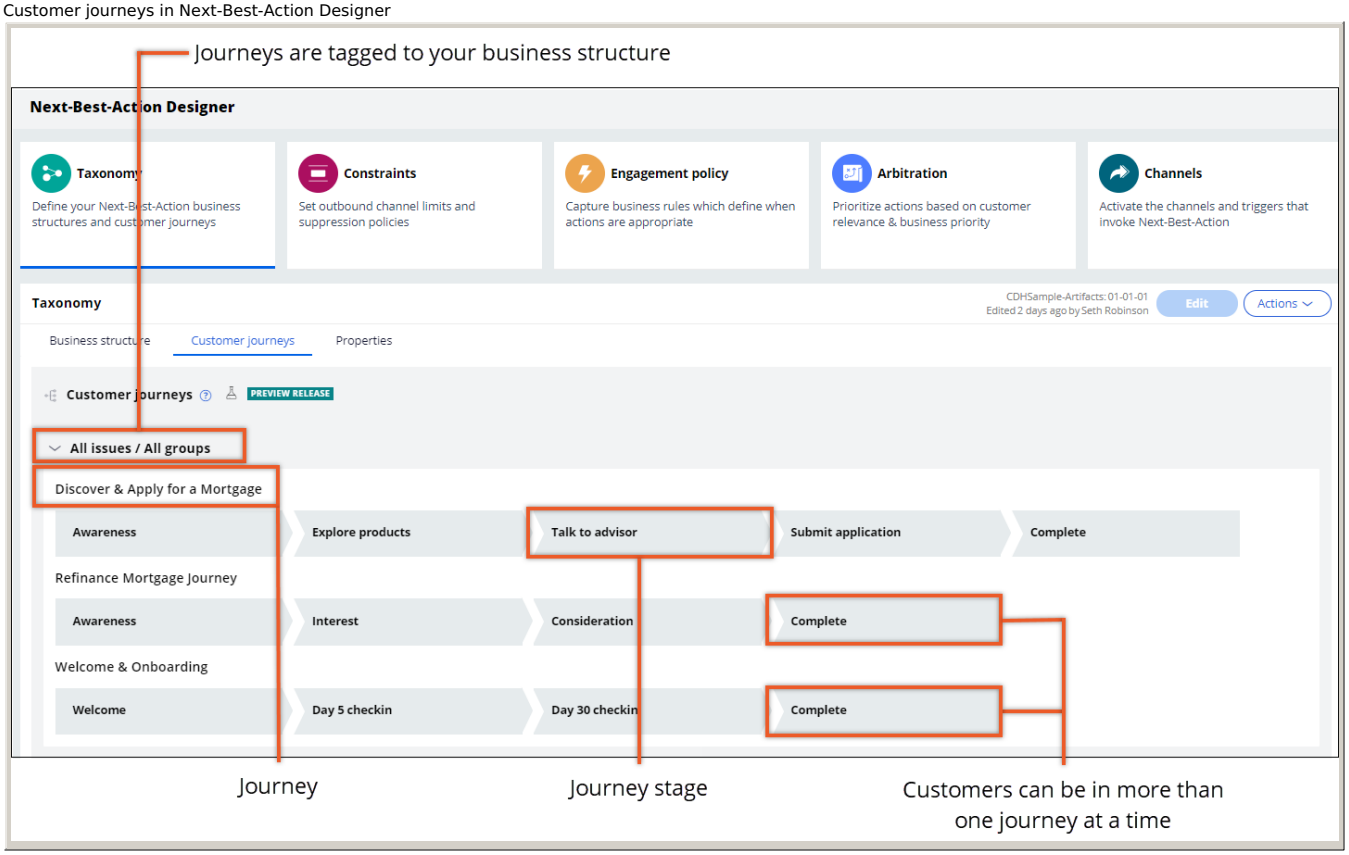
Pega Customer Decision Hub

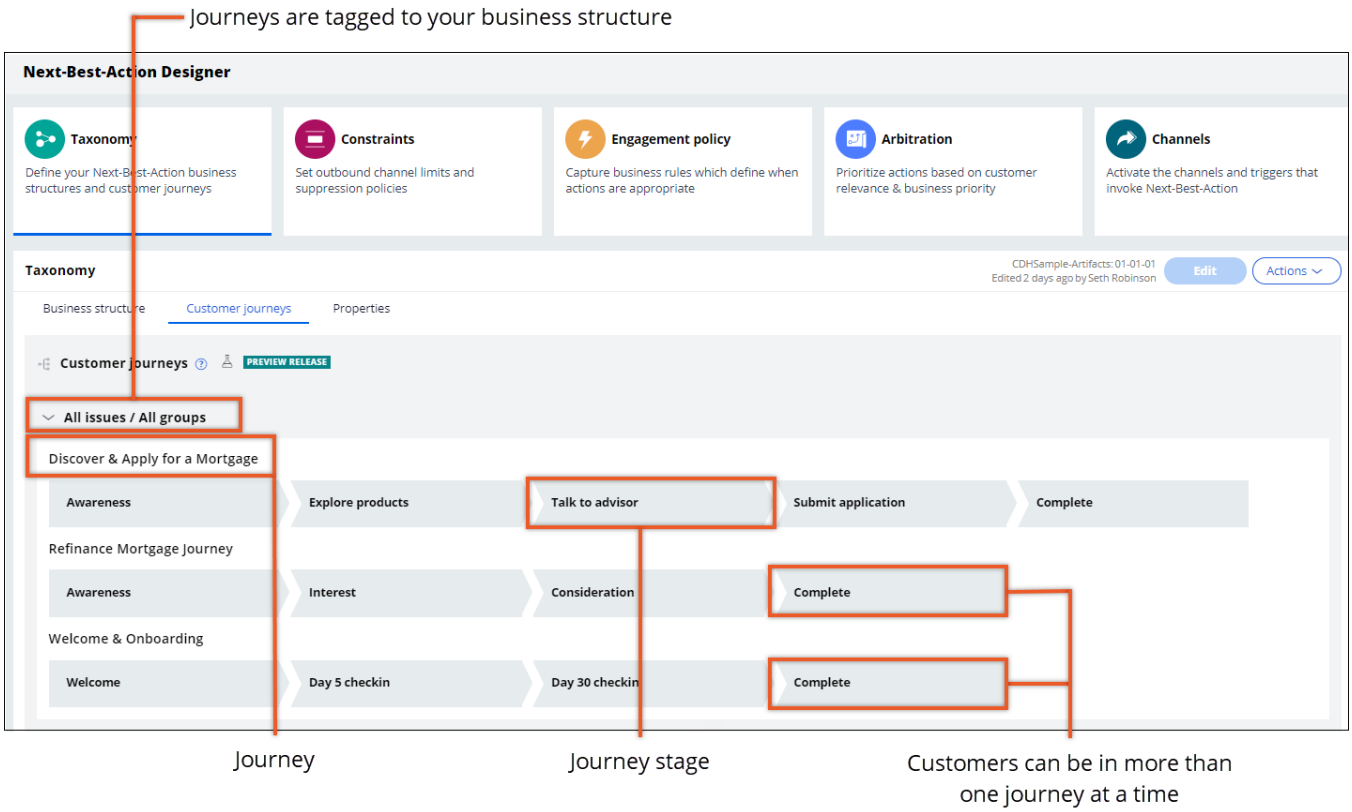
Customer journeys give marketers a familiar mental model to manage and work with their creative content. You can create customer journeys directly in Next-Best-Action Designer and map actions to all of the different journey stages. Each stage represents an experience that a customer goes through when interacting with your organization. By using customer journeys, you can see all of your actions and treatments for a particular journey laid out visually, so you can easily spot any gaps or white spaces in your engagement strategies.

For example: As a marketer responsible for customer acquisition for credit cards, you define a journey, with the appropriate stages, that a customer goes through when signing up for a new card. Customers first become aware of your card offering through an email, website, or paid channel. Then the acquisition process continues as they learn more about your specific offering and ultimately decide to submit an application. So the journey will likely consist of the following stages: Awareness, Consideration, Submit application, and Approval.

In addition to this mental model, customer journeys also influence the AI and the next-best-action decisions. The next-best-action strategy framework and analytical models take the previous stages and other journey metadata, and provide this as additional context to the analytics to help you make better decisions. Your next-best-action decisions are still going to make use of the arbitration formula (Propensity * Context * Value * Levers), but using customer journeys enhances the P (propensity) score with contextual customer journey data.

Customer journeys in Next-Best-Action Designer





To start building customer journeys, go to Creating a customer journey.

To learn more about how customer journeys influence NBA, see Understanding how customer journeys influence Next-Best-Actions.

- Creating a customer journey
- Specifying entry criteria for customer journey stages
- Associating actions with customer journey stages
- Visualizing your customer journeys
- Understanding how customer journeys influence Next-Best-Actions

Creating a customer journey

Customer journeys provide a way to organize your creative content around a specific outcome and to influence next-best-action decisions to determine the best action across multiple customer journeys.

Pega Customer Decision Hub

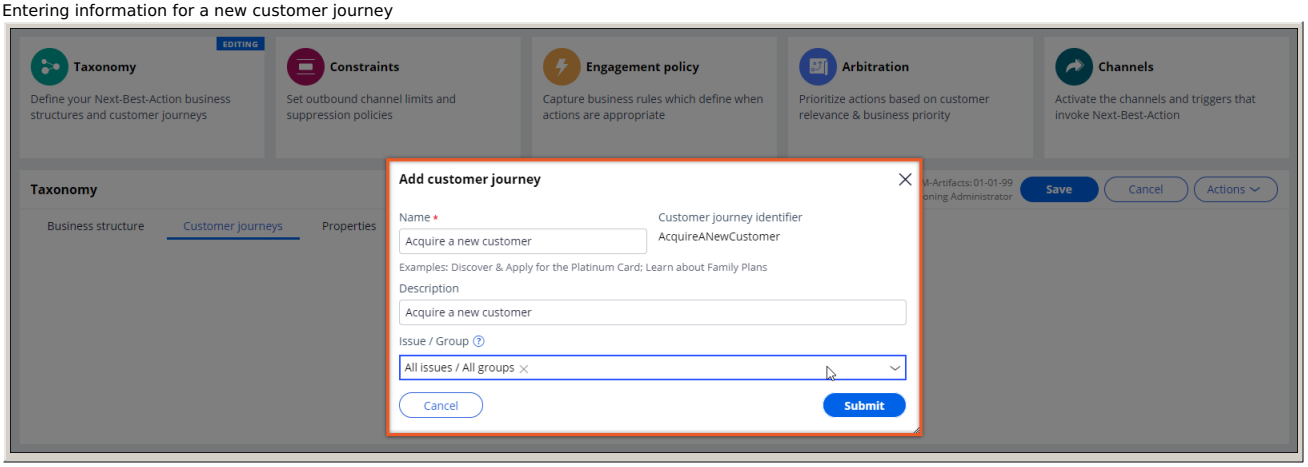
1. Log in as an operator who can access the Pega Customer Decision Hub portal.
2. Click **Next-Best-Action Designer Taxonomy** .
3. Click **Edit**, and then click the **Customer journeys** tab.
4. Click **Add customer journey**.
5. In the **Add customer journey** section, complete the following fields:

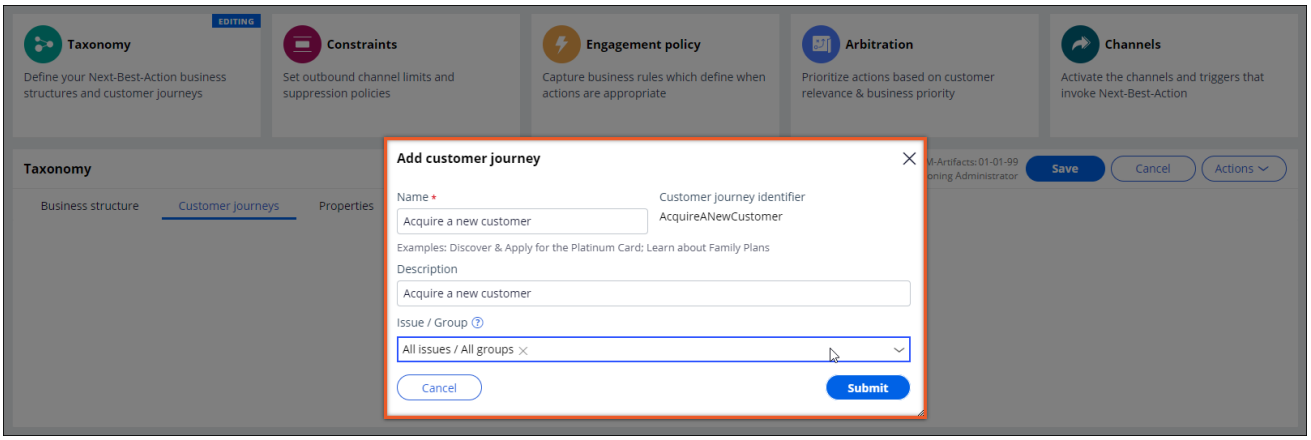
   Name
   : Enter a name that is descriptive and easily identifiable (for example, Acquire a new customer). After you add the name, the system generates the Customer journey identifier.

   Description
   : Describe the purpose of the journey and what outcomes you want to drive.

   Issue/Group
   : If you want to include actions from more than one issue and group in your journey, select **All Issues**. Otherwise, select a specific **Issue/Group**.

   Entering information for a new customer journey





6. Click **Submit**.
7. To edit the whole journey, click the **More** icon next to **+ Add stage** and choose **Edit journey**.
8. In the **More** menu next to the journey stage, define the stages that make up the customer journey.
   The Complete stage is mandatory for every customer journey and is automatically added. However, you can change the name of the Complete stage. The Complete stage indicates that a customer has finished their journey, for example, by purchasing goods or a service. There is no limit on the number of stages that a journey can have.
   - To add a new journey stage click **+ Add stage.**
   - To remove a journey stage, click the **More** icon next to the journey stage, and then select **Delete stage**.
   - To change name of the stage, click the **More** icon next to the journey stage, and then select **Edit stage name**.
   - To change the position of stages within the journey, drag and drop them.

   For example: A sales journey can consist of Awareness, Consideration and Complete stages, after which the journey is considered fulfilled.
9. Click **Save**.
   Note: You can have any number of journeys and your customers can be in more than one journey. Within a specific journey, customers will always be in the latest stage that they are eligible for. Journeys are not necessarily sequential, that is, customers do not have to go through all the stages that make up the journey.

What to do next: Specify the entry criteria for each of the stages and associate actions with each stage of your journey. For more information about it, see Specifying entry criteria for customer journey stages.

Specifying entry criteria for customer journey stages

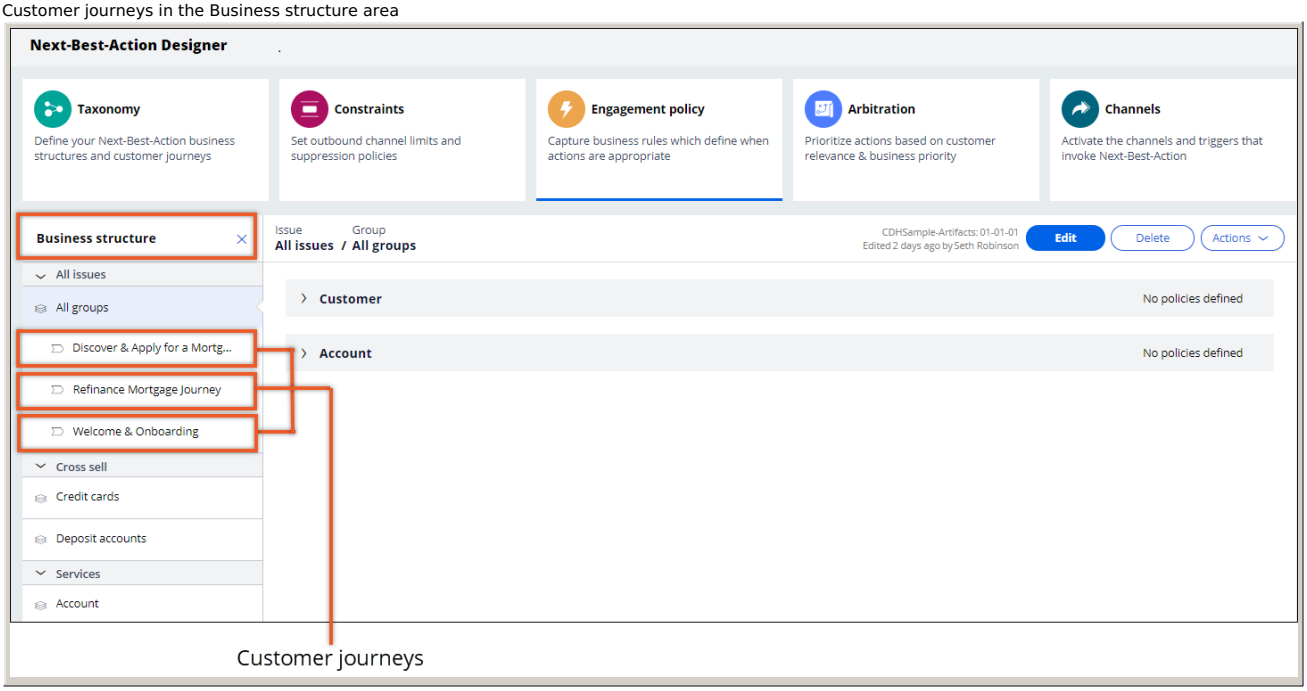
The entry criteria for each stage of your customer journey will determine whether an individual is eligible for a particular stage.

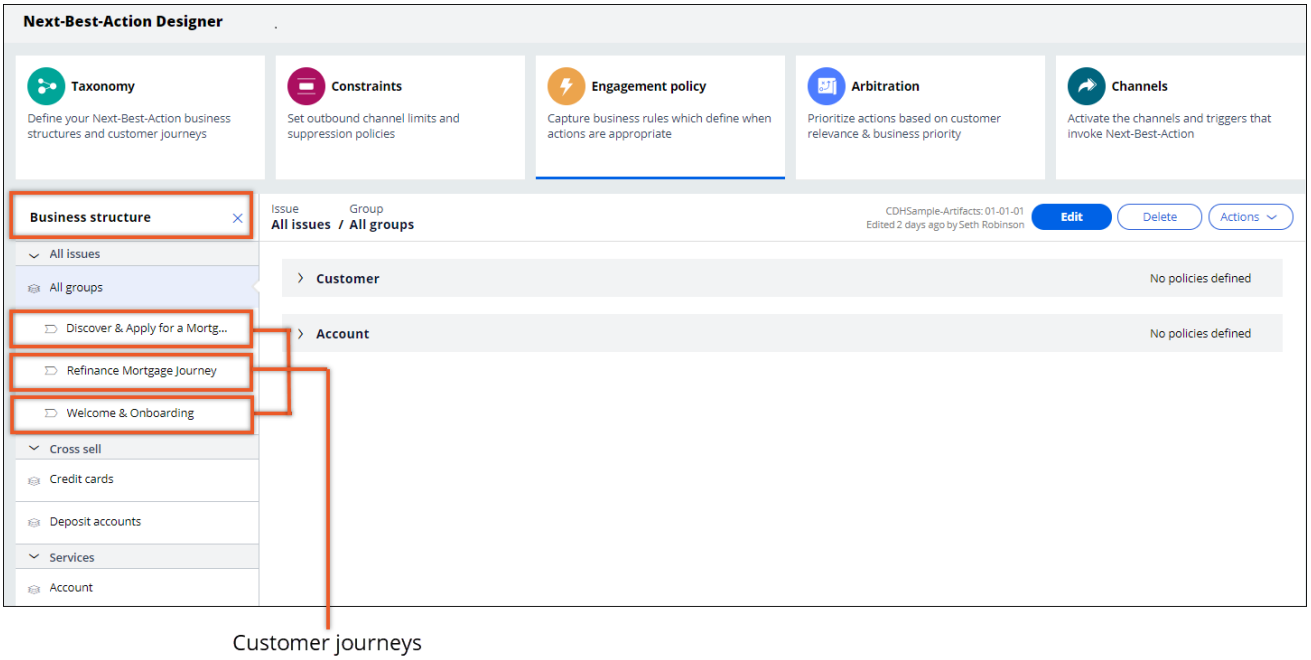
After defining the structure of your customer journey, specify the stage entry criteria.

Pega Customer Decision Hub

1. Log in as an operator with access to Pega Customer Decision Hub.
2. Go to **Next-Best-Action Designer Engagement policy** .
3. From the **Business structure** area, select one of your customer journeys:

Customer journeys in the Business structure area

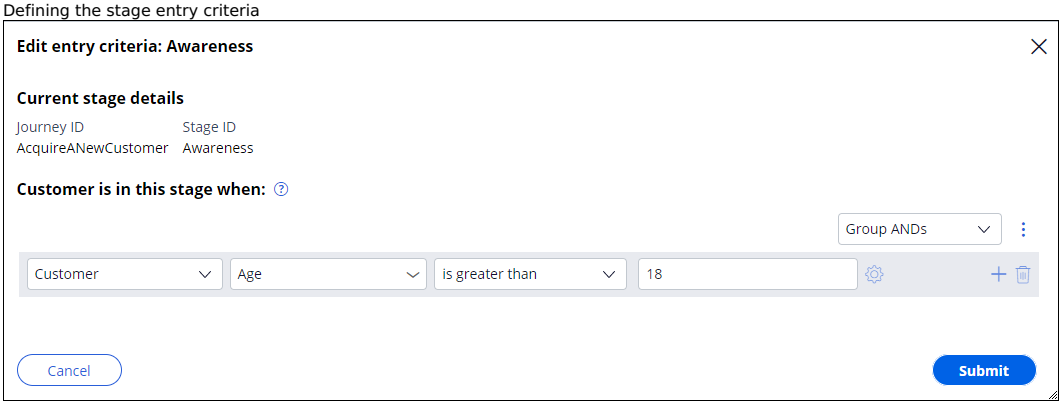




Customer journeys

4. Click the **More** icon on the specific journey stage.
5. Select **Edit stage entry criteria**.
6. In the **Customer is in this stage when** section, select the context, the strategies, and the when rules to define the stage entry rules.

Defining the stage entry criteria



7. Click **Submit**.
   Important: Besides the criteria that you define as part of the journey stage entry criteria, the regular engagement policy criteria must also be true for each individual action. A customer can be eligible or "in" a journey stage but not eligible for any actions in that stage.
8. Click **Save**. **Result:** The customer will be considered eligible for the journey stage if they meet the entry criteria.

What to do next: Associate actions with each stage of your journey. For more information, see Associating actions with customer journey stages. **Note:** Explore the journey that you created with the Journey Visualizer. For more information, see: Visualizing your customer journeys.

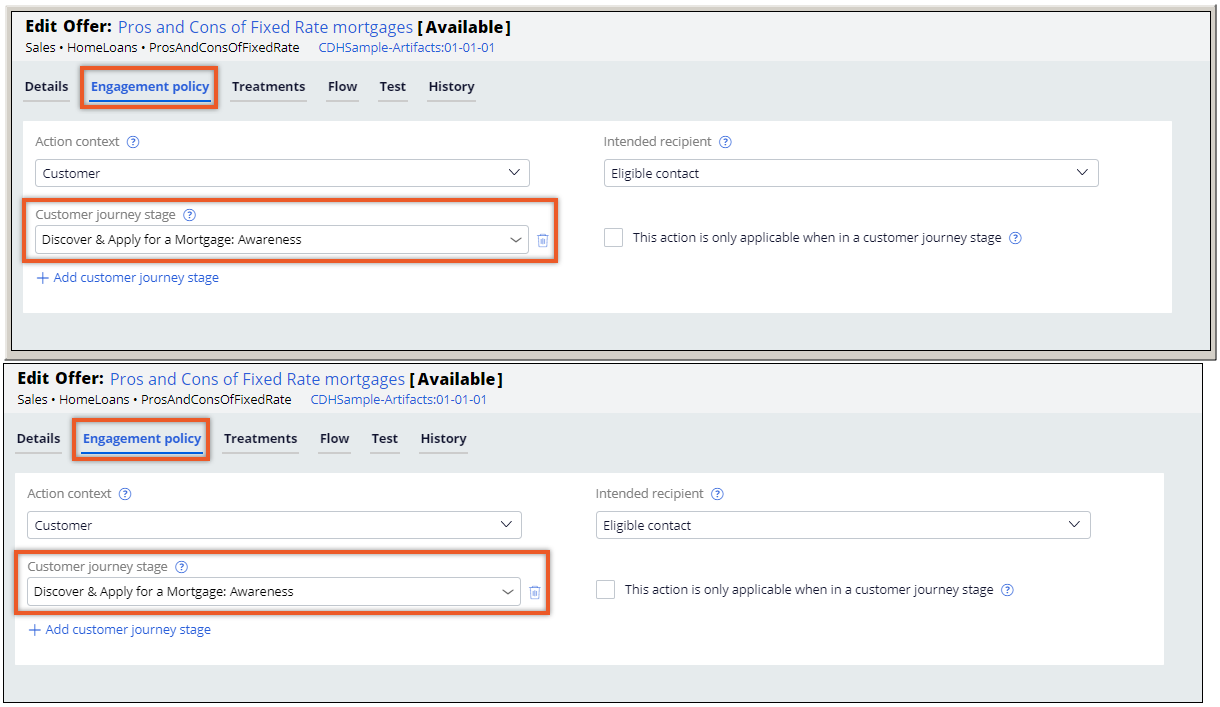
Associating actions with customer journey stages

After you specify the structure of a customer journey, and identify the entry criteria for each stage, you can associate actions with each stage of your journey.

Note: To learn about specifying the entry criteria for each customer journey stage, see Specifying entry criteria for customer journey stages.
Pega Customer Decision Hub

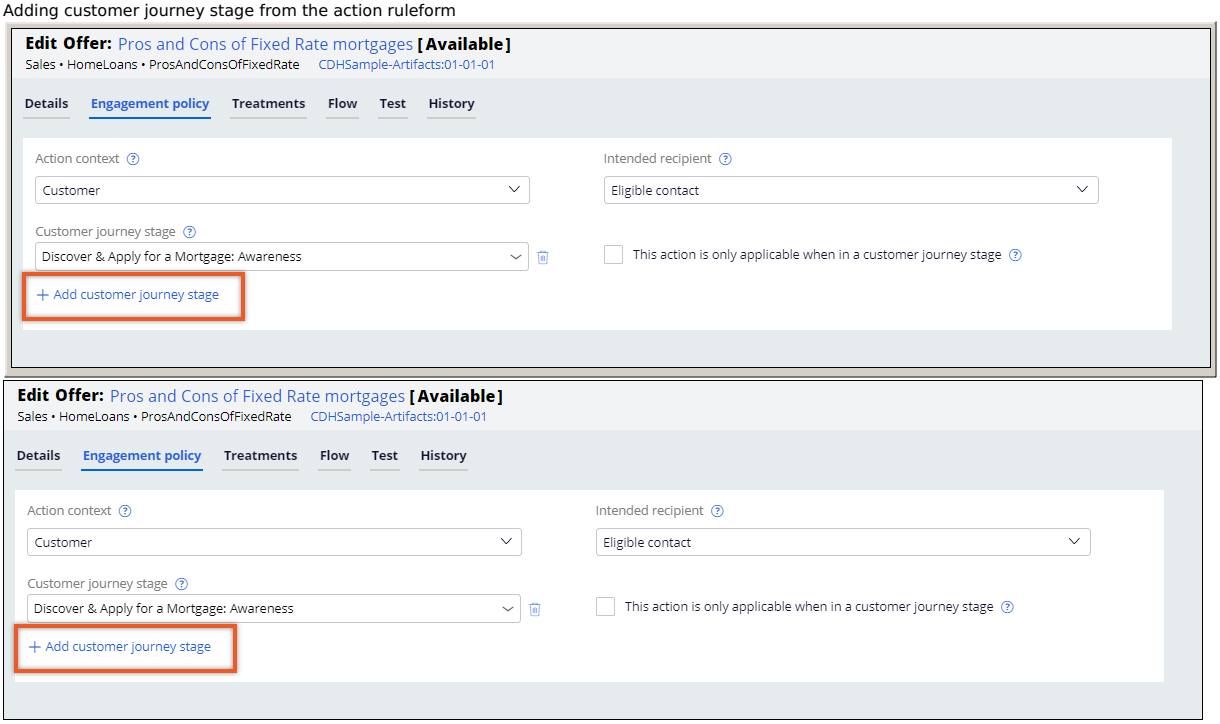
1. Log in as an operator with access to Pega Customer Decision Hub.
2. Go to **Next-Best-Action Designer Engagement policy** .
3. From the **Business structure** area, select one of your customer journeys.
4. Associate actions with each stage of the journey:
   a. Click the **More** icon by a specific journey stage.
   b. Select **Add/remove action**.
      Note: The actions displayed in the list are based on the level of the hierarchy that the journey is associated with. If the journey is defined at the top (All groups) level, you can select an action from any level of the hierarchy. If the journey is associated with a specific issue and group, for example Sales and Credit Cards, then you can only see Credit Card actions in the selector.
   c. Click the **Add** button next to the action that you want to add, and then click **Apply**.
      After you add the action, you can click the arrow next to it to see the treatments associated with that action.
   d. To associate more journey stages to an action, click an action to open it. On the **Engagement policy** tab of the action ruleform you can see the journey stage that this action is associated with, as in the following figure:

      The Engagement policy tab of the action ruleform



If you want to associate the action with additional journey stages, click **+ Add customer journey stage**

Adding customer journey stage from the action ruleform



What to do next: Explore the journey that you created with the Journey Visualizer. For more information, see Visualizing your customer journeys.

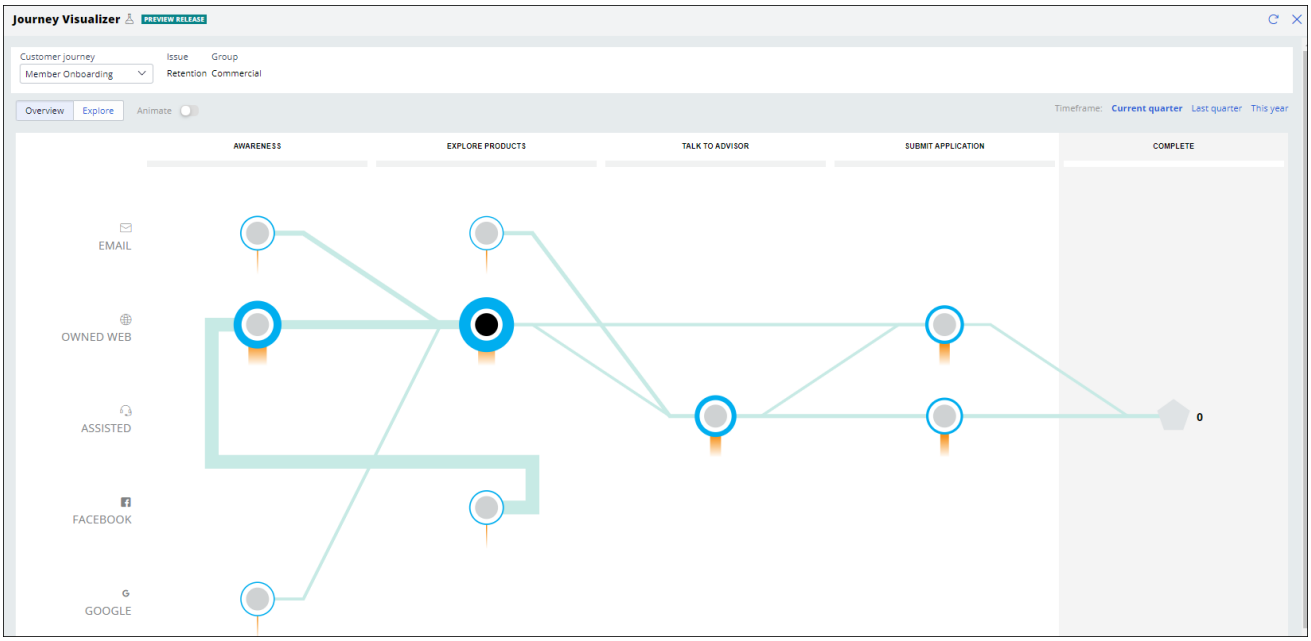
Visualizing your customer journeys

Journey Visualizer illustrates the paths your customers are taking through the journey, what channels, actions and treatments are working and where customers are getting stuck, indicating that more attention is needed in this area. It shows you what customers are currently doing in your customer journey.

After you define the structure of your customer journey and configure it, you can start visualizing those engagements. Next-best-action decisions will now include customer journeys and stages, and the appropriate associated actions. If these actions become the top actions for the specified channel, they will automatically be recorded into the customer journey data sets. You will see the view across all channels and be able to determine which channels are the most popular.
Pega Customer Decision Hub

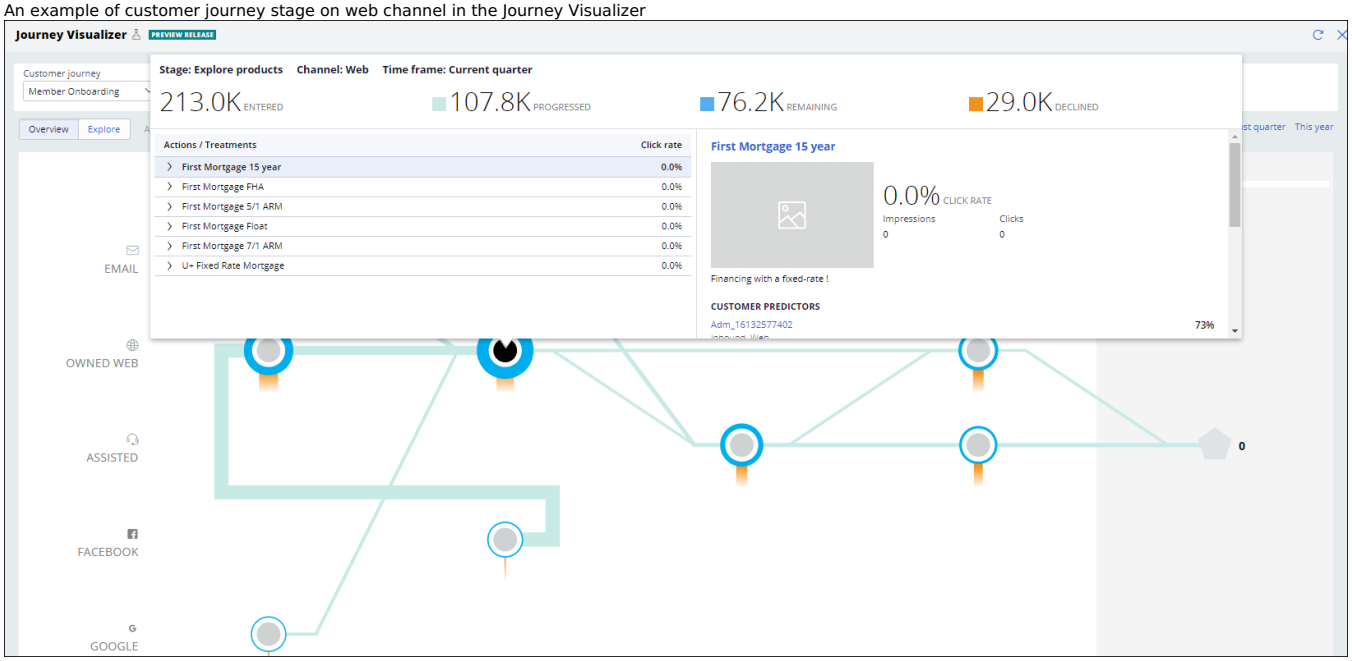
1. Log in as an operator with access to Pega Customer Decision Hub portal.
2. Click **Discovery Journey Visualizer** .
3. Select the journey that you want to examine from the **Customer journey** drop-down list. **Result:** The combined results for the customer journey that you selected are displayed.

An example of customer journey showing customers clicking Facebook ad and linking to a landing page



What to do next: Explore your journey by clicking on a journey stage. In the below example, the Journey Visualizer shows how many people entered this particular stage on the web channel, how many progressed to the next stage, how many people remain in this stage and how many declined the treatment and dropped out of the journey. The Journey Visualizer populates the data from across channels and customers, and shows how people are moving from stage to stage, as well as which actions are driving this behavior.

An example of customer journey stage on web channel in the Journey Visualizer



In a customer journey visualization, you can see where the customers learn about the product, what channels they engage through, and which messages help drive the desired outcome. When the customer gets stuck, you will see where the roadblocks are, where customers are dropping out, and where no one is interacting at all, so that you can prioritize key issues. If you find a journey that needs an upgrade, you can quickly adapt your strategy to guide customers around the problem.

Note: For more information about defining customer journeys, see Creating a customer journey. For more information about managing journey stages, see Specifying entry criteria for customer journey stages.

Understanding how customer journeys influence Next-Best-Actions

In addition to customer journeys providing a good mental model for managing content, they influence the next-best-action decisions made for your customers. You can configure the parametrized predictors in the adaptive models that affect the next-best-action decisions. You can also indicate which actions should be filtered out when the customer is in a specific journey stage.

Pega Customer Decision Hub

Adding parametrized predictors to adaptive models

The first approach to influencing a next best action is adding parametrized predictors to adaptive models, which will pass the current journey and stage as well as the previous stage as a parameter for next-best-action strategy framework. For new implementations of Pega Customer Decision Hub, adaptive models used by the next-best-action strategy framework already have the parametrized predictors added and mapped. If you are upgrading your Pega Customer Decision Hub from a version older than 8.6, you must manually add the parameters to the adaptive models, as well as the *ModelImpl* strategy that calls the models. For more information, see Adding predictors to existing adaptive models.

Note: You must add the predictors to the strategy models at two different levels: action and treatment.Add the following predictors to your adaptive models:

- *Journey* – represents the journey that the customer is currently in.
- *JourneyStage* – represents the stage that the customer is currently in.
- *LastJourneyStage* – represents the last stage of the journey that the customer has been in. This can be the current stage.

- *DaysinCurrentStage* – represents how many days have elapsed since the customer first entered the current stage.
- *PriorStageInJourney* – represents the stage prior to the one that the customer is currently in. If there were no stages before the current stage, the property is blank.

Filtering actions

By default, actions that are associated with customer journey stages can still compete in Arbitration, even if the customer is not in the current journey stage. When the customer is not in the current journey stage, the action is not tagged with the journey metadata. If you do not want an action to be eligible for arbitration unless the customer is in the journey stage that the action is associated with, filter out the action by configuring an action-specific engagement policy. For more information, see Filtering out actions not associated with the current customer journey stage.

- Adding predictors to existing adaptive models
- Filtering out actions not associated with the current customer journey stage

Adding predictors to existing adaptive models

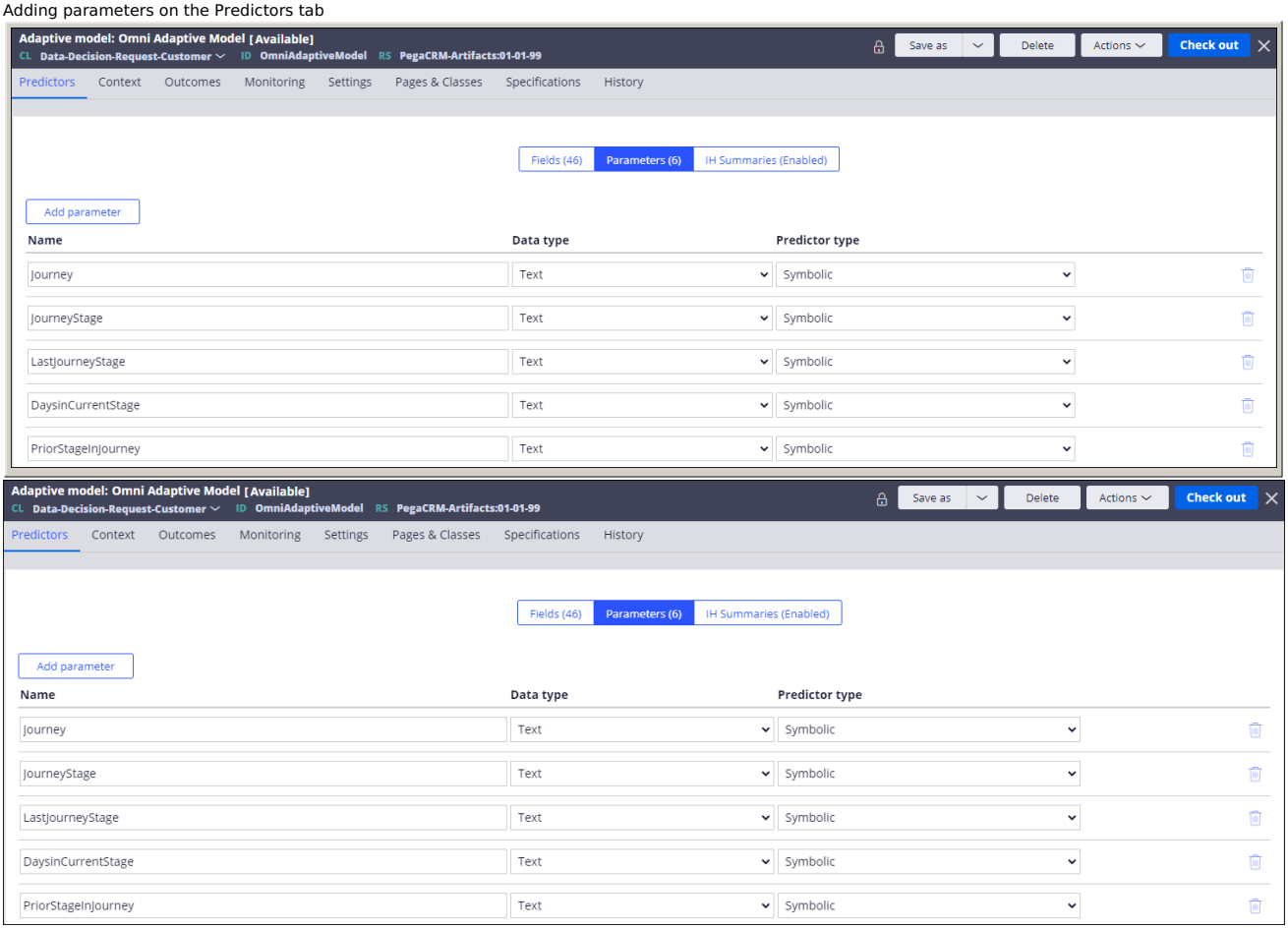
For new implementations of Pega Customer Decision Hub, when you save the Context Dictionary configuration, the journey predictors are automatically added to the adaptive models. If you are upgrading your application from version 8.5 or earlier, the predictors are not added automatically, and you must add them manually.

For more information about Context Dictionary, check Defining customer contexts for multilevel decisioning in Pega Customer Decision Hub.

Pega Customer Decision Hub
You must add the predictors to the strategy models at two different levels: action and treatment.

1. Log in as an operator with access to Dev Studio.
2. In the navigation pane of Dev Studio, click **Records Decision Strategy**
3. Select the strategy to which you want to add predictors
   - To add predictors to an action-level strategy model, select *ActionModelImpl*. The ruleset of the model should be *Artifacts*.
   - To add predictors to a treatment strategy model, select the treatments strategy, for example, *WebTreatmentModelImpl*. Every channel has a different treatment model.
4. Right-click on **Adaptive model component** and choose **Open Adaptive model**.
5. On the **Predictors** tab, click **Check out Parameters Add parameter** .
6. Add the following parameters:
   - *Journey*
   - *JourneyStage*
   - *LastJourneyStage*
   - *DaysinCurrentStage*
   - *PriorStageInJourney*
7. Make sure that the **Data type** for all added parameters is Text, and that **Predictor type** is set to Symbolic, as in the following figure:

Adding parameters on the Predictors tab

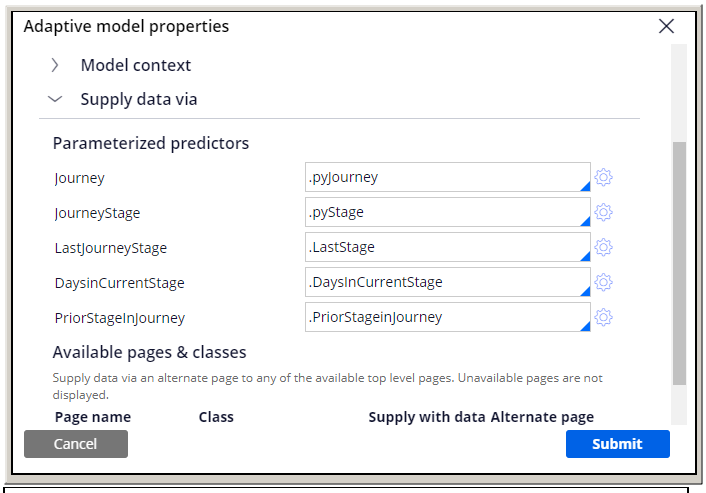


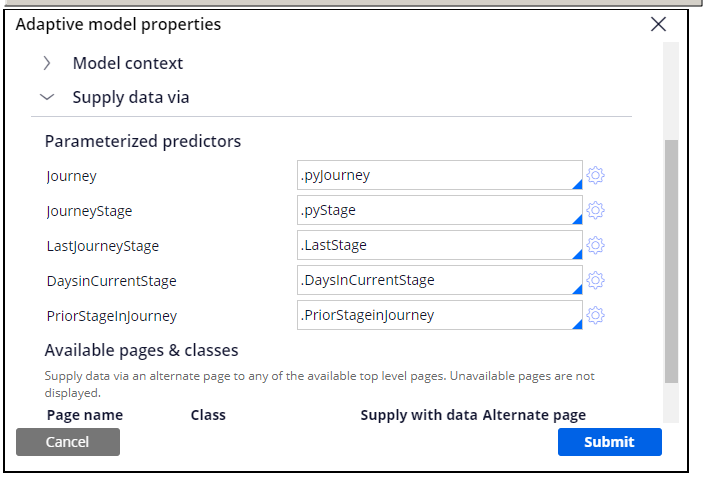
Adaptive model: Omni Adaptive Model [Available]
CL Data-Decision-Request-Customer ID OmniAdaptiveModel RS PegaCRM-Artifacts:01-01-99
Save as
Delete
Actions
Check out
Predictors
Context
Outcomes
Monitoring
Settings
Pages & Classes
Specifications
History
Fields (46)
Parameters (6)
IH Summaries (Enabled)
Add parameter
Name
Data type
Predictor type
Journey
JourneyStage
LastJourneyStage
DaysinCurrentStage
PriorStageInJourney
Text
Symbolic

8. Click **Save**, and go back to your strategy.
9. Right-click the **Adaptive model** component on the strategy canvas, and then click **Properties**.
10. In the **Parametrized predictors** section, map values to parameters:
    a. To the Journey parameter map *.pyJourney* value.
    b. To the JourneyStage parameter map *.pyStage* value.
    c. To the Last Journey Stage parameter map *.LastStage* value.
    d. To the DaysinCurrentStage parameter map *.DaysInCurrentStage* value.
    e. To the PriorStageInJourney parameter map *.PriorStageInJourney* value.

See the following figure for reference:

Mapping values to parameters





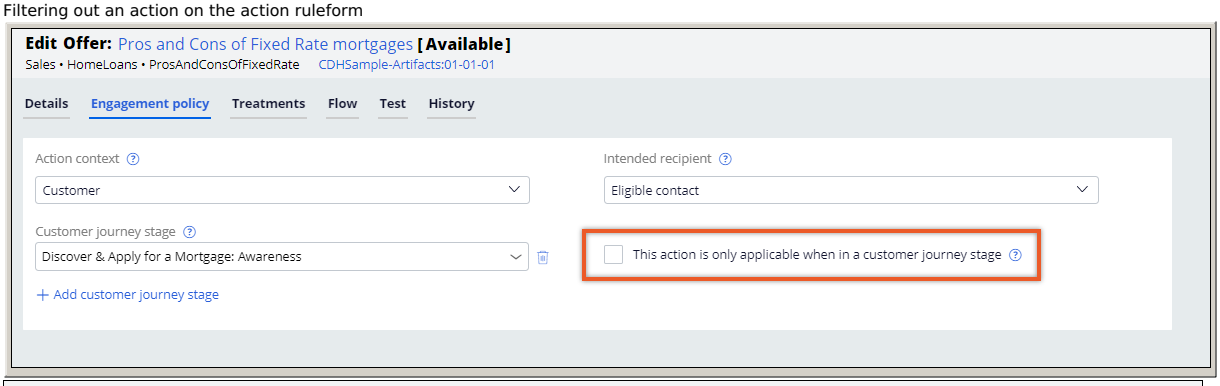
Filtering out actions not associated with the current customer journey stage

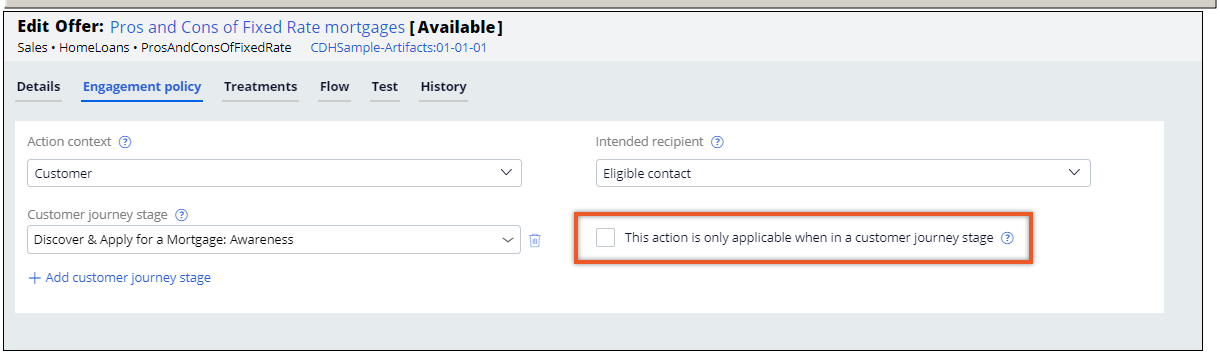
Actions that are associated with customer journey stages can still compete in Arbitration, even if the customer is not in the current journey stage. When the customer is not in the current journey stage, the action is not tagged with the journey metadata. If you do not want an action to be eligible for arbitration unless the customer is in the journey stage that the action is associated with, filter out the action by configuring an action-specific engagement policy.

Pega Customer Decision Hub

1. Log in as an operator with access to Pega Customer Decision Hub.
2. In the navigation pane, click **Next-Best-Action Designer Engagement policy** .
3. In the **Business structure** area, select one of your customer journeys.
4. In the specific customer journey stage, where you want to filter the action, click the action that you would like to modify.
5. On the **Engagement policy** tab of the action ruleset, select the **This action is only applicable when in a customer journey stage** check box.

Filtering out an action on the action ruleform





Result: This filters out the action, unless the customer is in the journey stage associated with the action.

Simulating Next-Best-Action configuration changes

To ensure that Next-Best-Action Designer configuration is correct, configure and run persona tests and simulation counts. With persona tests, you can check that you are getting the expected results for a specific user persona, and in this way validate your engagement policies and arbitration settings. After using a persona test to ensure that your engagement policies and arbitration are correct and give the expected results, run a simulation count to understand how many customers meet your specific engagement policies.

Pega Customer Decision Hub
In addition to manually running these tests, Pega Customer Decision Hub automatically adds all persona tests to a suite of tests, which you can use to manage regression testing for your application.

- Identifying opportunities to improve next-best-action strategies with Value Finder
- Testing Next-Best-Action configuration with persona testing
- Creating a data flow for testing Next-Best-Action strategy
- Deploying sample production data to a Business Operations environment for testing
- Simulation testing
- Optional: Comparing decision results in the Visual Business Director planner

Identifying opportunities to improve next-best-action strategies with Value Finder

Discover areas in which you can improve the next-best-action strategy by identifying where customers are left without actions. Learn for which groups of customers there are no relevant offers (underserved customers), and determine whether your prioritization is not sufficiently empathetic.

After you identify underserved customers, you can use this information to improve engagement by loosening restrictive policy rules, by adding relevant actions and treatments for your customers, or by adjusting the arbitration logic.

Pega Customer Decision Hub

Opportunities

The policy rules that filter out actions and treatments can leave customers with no action at different stages of the next-best-action decision funnel: eligibility, applicability, suitability, and arbitration. Value Finder analyzes what happens at every stage and examines policy rules at all levels: action, group, and global.

Value Finder also describes groups of underserved customers after eligibility, which means that after applying eligibility criteria, these customers do not receive any relevant actions, but are only eligible for low-propensity offers instead.

The current version of Value Finder does not yet fully support multilevel customer contexts. A multilevel data structure consists of more than one customer context, for example, an account, a subscription, an account owner, and a few device owners. When run on a multilevel data structure, Value Finder only considers the actions at the primary context level (for example, the account owner). Customer contexts below the primary context are disregarded.

Underserved customers

Are there groups of customers for which we do not have any relevant offers? A Value Finder analysis can help you answer this question. Value Finder describes these underserved customers by using available customer attributes, such as age or credit card ownership. These descriptions can help you introduce new actions and treatments that will be relevant to the underserved customers. If the group descriptions contain fields that do not provide actionable information, you can deselect these fields to exclude them from the descriptions. Saving a group as an audience can help you test distribution, to gain further insight.

A Value Finder analysis can also tell you if the arbitration logic leaves some customers without relevant actions. If there is too much focus on the business value of the offers, this might suppress the higher-propensity offers. As a result, the strategy might not be empathetic enough and the prioritization formula or the levers might need adjustment.

Value Finder simulations

The Value Finder landing page is the primary area of the Pega Customer Decision Hub portal where you can create, run, and review Value Finder simulations. Creating Value Finder simulations is also possible through simulation testing. For example, you can configure a Value Finder simulation test by using a data set with decision strategy results and customer data that originates from another environment.

If you want to retrieve the analysis results of a previous run, both landing pages list all Value Finder simulations, regardless of which landing page you used to create them. For example, a Value Finder simulation that you created on the Value Finder landing page is also available on the Simulation Testing landing page, and the other way round.

To learn when to use each option, see the following table:

Location	Use case
Discovery Value Finder	Create Value Finder simulations to analyze decision strategies that use the Next-Best-Action Designer framework. For more information, see Running Value Finder simulations.
Simulation Testing	Create Value Finder simulations through simulation testing. For more information, see Simulation testing. Use this option to: • Analyze decision strategies that do not use the Next-Best-Action Designer framework.**Note:** Ensure that the strategy results have the appropriate structure so that Value Finder can analyze the data. For more information, see the requirements for strategies. • Analyze decision strategies that you maintain in an older environment which does not support Value Finder.**Note:** This type of simulation requires a data set with decision strategy results and customer data. For more information, see the requirements for data sets.

Run simulations to discover gaps in customer engagement, analyze the simulation results to gain valuable insights, and use those insights to engage more customers.

- Running Value Finder simulations
- Setting the underserved threshold
- Analyzing group descriptions
- Saving customer groups as audiences

Running Value Finder simulations

Run a Value Finder simulation on an audience to discover which stages of your next-best-action strategy leave customers without any actions or with only low propensity actions, and identify groups of customers that are underserved.

Pega Customer Decision Hub
This procedure explains how to create and run simulations on the Value Finder landing page, on which you can perform the following tasks:

- Running and viewing analysis results for all Value Finder simulations that were created in your system.
- Creating Value Finder simulations to analyze strategies based on the Next-Best-Action Designer strategy framework.

Note: To analyze a strategy that was created outside Next-Best-Action Designer, or strategy results from a lower-version environment that does not include Value Finder, go to the Simulation Testing landing page and create a Value Finder simulation test. For more information, see Running simulation tests.

1. Log in to the Pega Customer Decision Hub portal.
2. In the navigation pane of the Pega Customer Decision Hub portal, click **Discovery Value Finder** .

3. To run a saved simulation, from the **Simulation** list, select a simulation. **Result:** Value Finder runs the simulation automatically.
4. To create a simulation and then run it, perform the following actions:
   a. Click **Create new**.
   b. In the **Create simulation** window, click **All issues / All Groups**.
   c. From the **Issue** list, select an issue.
   d. From the **Group** list, select a group.
   e. Click **Apply**.
   f. In the **Create simulation** window, click **Audience**, and then select the audience for the simulation.
   g. Click **Run**.

   Result: After you start a simulation, you can navigate away from the Value Finder landing page, and then can come back later when the analysis results are ready. Value Finder runs the simulation and saves it to the **Simulation** list. The simulation is also available on the Simulation Testing landing page.
5. To re-run a simulation that you ran before, under the **Simulation** list, click the **Re-run** icon.
6. **Optional:** To toggle the customer distribution view between counts and percentages, in the **Show customers as** field, select one of the following options:
   - To display results as the numbers of customers in each category and state, select **Counts**.
   - To display results as percentages of customers in each category and state, select **Percentages**.

Setting the underserved threshold

Value Finder identifies customers as underserved if the propensity of all the actions or treatments available to a customer is below the configured threshold propensity. To provide a convenient starting point, on first analysis, the propensity threshold value is chosen in such a way that 1 out of 20 customers is defined as underserved.

You can change the propensity threshold according to your business definition of underserved customers and then re-run the Value Finder analysis.

Pega Customer Decision Hub

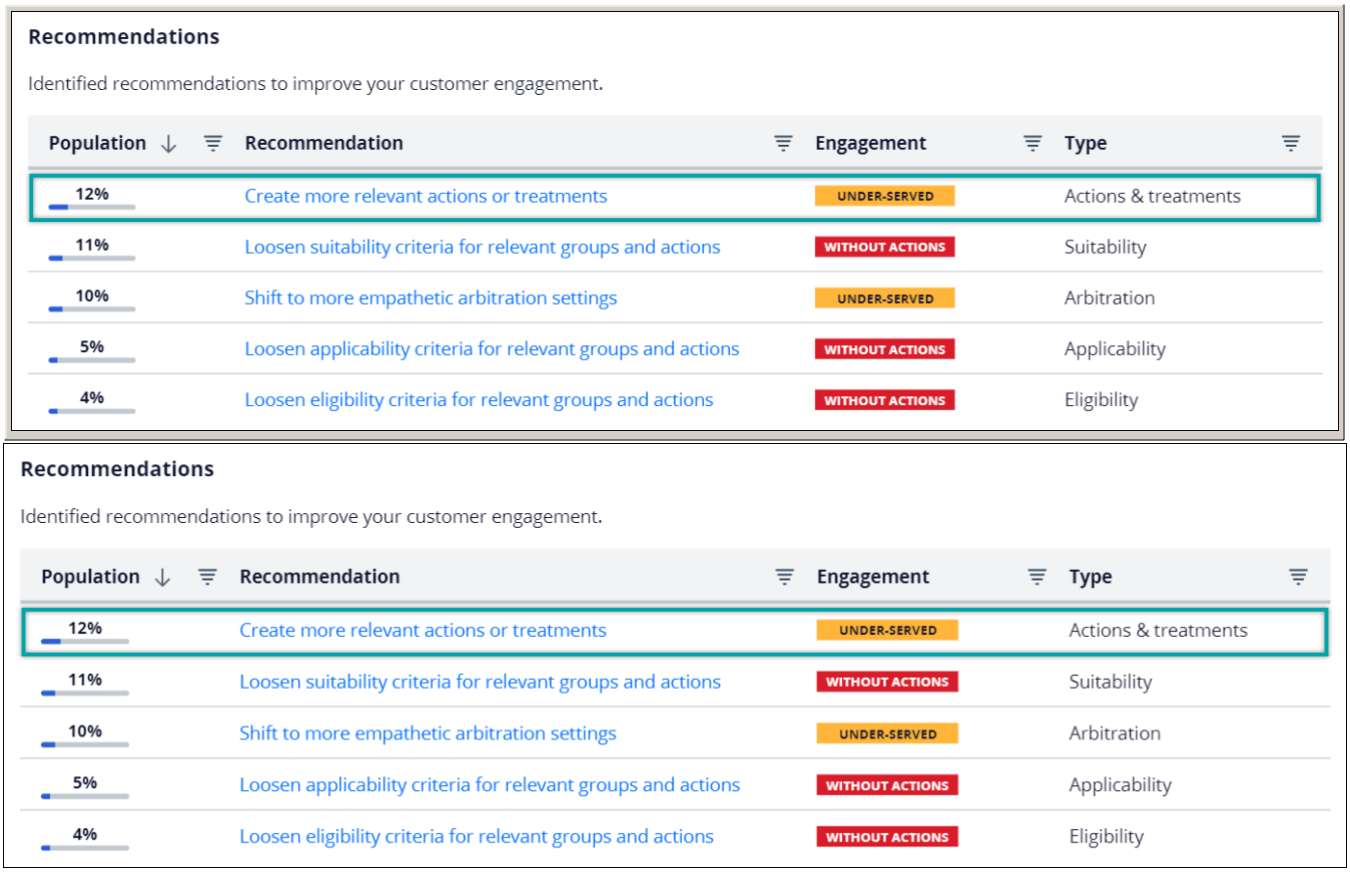
1. Log in to the Pega Customer Decision Hub portal.
2. In the navigation pane of the Pega Customer Decision Hub portal, click **Discovery Value Finder** .
3. Run a simulation.
   For more information, see Running Value Finder simulations.
4. Click the **Properties** icon under **Under-served threshold**.
5. In the **Configure under-served threshold** window, use the slider to select the maximum propensity at which to determine underserved customers.
   Value Finder identifies customers with a propensity that is lower than the configured threshold as underserved, and customers with a higher propensity as well-served.
6. Click **Submit**. **Result:** Value Finder recalculates and refreshes the results of the simulation.

Analyzing group descriptions

Value Finder provides insight about underserved customers by identifying and describing groups that have a high occurrence of underserved customers. Review each group's information to understand why there are no relevant actions available for these customers, and use this knowledge to improve your decision strategy.

Value Finder only provides customer group descriptions for the recommendation **Create more relevant actions or treatments**, as shown in the following figure:

Value Finder recommendations

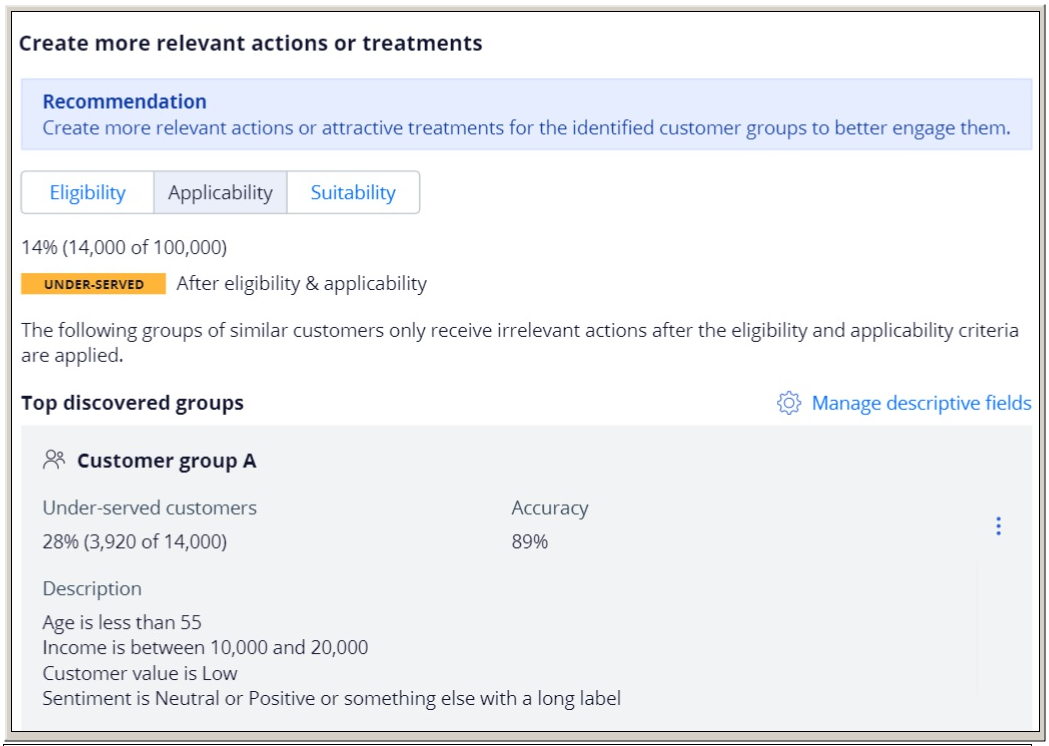


Recommendations
Identified recommendations to improve your customer engagement.
Population
Recommendation
Engagement
Type
12%
Create more relevant actions or treatments
UNDER-SERVED
Actions & treatments
11%
Loosen suitability criteria for relevant groups and actions
WITHOUT ACTIONS
Suitability
10%
Shift to more empathetic arbitration settings
UNDER-SERVED
Arbitration
5%
Loosen applicability criteria for relevant groups and actions
WITHOUT ACTIONS
Applicability
4%
Loosen eligibility criteria for relevant groups and actions
WITHOUT ACTIONS
Eligibility

Pega Customer Decision Hub

1. Log in to the Pega Customer Decision Hub portal.
2. In the navigation pane of the Pega Customer Decision Hub portal, click **Discovery Value Finder** .
3. Run a simulation.
   For more information, see Running Value Finder simulations.
4. In the **Recommendations** section, click the recommendation **Create more relevant actions or treatments**.

Create more relevant actions or treatments





5. In the top-left corner of the **Create more relevant actions or treatments** window, display the number of underserved customers after each stage by clicking **Eligibility**, **Applicability**, or **Suitability**.
   Note: The numbers are cumulative for each consecutive stage of the decision funnel.

   The tabs display the following information:

   Eligibility
   Shows how many customers are underserved after eligibility.
   Applicability
   Shows how many customers are underserved after eligibility and applicability.
   Suitability
   Shows how many customers are underserved after eligibility, applicability, and suitability.

6. In the **Top discovered groups** section, review the groups that Value Finder identified.

   Value Finder provides the following information about each group:

   Description
   Group characteristics, such as age or credit card ownership.
   Under-served customers
   The number of underserved customers in the group.
   Accuracy
   The number of underserved customers in the group divided by the total number of customers in the group, underserved plus well-served. If the accuracy is at 100%, this means that all customers in the group are underserved. In reality, the accuracy is never at 100%. An accuracy of 80% means that 80% of customers in the group are underserved. The remaining 20% have at least one action above the propensity threshold.

7. **Optional:** To get additional information about a group, click the **More** icon, and then click **View details**.

   Value Finder provides the following information for each group:

   Total customers in group
   : The number of customers in the group, both underserved and well-served.

   Percentage of under-served retrieved in this group
   : The number of underserved customers in the group divided by the overall number of underserved customers in the audience at this stage. This measure shows what fraction of all underserved customers in the audience Value Finder retrieves for this group.

8. **Optional:** To get more actionable group definitions that you can later save as audiences, manage the descriptive fields used to define the groups:
   Note: Deselecting fields to exclude them from the descriptions lowers the accuracy of the group descriptions.
   a. Click **Manage descriptive fields**.
   b. If you want to remove a field from the group description, in the **Used fields** list, click the **Close this item** icon next to the field name.
   c. If you want to add a field to the group description, in the **Ignored fields** list, click the **Close this item** icon next to the field name.
   d. Save the changes by clicking anywhere outside the dialog box. **Result:** Value Finder updates the group descriptions by using the selected fields and recalculates the number of underserved customers, accuracy, and recall measures.

Saving customer groups as audiences

Save or export as audiences the customer groups that Value Finder identifies as underserved after eligibility. You can then run distribution tests to get more insight into the current actions that these audiences receive. Use the Value Finder recommendations and your distribution test results as feedback for business stakeholders to inspire them to create new actions and treatments that will be relevant to these audiences.

Pega Customer Decision Hub
Before you begin: Review the customer groups that Value Finder identifies, and then manage the descriptive fields for the groups until you have meaningful and actionable groups from which you can create audiences. For more information, see Analyzing group descriptions.

1. Save a group as an audience by performing the following actions:
   a. To the right of the group description, click the **More** icon, and then click **Save as audience**.
   b. In the **Create audience** window, enter a name for the audience.
   c. Click **Submit**, and then click **Close**.
2. To export an audience as a CSV file, click the **More** icon to the right of the group description, and then click **Export as CSV**. **Result:** The system saves the file to the default download folder on your computer.

- Simulation testing

Testing Next-Best-Action configuration with persona testing

You can design persona-based tests to verify that the Next-Best-Action Strategy gives the expected results. For example, design a customer test persona, and then run a test to verify that the customer receives a specific action with high propensity.

Pega Customer Decision Hub

1. On the **Engagement policy** tab, click the name of any group, or click **All groups** to run the simulation for all issues and groups in your business structure.
2. Click **Actions Persona testing** .
3. If you did not previously configure a persona test, click **Create test**.
4. In the **Create test** window, configure the conditions for the test to pass. **For example:** Select the action and treatment that you expect the test persona to receive according to your Next-Best-Action Strategy.
5. In the **Persona (Data transform)** field, select the persona against which you want to test the strategy. For more information, see Configuring persona tests.
6. In the **Next-Best-Action scope** section, select whether the test should only check engagement policy configuration, or also include additional elements, like constraints and arbitration.
7. Click **Create**.
8. Select the test and click the **Run selected tests** icon.
9. After the persona test finishes, analyze the results to see whether the result matches the expected outcome.

- Configuring persona tests

Configuring persona tests

Use persona tests to define test cases and the expected results to ensure you do not introduce any regressions as you define your engagement policies and arbitration. You can define and run persona tests at each level of your business hierarchy.

Persona-based tests use customer personas with specific characteristics to evaluate your Next-Best-Action strategy results. You can design these personas according to your requirements and the strategy that you are testing. For example, to verify that a Retention action is prioritized for a customer who are at risk of terminating their contract, you can design a customer persona with specific characteristics and check which action the Next-Best-Action strategy selects for them.

Pega Customer Decision Hub

1. On the **Engagement policy** tab, click the name of any group, or click **All groups** to run the simulation for all issues and groups in your business structure.
2. Click **Actions Persona testing** .
3. If you did not previously configure a persona test, click **Create test**.
4. In the **Persona (Data transform)** field, enter the name for the customer persona, for example, Retention Customer, and click the **Open** icon.
5. On the **Definition** tab of the data transform, specify the characteristics of the persona.
   The persona characteristics are defined as properties of the customer, their journey, and other related data. For example, if you have a specific test customer contact that you want to use as a test persona, enter their ID in the **Set .ContactId equal to** row. You can add new properties by right-clicking on a row and selecting **Add child**. For more information, see Configuring a data transform.
6. Click **Create**. **Result:** The persona test is created in the ApplicationName-NBADTests ruleset. Each test is also added to a test suite and to your DevOps pipeline process, if applicable.
7. In Next-Best-Action Designer, continue with the persona test configuration, as described in Testing Next-Best-Action configuration with persona testing.

Creating a data flow for testing Next-Best-Action strategy

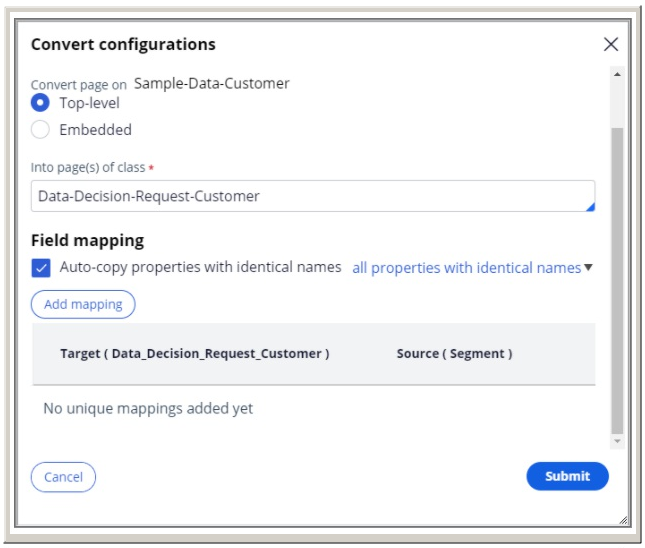
Improve the accuracy of your Next-Best-Action strategy by testing the framework. By creating a data flow which retrieves customer data from your relational database, and then running the data flow, you can verify that the Next-Best-Action strategy framework gives correct results.

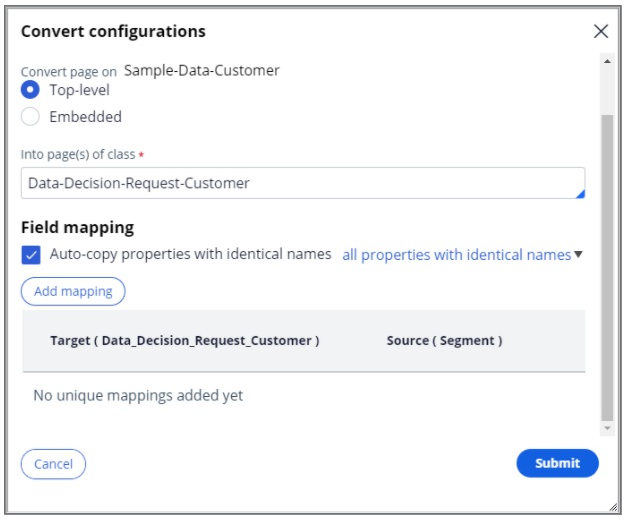
Pega Customer Decision Hub

1. Log in to App Studio as an operator with access to data flows. Only users with an access group that includes the user role *CustomerDecisionHub:DataAdmin* can access data flows.
2. In the navigation panel of App Studio, click **Data Data Flows** .
3. On the *Data Flows* landing page, click **Create**.
4. On the *Create Data Flow* page, enter the data flow description, for example, NBA strategy test.
5. In the **Apply to** field, verify that the class to which this data flow applies is the primary customer class for your application.
   Tip: You can use the Context Dictionary to check the primary customer class for your application. For more information, see Complying with policies or regulations by detecting unwanted bias.
6. Click **Create and open**.
7. On the data flow canvas, right-click the **Source** shape, and then select **Properties**.
8. In the **Import data from** list, select **Data set**.
9. In the **Data set** field, select **Segment**.
10. Click **Submit**.
11. Add a **Convert** shape to the data flow canvas, and then right-click the shape to edit its properties.
12. In the Into page(s) of class field, enter Data-Decision-Request-Customer. This option converts the source data from your customer class into a page of the *Data-Decision-Request-*

Customer class.

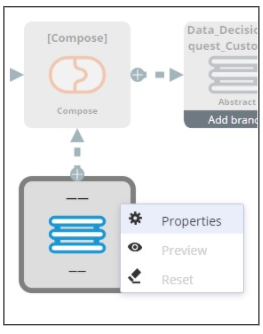
13. Select the **Auto-copy properties with identical names** check box, as in the following example:





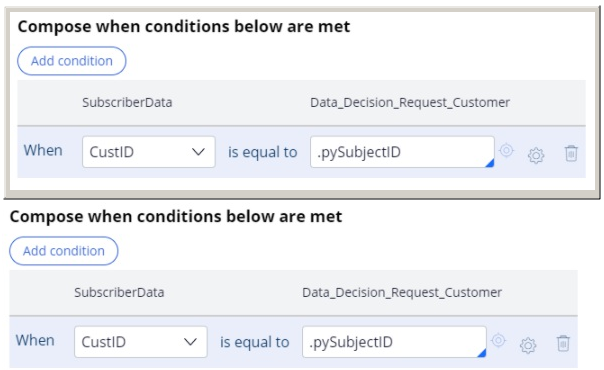
14. Click **Submit**.
15. Add a **Compose** shape to the canvas. The Compose shape pulls in all customer data, including associated data that you configured in the Context Dictionary.
16. Configure the Compose shape:
    a. Right-click the Source shape for the Compose shape, and then click Properties.





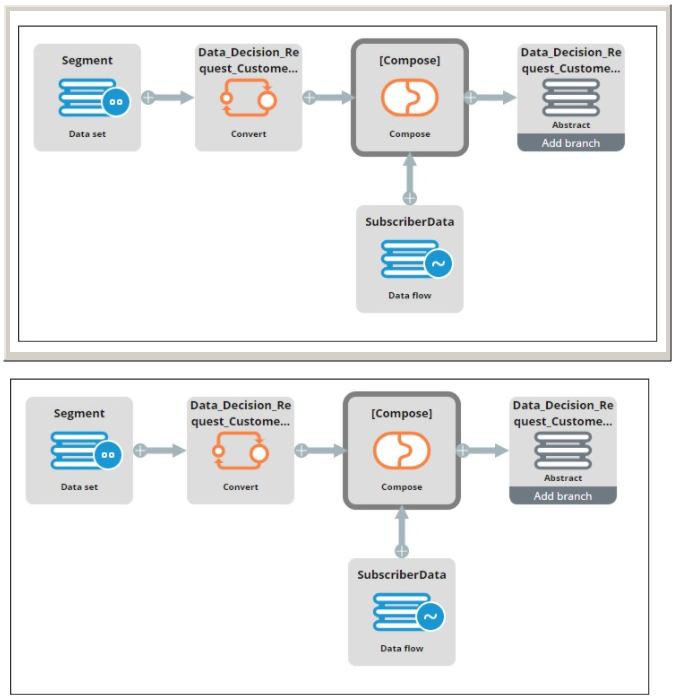
    b. In the **Import data from** list, select **Data flow**.

c. In the **Input class** field, select the primary customer class for your application.
Tip: Use the Context Dictionary to check the primary customer class for your application. For more information, see Complying with policies or regulations by detecting unwanted bias.
d. In the **Data flow** field, select the **CustomerData** data flow for your primary customer class.
Note: The name of the data flow depends on the name of the primary customer context in your application. For example, if your primary customer context is Subscriber, the name for the *CustomerData* flow is *SubscriberData*. For more information, see Complying with policies or regulations by detecting unwanted bias.
e. Click **Submit**.
f. Right-click the **Compose** shape, and then click **Properties**.
g. In the **Compose Data_Decision_Request_Customer with data from** field, select the property that corresponds to your primary customer context. For example, if your primary customer context is Subscriber, select the **.Subscriber** property.
h. In the **Compose when conditions below are met** section, enter the following Join condition: **When** *<your customer class key>* **is equal to** *.pySubjectID* as in the following example:



i. Click **Submit**.

17. Verify that the final shape in the data flow is an Abstract shape which outputs the results, as in the following example:



18. Click **Save**.

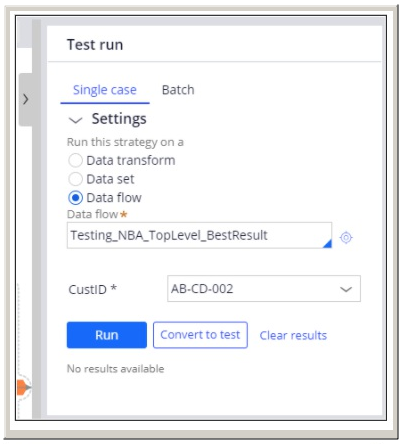
- Testing the Next-Best-Action strategies with data flows

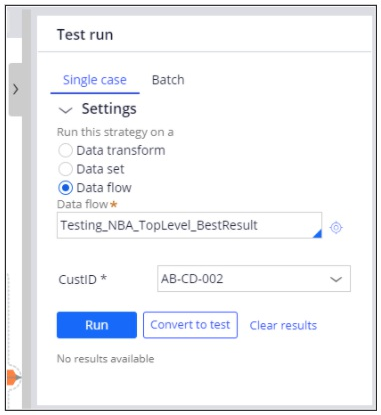
Testing the Next-Best-Action strategies with data flows

After you create a data flow, start a test run panel of your Next-Best-Action strategy. You can test the full strategy framework by running the test at the top level of the Next-Best-Action hierarchy, or test the configuration of a specific business group by running the test only for that group.

Pega Customer Decision Hub

1. In the navigation panel of Dev Studio, click **Records Decision Strategy** .
2. Open the Next-Best-Action strategy that you want to test. For example, to test the full Next-Best-Action strategy framework, open the *NBA_TopLevel* strategy.
3. Open the **Test run** panel by clicking the **Toggle side panel** icon on the right side of the strategy canvas.
4. In the **Single case** tab of the **Test run** panel, select **Run this strategy on a data flow**.
5. In the **Data flow** field, select your new data flow.
6. In the **Customer ID** list, select the ID of a customer.





7. Click **Run**.

Result: The test data flow retrieves the customer data from the database. The Next-Best-Action strategy then uses the data to decide on the Next-Best-Action for the customer based on the customer's eligibilities, Interaction History, and other factors. You can see the decision results directly on the strategy canvas.

Deploying sample production data to a Business Operations environment for testing

Simulation tests provide you with significant insight into how your changes to decision logic affect the results that your strategies produce. However, if tests are based on sample data, the results do not always reflect the effect that your changes have in real-life scenarios. To ensure that simulations are reliable enough to help you make important business decisions, you can deploy a sample of your production data to a dedicated Business Operations environment for testing.

Pega Customer Decision Hub

By running simulations on sample production data that you migrate to the testing environment, you can predict the impact of changes to your decision logic before running them on a live production environment. This functionality helps you optimize and adjust your decision strategies by analyzing and comparing realistic simulation results.

Learn how you can configure the system to migrate sample production data to run reliable simulations:

- Data migration pipeline overview
- Configuring Deployment Manager for data migration pipelines
- Testing your implementation
- Configuring the system for running simulations of decision strategies

Data migration pipeline overview

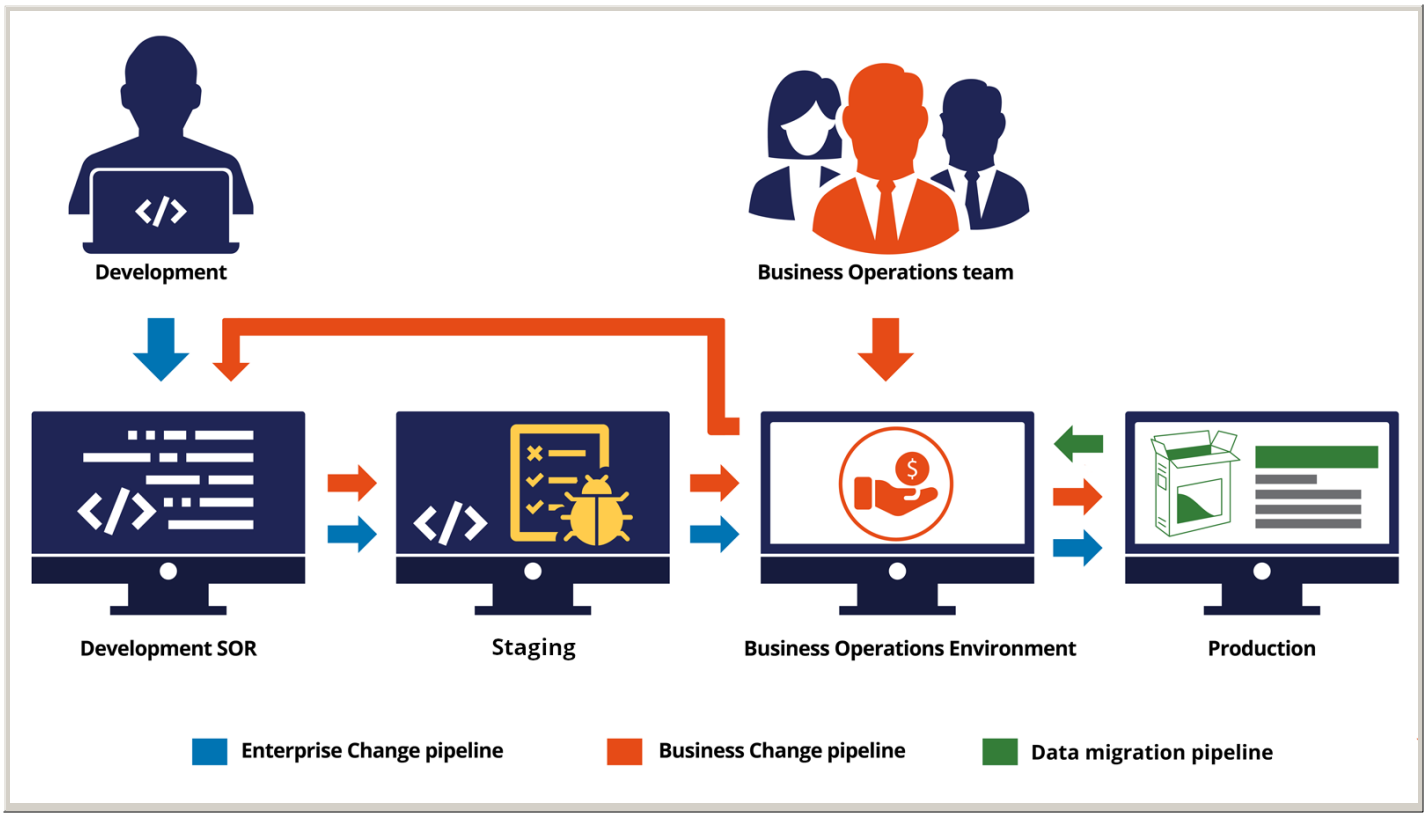
The automated data migration pipeline provides you with the means to test how the changes that you make to decision strategies affect their results, without running these strategies in a live production environment.

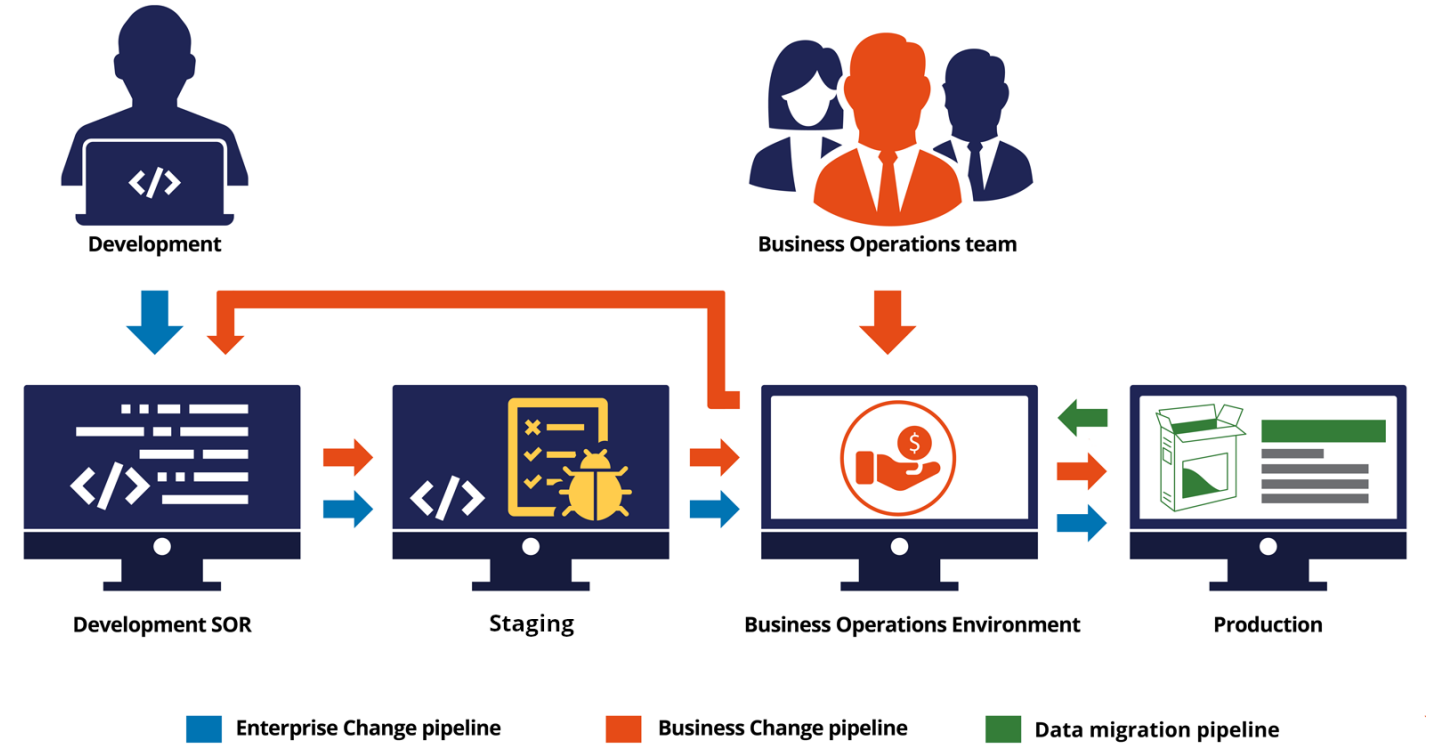
Pega Customer Decision Hub

With a separate system instance that is reserved for simulations, you can sample a selection of real customer and internal Pega Platform data, and then migrate the sample to that dedicated test instance. As a result, the simulations are based on the migrated production data, that is, customer details, adaptive model data, and interaction history, which helps you assess how your changes are reflected in real-life scenarios.

Setting up a data migration pipeline comprises of the following components that together migrate the data according to the process in the following diagram:

- Development environment generating the migration artifacts (enterprise changes)
- Sandbox environment for simulating business changes through Revision Management
- Production environment for collecting customer data samples and sourcing the data for migration to the Business Operations environment
- Business Operations environment for running simulations with the data from the production environment
- Deployment Manager (orchestration server) for running the data migration (future release)





You can run the data migration pipeline in both Pega Cloud services environments, which provide automatic setup of all the data migration settings, and in on-premises configurations. For systems without Deployment Manager, you can run the data migration manually.

For systems with Deployment Manager, you can create pipelines to automatically export data from the production system and import it into the simulation system. For more information about configuring Deployment Manager, see Configuring Deployment Manager for data migration pipelines.

Sample use case

A dedicated Business Operations environment can help you facilitate multiple business processes. Discover a sample scenario of a successful Business Operations environment implementation to learn how you can use simulations based on sample production data to produce better business results. For more information, see Configuring the system for running simulations of decision strategies.

Configuring Deployment Manager for data migration pipelines

Use Deployment Manager to configure and run continuous integration and delivery (CI/CD) workflows for your Pega applications from within Pega Platform. You can create a standardized deployment process so that you can deploy predictable, high-quality releases without using third-party tools.

With Deployment Manager, you can fully automate your CI/CD workflows, including branch merging, application package generation, artifact management, and package promotion to different stages in the workflow. You can download Deployment Manager for Pega Platform from the Deployment Manager Pega Marketplace page.

Pega Customer Decision Hub
Before you begin: Learn about the architecture of a Pega Customer Decision Hub system, to which you want to add Deployment Manager. For more information, see Deployment architecture for Pega Customer Decision Hub. Use Deployment Manager 4.6.x or later in data migration pipelines so that you can automatically export simulation data from a production system and import it into a Business Operations environment.

1. Install or upgrade to Deployment Manager 4.6.x or later.
   - For first-time installations or upgrades from Deployment Manager 3.2.1 on the candidate systems (production and Business Operations environments) and the orchestration server, upgrading is done automatically and you do not need to do post-upgrade steps.

     For more information, see Using Deployment Manager for model-driven DevOps
   - If you are upgrading from a Deployment Manager release earlier than 3.2.1, do post-upgrade steps.

     For more information, see Using Deployment Manager for model-driven DevOps
2. For first-time installations, configure communication between the orchestration server and the candidate systems.
   a. Enable default operators on each system.
      For more information, see Using Deployment Manager for model-driven DevOps.
   b. Configure the authentication profiles, which enable communication between systems, on each system
      Deployment Manager provides default authentication profiles, or you can create your own.
3. **Optional:** To move the orchestration server to a different environment, migrate your pipelines to the new orchestration server, and then configure, on the new orchestration server, the URL of the new orchestration server.

   This URL is used to update the task status on the orchestration server and for diagnostics checks. For more information, see Using Deployment Manager for model-driven DevOps and Deployment Manager FAQ page.

Testing your implementation

Verify whether your environments and pipelines work as designed by running a change management test case and deploying the associated revision to production.

Pega Customer Decision Hub

1. Run a change management test case.
   a. Create a change request in Pega 1:1 Operations Manager and run it through all the stages to approval.
      For example, create or edit an action or treatment.
   b. Check the approved change request in Revision Management.
      Verify if the Next-Best-Action Designer business structure, group level engagement, and arbitration are correct.
   For more information, see the *Pega 1:1 Operations Manager User Guide* on the Pega Customer Decision Hub product page.
2. Deploy the revision to the Business Change pipeline.
   a. Complete the currently active revision.
   b. Promote the completed revision to the Business Change pipeline.
   c. Test the merged revision.
   d. Validate and promote the revision through the pipeline (Deployment Manager 4.8 or later) or manually.
   For more information, see Deploying revisions with Deployment Manager.

Configuring the system for running simulations of decision strategies

When you make changes to your decision data, data flows, and data models, the results of your updates impact your decision strategies. To verify what effects your changes produce on your customer-oriented campaigns, you can simulate the decision process with these updates on sample production data, including customer information, Adaptive Decision Manager input, and interaction history. After comparing the simulation results with your current outcomes, you can decide whether you want to implement your modifications.

Pega Customer Decision Hub

Use case

uPlusTelco is about to release a new line of tablets. Before the launch, the marketing team is determining the demand for the new products among both their current and prospective customers. As the lead marketing strategy specialist in that team, you are responsible for designing and implementing the simulation plan for the products release. To ensure that the results are reliable, you want to use real customer data of your company to test how the offers with new products might be received, to update your global marketing strategy accordingly.

To achieve that goal, simulate how changes to decision strategies impact their outcomes by performing the following procedures:

1. Configuring the data migration settings.
2. Promoting an application in the production and Business Operations environments.
3. **Optional:** Configuring Deployment Manager for data migration pipelines.
4. Export and import data by doing one of the following actions:
   - Automatically, by Deployment Manager: Exporting and importing simulation data automatically with Deployment Manager.
   - Manually: Exporting data from the production environment manually and Importing sampled data into the simulation environment manually.**Important:** If your Pega Customer Decision Hub application is deployed on Pega Cloud environment, you must use Deployment Manager to migrate the sample customer data.
5. Testing the impact of strategy design changes with simulation runs
6. Running a decision strategy simulation on sampled production data.
7. Using sampled production data to test the performance of proposition filters

Configuring the data migration settings

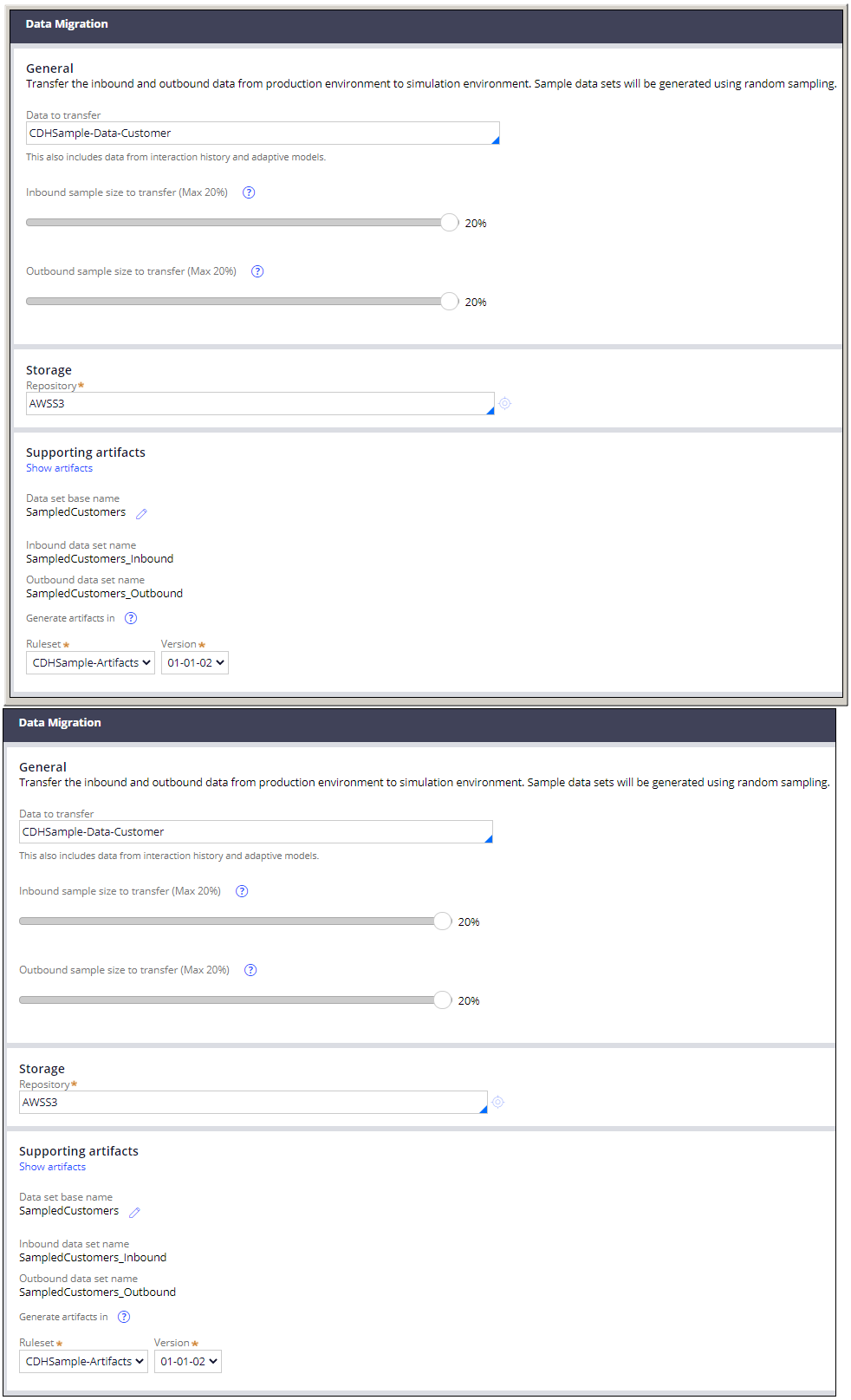
Configure the Dev Studio migration settings to build a framework for your Business Operations environment and to run tests on a sample set of migrated production data.

Pega Customer Decision Hub
Before you begin:

Create a repository for storing the data during migration from the production environment to the Business Operations environment. The repository must be visible by these two environments. For more information, see File and content storage.

1. In the header of Dev Studio, click **Configure Decisioning Infrastructure Data Migration** .
2. On the **Data Migration** landing page, in the **Data to transfer** field, enter the name of the class with customer data that you want to use as the source of production data for simulations, for example, Data-Decision-Request-Customer.
3. **Optional:** Move the **Inbound sample size to transfer** slider to a position that represents the percentage of data that you want to migrate to the Business Operations environment for your inbound sample; the default inbound sample size is 5%.
4. **Optional:** Move the **Outbound sample size to transfer** slider to a position that represents the percentage of data that you want to migrate to the Business Operations environment for your outbound sample; the default outbound sample size is 10%.
   Note: This option is only available in systems with Pega Customer Decision Hub.
5. For on-premises configurations, in the **Storage** section, in the **Repository** field, enter the name of the repository that you want to use to store the data before importing into the simulation environment.
6. In the **Supporting artifacts** section, select an open ruleset and version in which you want to save the artifacts for the data transfer.
7. Confirm the migration settings by clicking **Save**. **Result:** When you save the configuration a set of migration artifacts is generated, including:
   - Sampling data sets and data flows
   - File repository data sets
   - Data export data flows

- Data import data flows



8. In the **Supporting artifacts** section, verify the migration artifacts by clicking **Show artifacts**, and then analyzing the records that were generated.

Result: You have configured the data migration settings and generated sample artifacts. You can now promote the application in the production and Business Operations environments. **What to do next:**

Promote the application in the production and Business Operations environments. For more information, see Promoting an application in the production and Business Operations environments.

To view the main process outline for this tutorial, see Configuring the system for running simulations of decision strategies.

Promoting an application in the production and Business Operations environments

Promote artifacts that you generated from the development system to the production and simulation systems. To make the process efficient, migrate your application by updating the applicable dynamic system settings (DSS).

1. In the navigation panel of Dev Studio, click **Records SysAdmin Dynamic System Settings** .
2. In the list of DSS instances, find and edit the following settings:
   a. To enable inbound sampling in production, find the *simulation/enableSampling DSS* and set the content of **Value** field to True.
   b. To enable outbound sampling in production, find the *PegaMKT-Engine CDHMigrateOutboundSample* and set the content of **Value** field to True.
   c. For on-premise installations, specify the maximum percentage of customers to sample by entering a number in the **simulation/maxSamplingPercentage DSS**. For cloud installations, this setting is configured automatically during provisioning.
3. After configuring the dynamic system settings, export the product file to promote the artifacts from the development environment, and then import the file in the Business Operations environment. When you migrate the application, include the associated data instances. For more information, see Moving applications between systems.

Result: You have configured the dynamic system settings for data migration and promoted the migration artifacts by exporting your application to the Business Operations environment. You can now migrate the data to the simulation environment. **What to do next:**

Migrate the sample customer data that you collect, to use in tests. Manually export the data from the production environment or automatically export data and import it with Deployment Manager. **Important:** If your Pega Customer Decision Hub application is deployed on Pega Cloud environment, you must use Deployment Manager to migrate the sample customer data. For more information, see one of the following articles:

- Exporting and importing simulation data automatically with Deployment Manager.
- Exporting data from the production environment manually.

To view the main process outline for this tutorial, see Configuring the system for running simulations of decision strategies.

Exporting and importing simulation data automatically with Deployment Manager

You can create and run data migration pipelines in Deployment Manager to automatically export simulation data from the production environment into the Business Operations environment.

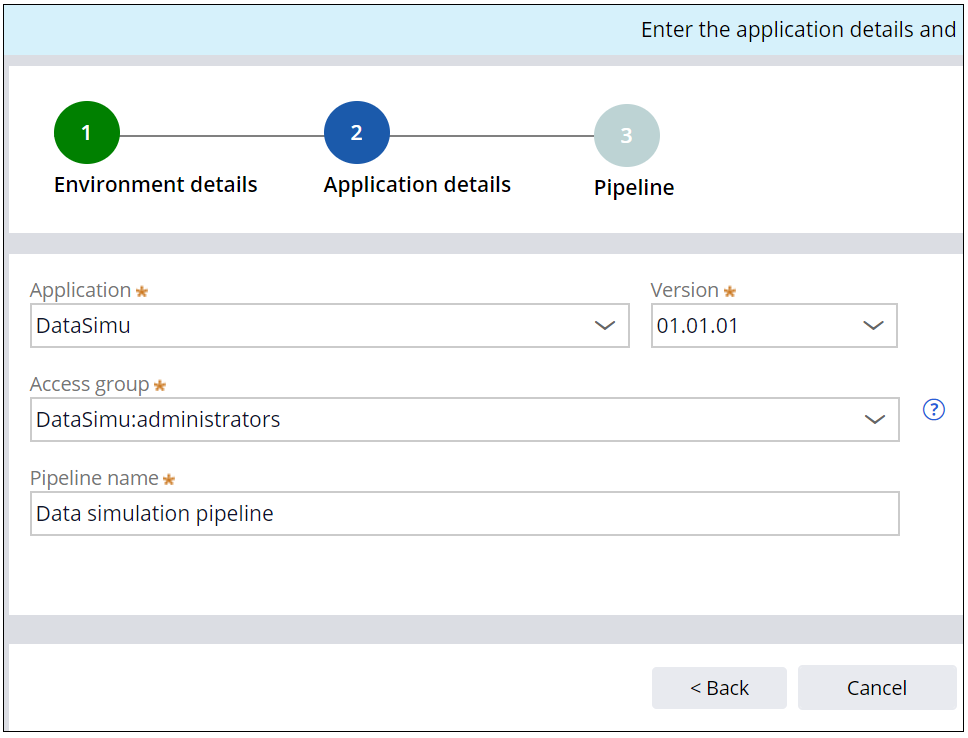
Important: If your Pega Customer Decision Hub application is deployed on Pega Cloud environment, you must use Deployment Manager to migrate the sample customer data.
Pega Customer Decision Hub
Before you begin:

Ensure that you have installed and configured Deployment Manager on the production and Business Operations environments and the orchestration server. For more information, see Using Deployment Manager for model-driven DevOps.

1. In the navigation pane of Deployment Manager, click **Pipelines Data migration pipelines** .
2. Click **New**.
3. If you are using Deployment Manager on-premises, on the Environment Details page,configure environment details.

   This information is automatically populated if you are using Deployment Manager in Pega Cloud Services environments, but you can change it.

   a. In the **Environment** fields, enter the URLs of the production and Business Operations environments.
   b. If you are using your own authentication profiles, in the **Auth profile** lists, select the authentication profiles that you want the orchestration server to use to communicate with production and Business Operations environments.
   c. Click **Next**.
4. On the Application details page, configure the application information for which you are creating the pipeline.
   a. In the **Application** list, select the name of the application.
   b. In the **Version** list, select the application version.
   c. In the **Access group** list, select the access group for which you run pipeline tasks. This access group must be present on the production and Business Operations environments and have at least the sysadmin4 role.
   d. In the **Name of the pipeline** field, enter the pipeline name.
   e. Click **Next**.
5. Click **Finish**.



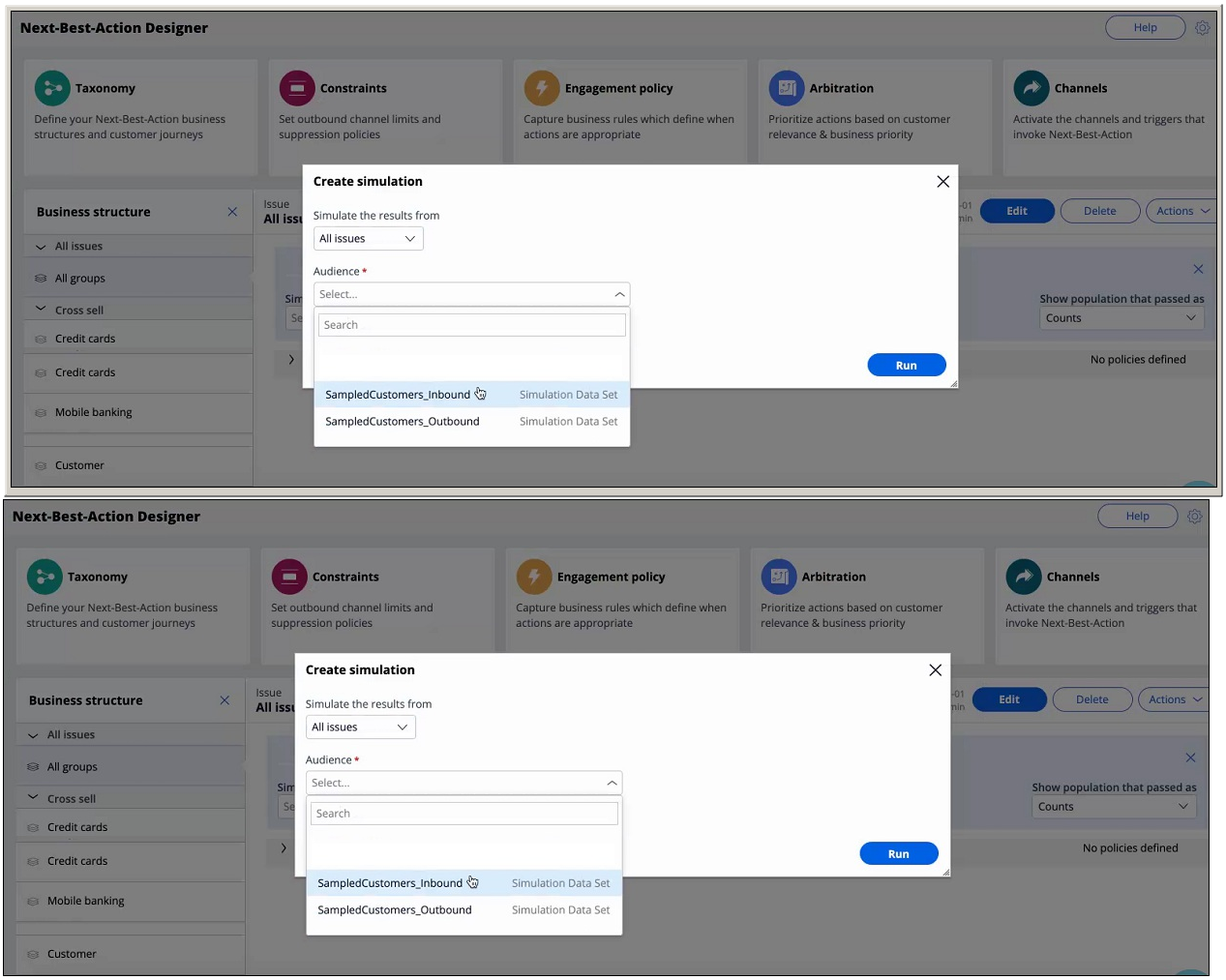
6. Start a pipeline by doing one of the following actions:
   - If the pipeline for which you want to run a data simulation is open, click **Start data migration**.
   - If the pipeline is not open, click **Pipelines Data migration pipelines** , and then click **Start data migration**.
7. In the **Start data migration** dialog box, click **Yes**. **Result:** You have created a data migration pipeline in Deployment Manager and automatically migrated sample data from the production to the Business Operations environment. You can now verify how the changes that you made to the decisioning framework impact the results of your strategies and prediction outcomes by running tests using that data.

What to do next:

In your Business Operations environment, run tests on the sample data that you imported. For more information, see Running a decision strategy simulation on sampled production data.

To view the main process outline for this tutorial, see Configuring the system for running simulations of decision strategies.

Tip: You will see the inbound and outbound data sets while creating a simulation in Next-Best-Action Designer. When you choose the **Audience**, the *Simulation Data Set* will be visible in the list, as in the following figure:



Exporting data from the production environment manually

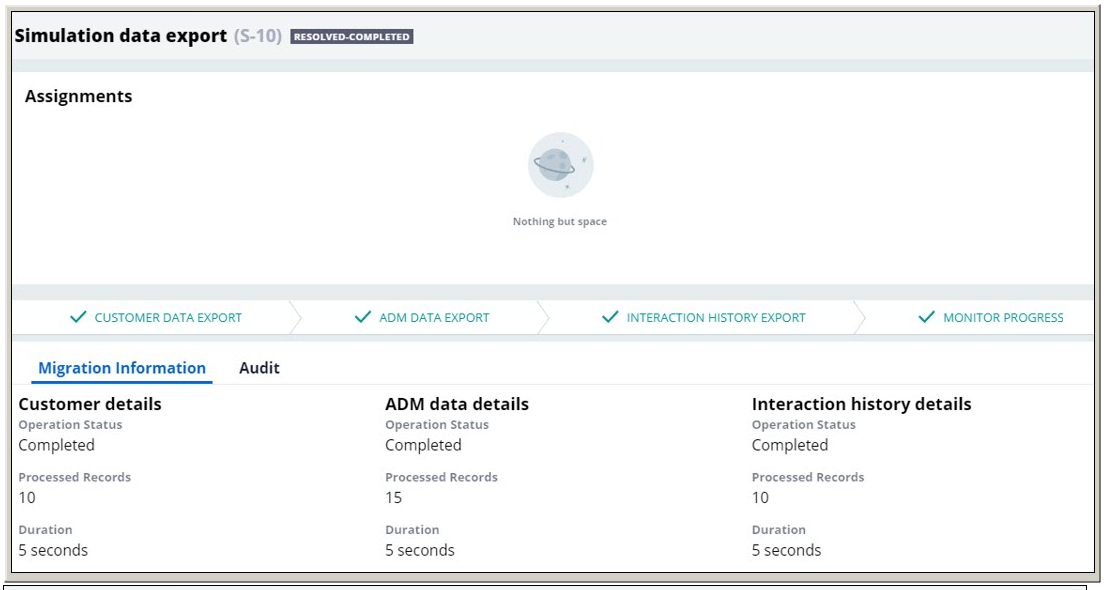
In configurations without Deployment Manager, after collecting the sample data, trigger the simulation data export case type in the production environment.

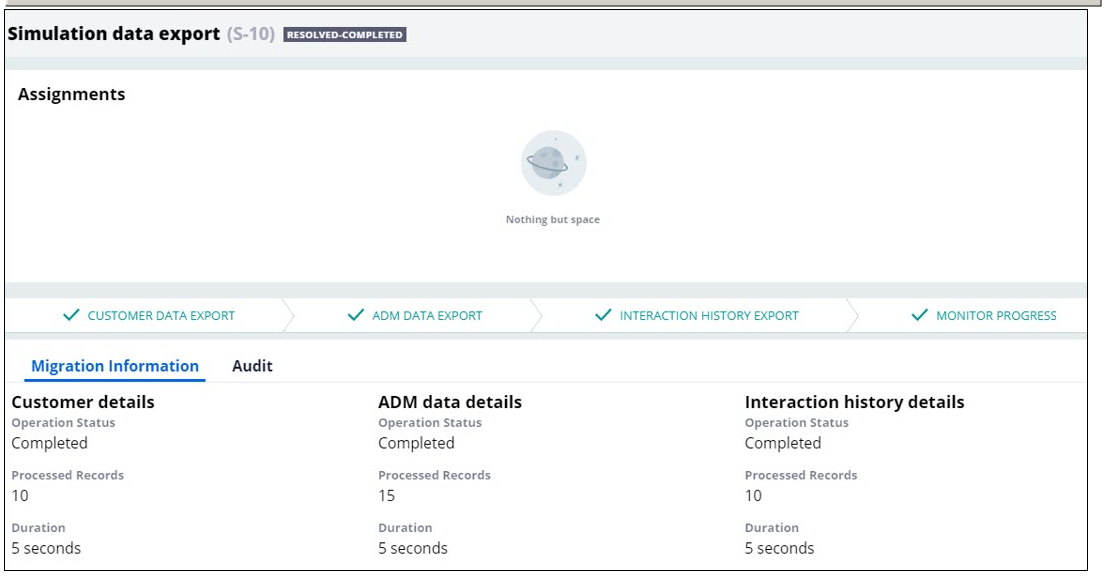
Pega Customer Decision Hub
Before you begin: The file data sets that are included in data migration use a default custom stream processing algorithm. In configurations without Deployment Manager, either disable or update that algorithm to use your own custom implementation.

1. In your production environment, in the navigation panel of Dev Studio, click **Case types Simulation data export** .
2. In the **Edit case type** work area, click **Save and run**.

   A simulation data export case is initiated to export the sample customer data, ADM data, and interaction history data.

3. Monitor the new case at regular intervals until all the data is exported, as in the following example:





4. Optional: When the case is resolved, check what records were exported by clicking the **Audit** tab, and then analyzing the list of instances.

Result: You have exported your sample data to the migration data set. You can now import the data to the Business Operations environment for testing.**What to do next:**

In your Business Operations environment, import the production data. For more information, see Importing sampled data into the simulation environment manually.

To view the main process outline for this tutorial, see Configuring the system for running simulations of decision strategies.

Importing sampled data into the simulation environment manually

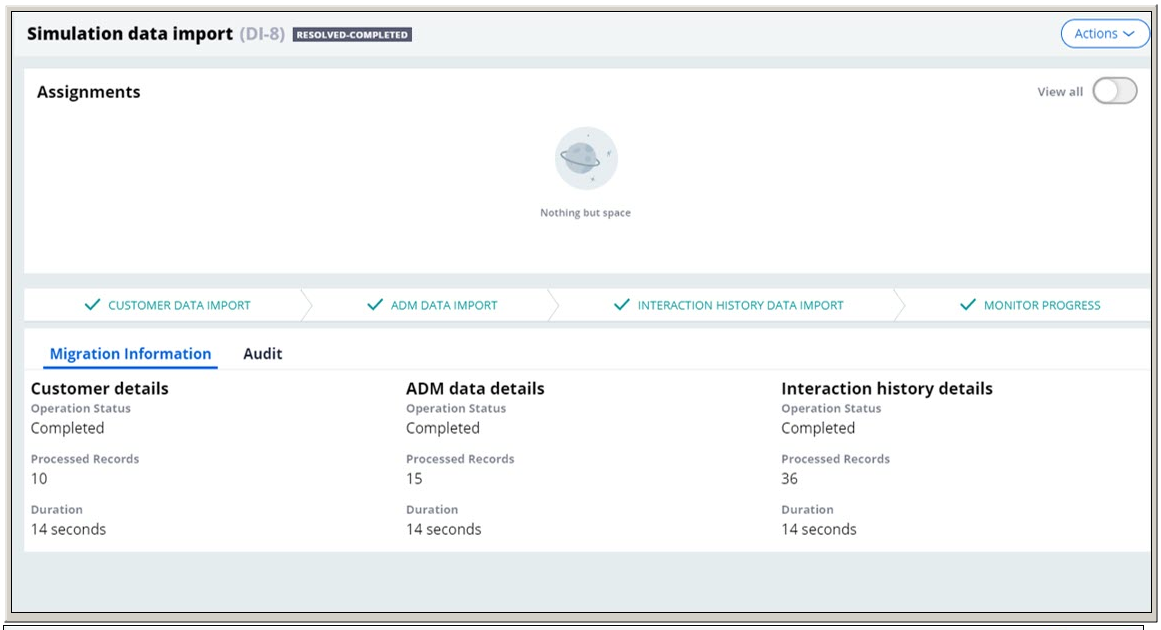
In configurations without Deployment Manager, after exporting the data, trigger the simulation data import case type in the simulation environment.

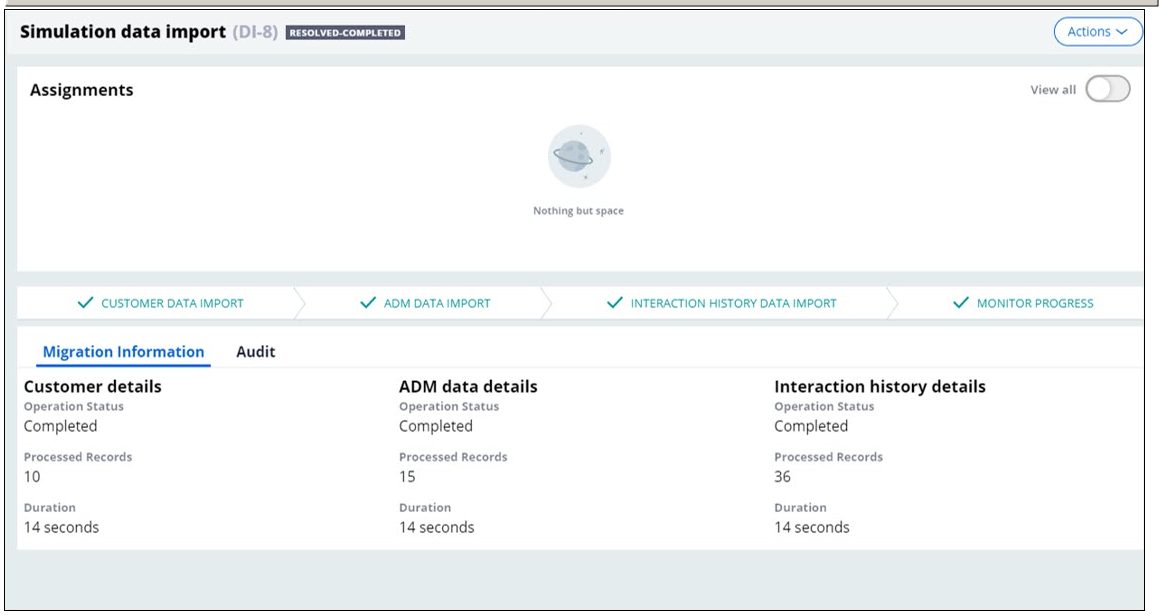
Pega Customer Decision Hub
Before you begin: The file data sets that are included in data migration use a default custom stream processing algorithm. In configurations without Deployment Manager, either disable or update that algorithm to use your own custom implementation.

1. In your simulation environment, in the navigation panel of Dev Studio, click **Case types Simulation data import** .
2. In the **Edit case type** work area, click **Save and run**.

   A simulation data import case is initiated to import the sample customer data, ADM data, and interaction history data.

3. Monitor the new case at regular intervals until all the data is imported and the case is resolved, as in the following example: **For example:**





Result: You have migrated your sample data to the simulation environment. You can now verify how the changes that you made to the decisioning framework impact the results of strategies and prediction outcomes by running tests with that data.**What to do next:**

In your simulation environment, run tests on the sample data that you imported. For more information, see Running a decision strategy simulation on sampled production data.

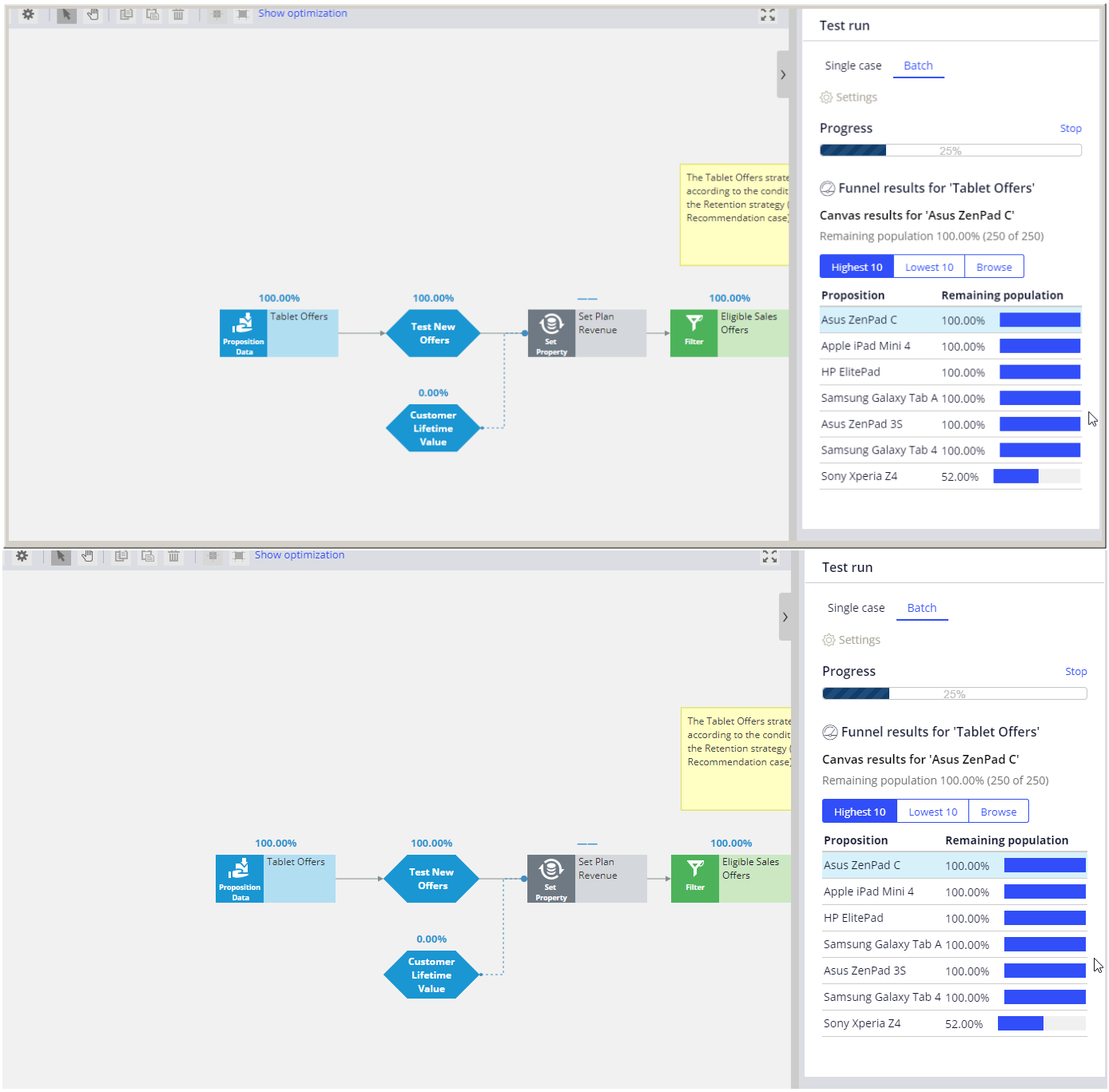
To view the main process outline for this tutorial, see Configuring the system for running simulations of decision strategies.

Testing the impact of strategy design changes with simulation runs

You can use a batch case run to test the performance of your strategy as you design it, and identify which components need optimization. As you edit the strategy, run a simulation to identify the most popular propositions, check whether customers are receiving actions, and make sure that your strategy is executed as intended.

Pega Customer Decision Hub

1. In your Business Operations environment, search for and open a Strategy rule.
2. On the right side of the strategy canvas, expand the **Test run** panel.
3. In the **Test run** panel, click the **Batch** tab.
4. Click **Settings**.
5. In the **Analyze** section, click **Decisions**.
6. Select the simulation test that you want to run from the following options:
   - Select **An existing simulation test**, and then select the simulation ID.**Tip:** To check the configuration of the simulation test, click **Open**.
   - Select **A new simulation test** and specify the data flow, data set, or report definition that you want to use as the input data of the simulation test.
7. Click **Run**. **For example:** For each supported component, the canvas displays the decision results as a percentage of the remaining population. That is, the results show the percentage of the audience that can receive a specific proposition, as well as the impact that each component has on the strategy results. Some components are not included in decision simulation tests. For example, in the figure below, the Set Property component is not part of the test, and thus has no corresponding results on the canvas. Unsupported decision components include adaptive and predictive models, scorecards, decision trees, and decision tables.



Result:

You have run a simulation to test how the changes that you made impact your decision strategy. You can now decide whether you want to implement the changes on a live system.

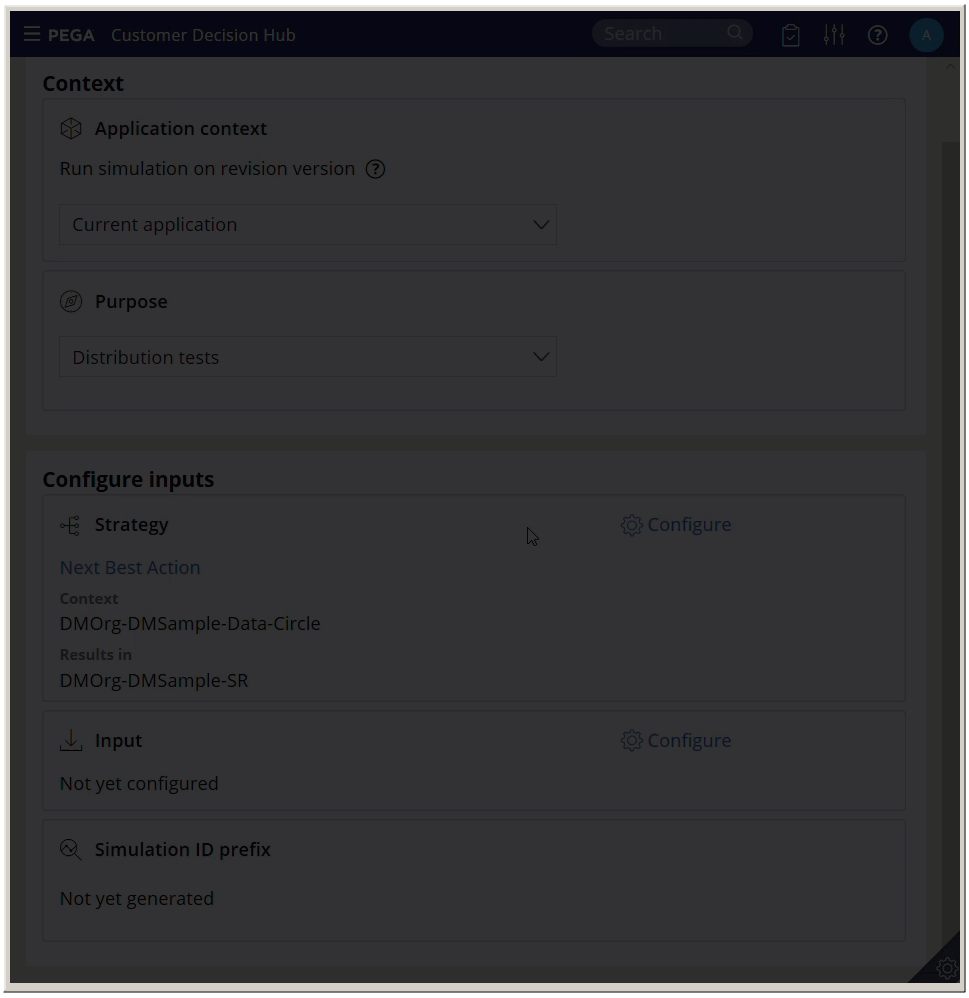
To view the main process outline for this tutorial, see Configuring the system for running simulations of decision strategies.

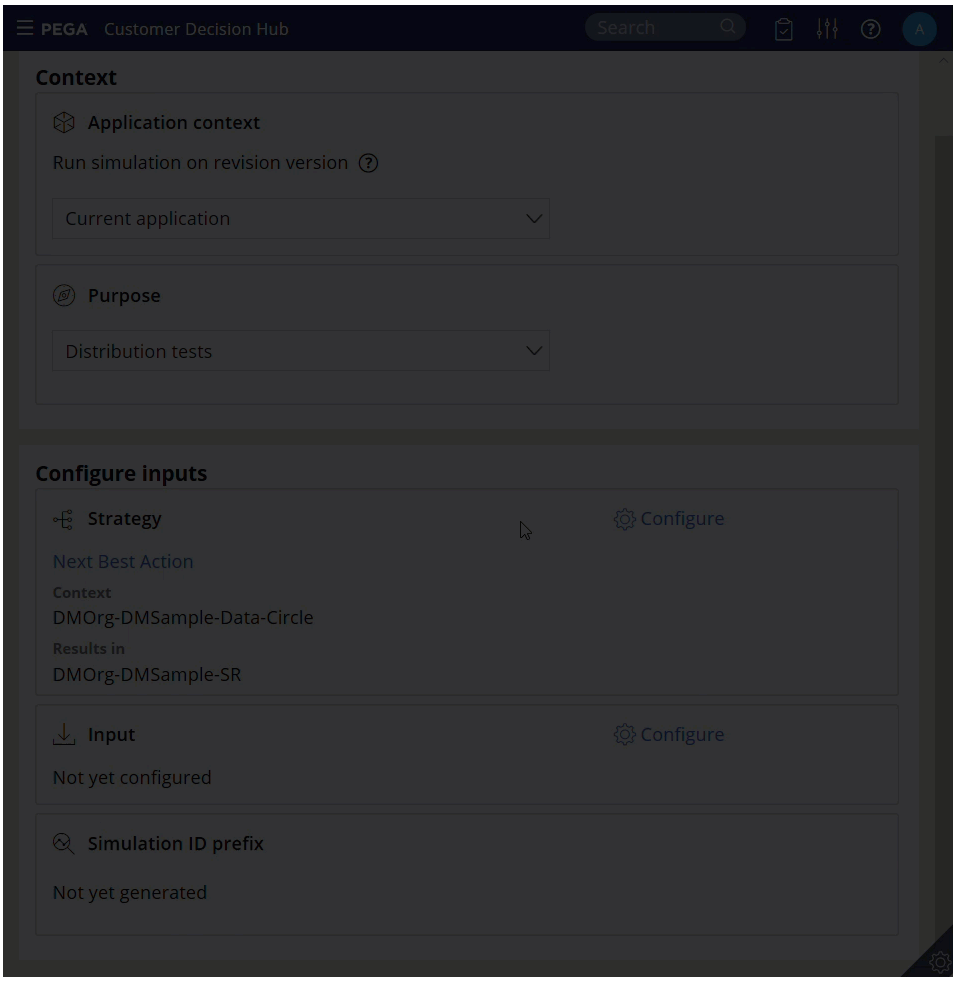
Running a decision strategy simulation on sampled production data

After migrating the data, run simulations on the sample production data that you imported. You can use the customer sampling data set as a source to run simulations in the Pega Customer Decision Hub portal.

Pega Customer Decision Hub

1. In your Business Operations environment, in the navigation pane of the Pega Customer Decision Hub, click **Simulation testing**.
2. On the Simulation Testing landing page, open a new simulation instance by clicking **Create**.
3. In the **New Simulation Test** work area, configure the simulation settings:
   a. In the **Context** section, in the **Application context** list, select the application revision version on which you want to run the simulation.
   b. In the **Purpose** list, select the reason why you want to run the simulation.
4. In the **Configure inputs** section, define the data sources for your simulation:
   a. In the **Strategy** panel, click **Configure**.
   b. In the **Strategy** dialog box, select the strategy on which you want to run the simulation by clicking **Add** in the applicable row, and then clicking **Apply**.
   c. In the **Input** panel, click **Configure**.
   d. In the **Input** dialog box, select the data set that contains the sample artifacts that you generated in Dev Studio by clicking **Add** in the applicable row, and then clicking **Apply**. **For example:** Select **SampledUniqueCustomers**, as in the following graphic:


PEGA Customer Decision Hub
Search
Context
Application context
Run simulation on revision version
Current application
Purpose
Distribution tests
Configure inputs
Strategy
Configure
Next Best Action
Context
DMOrg-DMSample-Data-Circle
Results in
DMOrg-DMSample-SR
Input
Configure
Not yet configured
Simulation ID prefix
Not yet generated



e. **Optional:** In the **Simulation ID prefix** panel, change the default prefix for the simulation name by clicking the **Edit** icon, entering your custom name, and clicking **Done**. For example, enter: NextBestAcSampledUni.

5. In the **Assign output destinations** section, specify where you want to save the output of your simulation:
   - To select an existing destination, click **Add existing**, click **Add** in the applicable row, and then click **Apply**.
   - To add a new destination, click **Create new**, specify the name and type of the new output, and then click **Done**.

     For example, you can save your simulation results in a database table or in Visual Business Director. The output name cannot contain any special characters.

6. **Optional:** To clear the results of previous simulations tests in the outputs for every run, select the **Overwrite outputs from previous simulations** check box.
7. In the **Assign reports to outputs** section, define which reports you want to link to the outputs:
   a. In the **Reports** panel, click **Configure**.
   b. In the **Assign reports to outputs** dialog box, select an output, a report category, and a report that you want to assign to the output, and then click **Done**.
8. On the Simulation Settings landing page, click **Submit and Run**, and then monitor the progress of the simulation.

Result: You have run a simulation to test how the changes that you made impact your decision strategy. You can now decide whether you want to implement the changes on a live system.**What to do next:** After the simulation run completes, in the **Assigned reports** section, open the output of the simulation, such as a report or a Visual Business Director graph, as in the following example:



Compare the results with the outcome of the original strategy before the changes to help you verify whether the updates produce the results that you expected.

To view the main process outline for this tutorial, see Configuring the system for running simulations of decision strategies.

Using sampled production data to test the performance of proposition filters

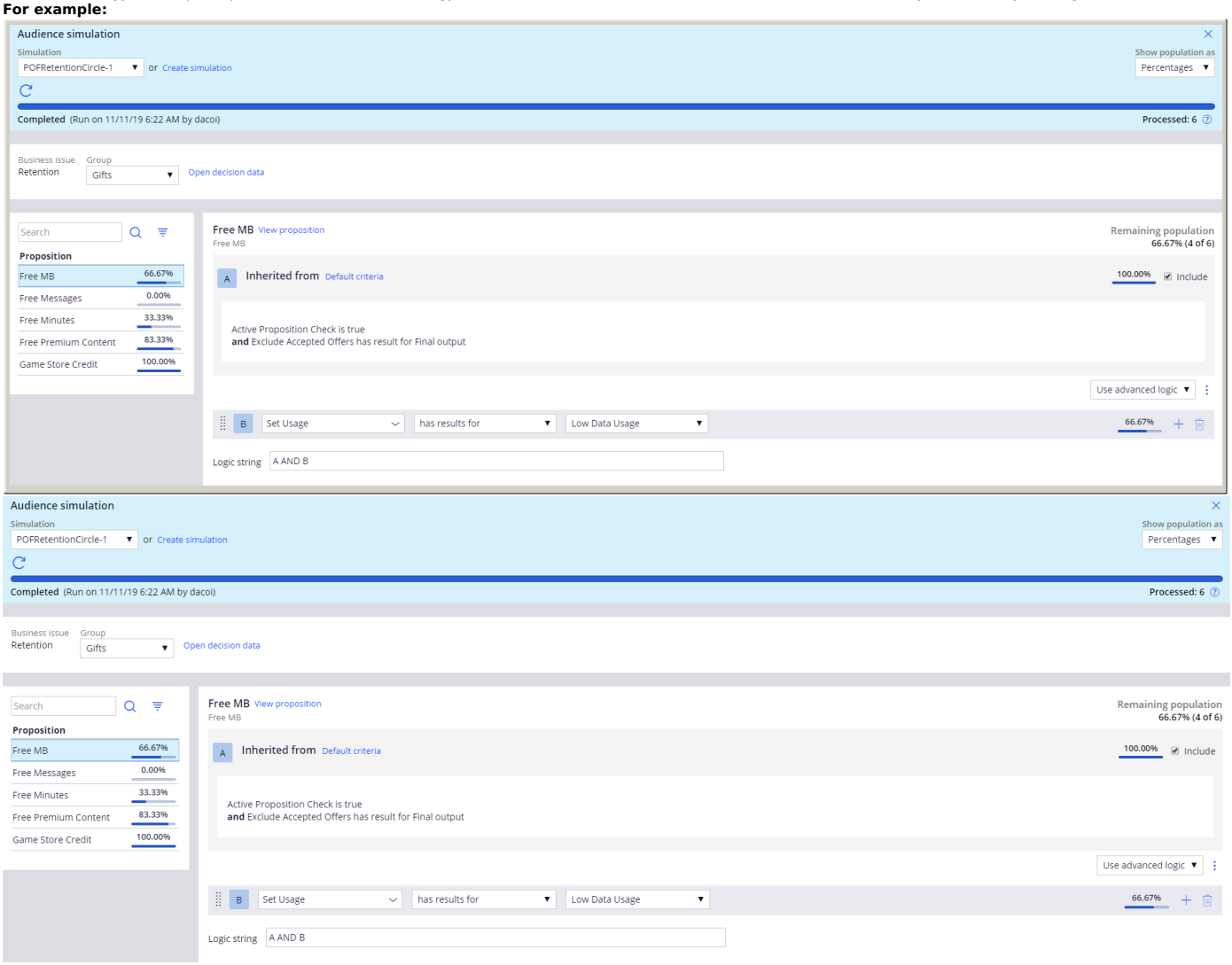
You can use sampled production data to improve the performance of your proposition filters by testing them against simulated audiences. In this way, you can check how many potential actions are filtered out by each component of the filter, and discover if a particular filter criterion is too broad or too narrow for your requirements.

Pega Customer Decision Hub

1. In your Business Operations environment, in the navigation pane of Dev Studio, click **Records**.
2. Expand the **Decision** category, and then click **Proposition Filter**.

3. In the **Proposition Filter** tab, open or create an instance of the Proposition Filter rule:
    - To open an existing instance of the Proposition Filter rule, in the **Instance name** column, select one of the available rules, for example, **EligibleSalesOffers**.
    - To create an instance of the Proposition Filter rule, click **Create**.
4. In the tab of the selected Proposition Filter rule, click **Actions Audience simulation** .
5. In the **Audience simulation** section, select or create a simulation with which you want to test the proposition filter:
    - To use an existing simulation, select it in the **Simulation list**.
    - To create a new simulation, click **Create simulation**, and then select the audience to use for the simulation. If the proposition filter criteria include a When condition, or a strategy that requires parameters, select the Strategy rule that the simulation uses to evaluate the condition. Confirm your choice by clicking **Run**.

For example:



6. In the **Show population that passed as** list, select whether the results should be displayed as a percentage of the starting population, or as a count of audience members.
7. After the simulation test finishes, analyze the results to see what part of the audience would receive each action according to the current proposition filter criteria:
    - To view the results for a specific group, select the group in the **Group** list.
    - To view the results for a specific proposition, in the **Proposition** section, click the proposition name, and then analyze the details on the right side of the screen.

    For each component of the Proposition Filter rule, the simulation test shows percentage values that indicate the percentage of the selected audience that receives the proposition based on the current criteria. For example, if the result for a criterion that checks if the proposition is active is 100.00%, then no audience members were filtered out by this component.

Result:

You have run a simulation to test how the changes that you made impact your proposition filter. You can now decide whether you want to implement the changes on a live system.

To view the main process outline for this tutorial, see Configuring the system for running simulations of decision strategies.

Simulation testing

By running simulation tests in Pega Customer Decision Hub, you can derive useful intelligence that can help you make important business decisions. For example, you can examine the effect of a new product offer or assess risk in a variety of marketing or nonmarketing scenarios.

Pega Customer Decision Hub

With simulation tests, you can run strategies of varying complexity on a preselected sample set of customers. By doing so, you can make millions of decisions at the same time and simulate the outcome of your decision management framework. After a simulation test has been completed, you can visualize the results in Scenario Planner, where you can check whether the new strategy produces the expected result. For example, you can check whether customers are offered a new phone or Internet plan when certain conditions that are specified in the strategy are met. You can also assess the effect of the new product on your existing product offering.

In Pega Customer Decision Hub, you can run simulation tests with minimal amount of configuration. For example, you do not have to configure any simulation data flows or data set where the simulation result will be stored.

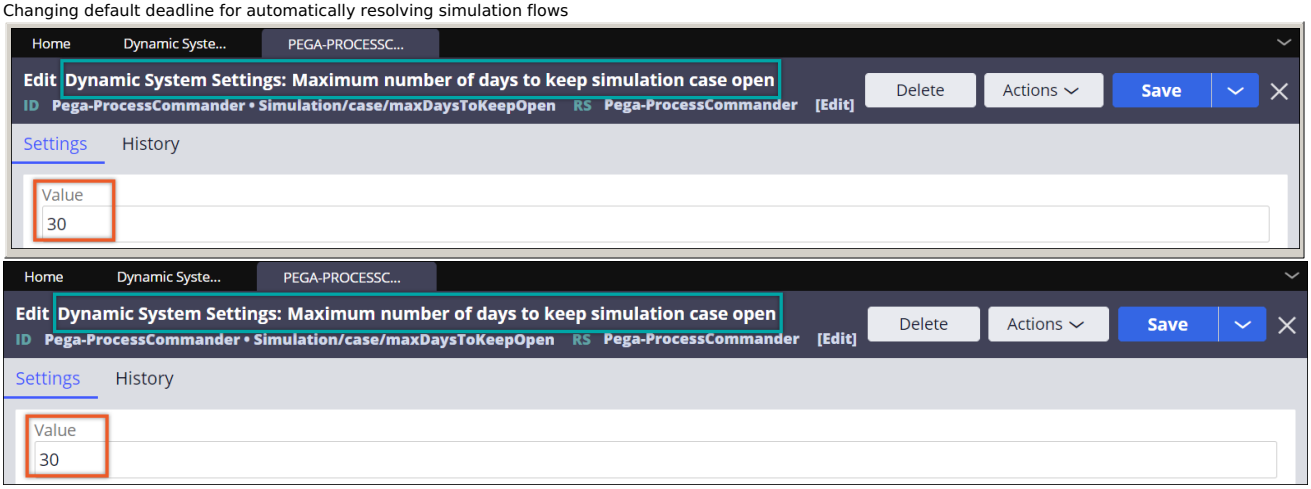
Additionally, you can perform various operations on already completed simulation tests, such as assigning additional reports to a simulation test or comparing simulation tests in Scenario Planner. You can also schedule a simulation test to run in the future. To evaluate your new strategies on the spot, you can simulate strategies directly from the Strategy canvas.

Understanding the simulation flow

Simulations progress through the following stages in Pega Customer Decision Hub:

1. *Setup* - In this stage, the simulation settings are configured.
2. *Execute* - In this stage, the simulation runs.
3. *Review* - In this stage, the simulation results are available for your review.
4. *Resolve* - This stage signifies that the simulation is completed and the results have been reviewed.**Note:** Simulation flows can be resolved manually or automatically. Simulation flows will be resolved automatically once the deadline (30 days, by default) is reached. For the simulation flows that are resolved automatically, the simulation data flow will be cleaned up and the case status will change to Resolved-Completed. To modify the default deadline, go to **Dev Studio Records Dynamic System Settings** and open the *Simulation/case/maxDaysToKeepOpen* ruleset. The value determines the deadline in days.

Changing default deadline for automatically resolving simulation flows



Simulation tests and the decision funnel

You can use simulation tests to understand the decision funnel, that is, the effect of the components that influence the outcome of a decision. Decision funnel explanation simulations break down the results of a decision funnel into granular analyses that help you understand how certain components of a decision influence the overall outcome. This type of simulation test includes predefined reports that drill down to the details of proposition counts by proposition filter, prioritization, switch and champion-challenger components.

To view the proposition count breakdown for a decision, select **Decision funnel** when you create a simulation.

Using simulation tests to discover unwanted bias

You can run simulation tests in order to detect unwanted bias that may be present in your strategy results. For example, you can test whether your strategies generate biased results by sending more actions to female rather than male customers.

Value Finder simulations

Run a Value Finder simulation on an audience to discover which stages of your next-best-action strategy leave customers without any actions or with only low propensity actions, and identify groups of customers that are underserved.

Simulation tests and revision management

If the environment where you want to simulate strategies is enabled for revision management, you can run simulation tests in the context of a specific revision. In revision management, simulation outputs are created as part of enterprise application rulesets, instead of revision branches or revision rulesets. Therefore, simulation outputs are not available for packaging.

Additionally, you can run simulation tests for strategies that are part of unsubmitted change requests. This option is available only when you do not simulate a strategy against a specific revision.

- Sampling audiences for simulations
- decision funnel test
- Running simulation tests
- Running simulation tests from the strategy form
- Managing simulation tests in Pega Customer Decision Hub
- Testing Next-Best-Action configuration with audience simulations
- Complying with policies or regulations by detecting unwanted bias

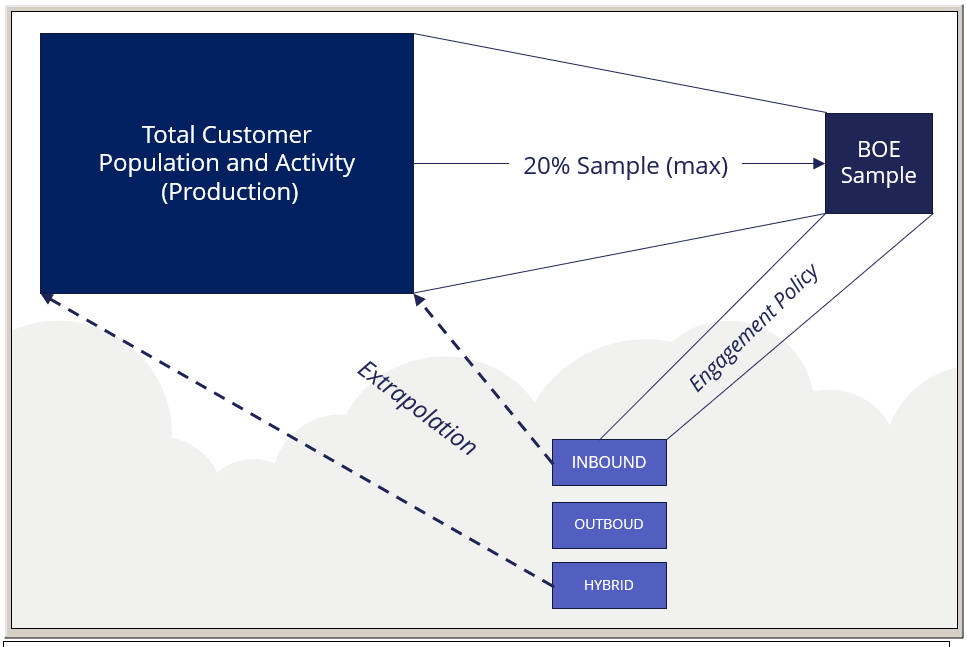
Sampling audiences for simulations

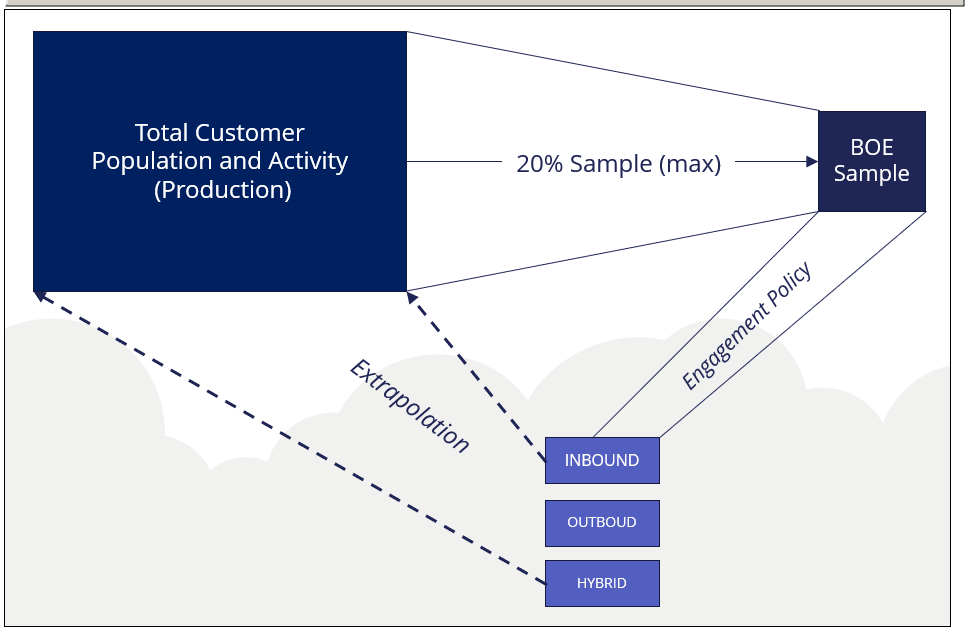
The audience is a set of individuals, that you are targeting with your marketing message. Sampling is the process that allows you to focus on a representative subset of a large population, and therefore you do not have to analyze large groups of subjects. You can also use sampling to reduce the amount of time needed to run a simulation that delivers accurate projections when extrapolated across the entire population.

Pega Customer Decision Hub

Pega Customer Decision Hub provides separate inbound and outbound samples of your production customer data that you can use as the audience input for a simulation. Sampled customer data is captured in your Production Environment and migrated to the Business Operations Environment (BOE) by a data migration pipeline. The inbound and outbound sample datasets in the BOE can be refreshed at scheduled intervals. For more information about scheduling data migration, see: Configuring the system for running simulations of decision strategies.

Important: Pega Customer Decision Hub follows best practice guidelines for business change management. Because of that, you should only run simulation testing in a dedicated Simulations or Business Operations Environment.





Inbound sample

The Pega Customer Decision Hub inbound sampling algorithm randomly samples up to 20% of your inbound decision requests (default sample size is 5%). The final inbound sample includes only one random interaction per subjectID (for example, account, customer). If you use a multilevel customer entity, the sampling algorithm captures all the entity data required to decision the next best action for the sampled request.

For example, if a decision for the head of household requires customer data for all members of the household, then customer data for all members of the household and their devices will be sampled so the interaction can be replayed in a BOE simulation. In addition to customer data, the container payload, other inbound contextual data, and Interaction History for the customer are all captured and migrated to the BOE as a part of the inbound sample.

Outbound sample

The Pega Customer Decision Hub outbound sampling algorithm randomly samples up to 20% of your customer data from production (default sample size is 10%). If you use a multilevel customer entity, the sampling algorithm samples at the top level (for example, account) and includes all underlying entity data (for example, customer, device) in the sample. In addition to customer data, the Customer Insights and Interaction History for the customer all captured and migrated to the BOE for the outbound sample. For more information, see Enabling the migration of historical data for Scenario Planner.

Note: To enable and configure inbound and outbound sampling, see Configuring the data migration settings.

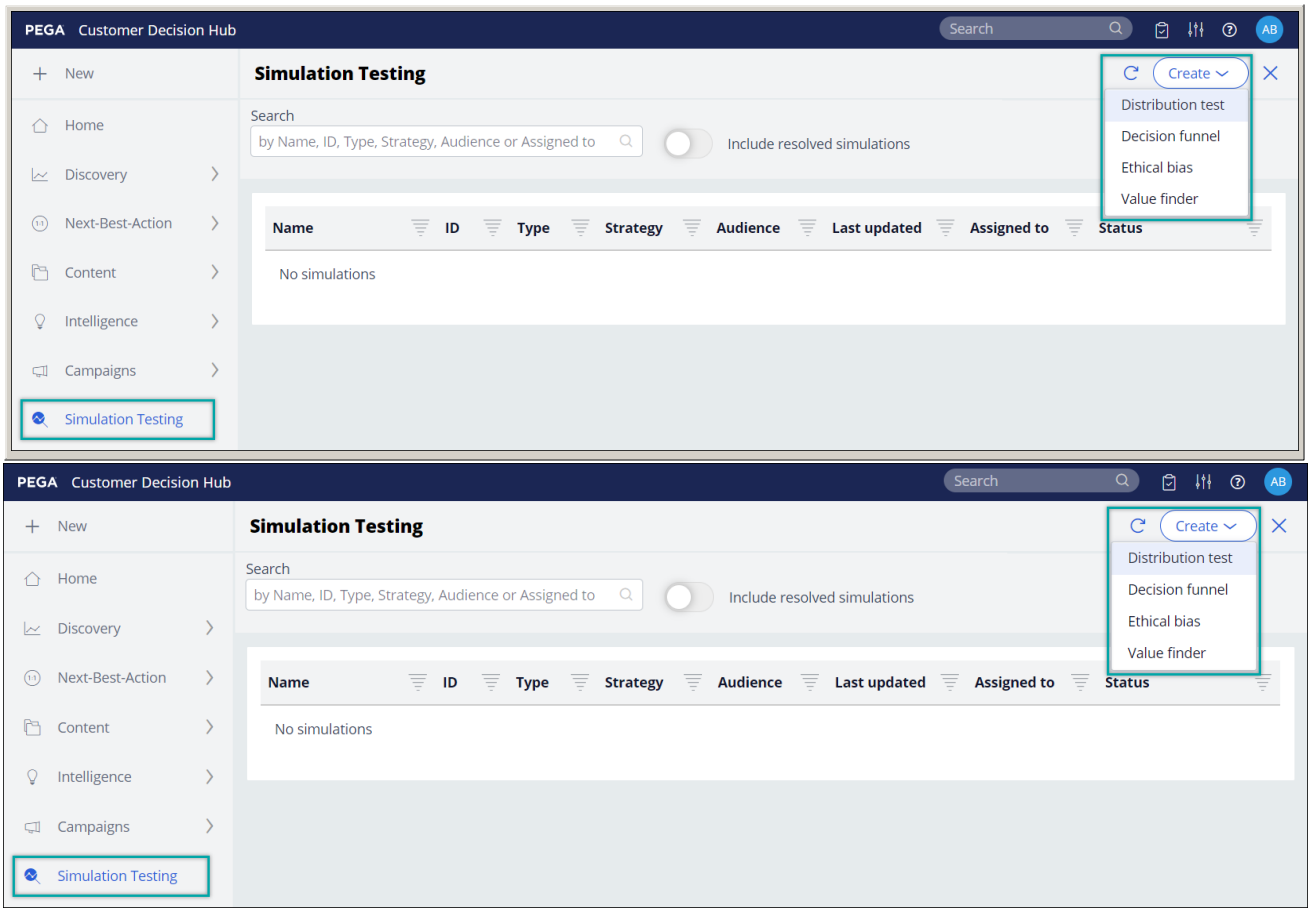
Creating simulation tests in Pega Customer Decision Hub

Create a simulation test to understand the effect of business changes on your decision management framework. For example, you can create a simulation test to investigate whether the introduction of a new business logic affects the frequency with which propositions would be offered across a segment of customers. When you complete a simulation test, you can view its output in Scenario Planner or in the simulation testing UI. You can also save the results to a data set for further processing.

Pega Customer Decision Hub

1. In the Pega Customer Decision Hub portal, click **Simulation Testing**.
2. In the top right corner of the **Simulation Testing** screen, click **Create**, and then specify the simulation test type:
   - To assess the actions made and the customers targeted when a campaign is executed, select **Distribution test**.
   - To assess how certain components and expressions influence the outcome of a decision framework, select **Decision funnel**.
   - To screen the decision strategy for bias related to factors like age, ethnicity, or gender, select **Ethical bias**.
   - To identify customers who are left without actions or with no relevant actions after any stage of the decision funnel, select **Value finder**.

     Simulation test types on the Simulation Testing landing page



3. In the **Application context** section, select the application context for the new simulation test. **For example:** You can select the current application or a specific revision as the simulation context, if your application is enabled for revision management.
4. Configure the inputs for the simulation:

Choices	Actions
Run a test on a strategy rule	a. In the **Strategy** section, click **Configure**. You can simulate only one strategy at a time. When you select a strategy, you can view the application context of the selected strategy and its Strategy Result (SR) class. b. Click **Add** next to the strategy that you want to simulate, and then click **Apply**. **Note:** To run a Value Finder simulation test on a strategy that does not use the Next-Best-Action Designer framework, ensure that the strategy has the appropriate structure. For more information, see the requirements for strategies. c. In the **Audience** section, click **Configure**. d. Click **Add** next to a data source to select it as the audience for the simulation test, and then click **Apply**. You can select a data set, data flow, customer segment, or report definition rules as the audience. For example, you can use a Monte Carlo data set to create a sample set of customer data for simulation purposes. If there is a specific data flow that you want to use as an audience, and you cannot see it for selection, ensure that the data flow meets the following requirements: ▪ The source is not abstract. ▪ The destination is not abstract. ▪ The output class is the strategy's Apply-to class, or a child of this class.
Run a test on a data set with strategy results**Note:** The option is only available for Value finder simulation tests.	a. In the **Configure inputs** section, select **Data set with strategy results and customer data**. b. In the **Data set** section, click **Configure**. c. Click **Add** next to the data set that you want to simulate, and then click **Apply**. **Note:** Ensure that the data set includes the required fields and has the appropriate structure. For more information, see the requirements for data sets.

5. **Optional:** Edit the default simulation name by clicking the **Edit** icon in the **Simulation ID prefix** section.
6. **Optional:** In the **Assign output destinations** section, define the storage point of simulation results by doing one of the following actions:
   - To configure an existing rule instance as the simulation output, click **Add Existing** and select an output from the list.
   - To create an output target for the simulation test, click **Create New** and enter parameters in the **Name** and **Type** fields of the new output target.

   Note:
   - Simulations of the type **Decision funnel** use a predefined *ExplainDetails* database table as the output destination. You must define an additional output destination for this simulation type if you want to assign additional reports in the **Reports** section.
   - If you selected an output of the Visual Business Director type for the simulation, a corresponding Visual Business Director report is automatically added in the **Reports** section.
   - You can add multiple outputs to a simulation test. The available output target types are **Database Table** and **Visual Business Director**.
7. **Optional:** Add reports to the simulation output:
   a. In the **Assign reports to outputs** section, click **Add report**.
   In this section, you can view all the outputs that you configured for this simulation.
   b. In the **Output** column, assign a report to an output type.
   c. In the **Report category** column, select the report category, for example, **Simulations**.
   d. In the **Report** column, select the report to assign to the simulation output.
   For example, if you selected **Simulations** as the report category, you can select **Channel distribution** as the report to simulate how a new proposition is being distributed across a specific channel.
   e. Click **Save**.

   Note: Simulations of the type **Decision funnel** use a predefined set of reports out of the box. To define additional reports for this simulation type, define an additional output destination as described in step 6.
8. To save the simulation test and run it immediately, click **Submit and run**.
   Tip: You can also click **Submit** to save the simulation for later use. You can later find the saved simulation on the **Simulation Testing** landing page in the Pega Customer

Decision Hub portal. For more information, see Running simulation tests.
9. After the simulation has finished running, review the results by clicking **Review reports**.
For more information, see Viewing and managing reports assigned to a simulation and Viewing additional simulation details.
10. **Optional:** If required, you can edit and re-run the simulation after it has finished running, by clicking **Actions Edit simulation** .
Editing the simulation settings removes data about the previous simulation run, and changes the simulation status back to In Build.
11. To mark the simulation as completed, click **Wrap up**.
After you wrap up a simulation, it is considered resolved and you can no longer rerun it. However, you can still review any reports created for the simulation.**Tip:** On Pega Cloud, you can configure an automatic archiving job for simulation-related case data. For more information, see Creating an archival policy.

- Value Finder requirements for strategies created outside of Next-Best-Action Designer
- Value Finder requirements for data sets

- Running simulation tests
- Running simulation tests from the strategy form
- Managing simulation tests in Pega Customer Decision Hub

Value Finder requirements for strategies created outside of Next-Best-Action Designer

You can analyze a custom strategy that does not use the Next-Best-Action Designer strategy framework by running a Value Finder simulation test. Value Finder can analyze the custom strategy as long as it has the appropriate structure of eligibility rules and arbitration that contains propensities from adaptive models.

Pega Customer Decision Hub

Required fields

Ensure that the strategy results include the following fields and values:

- *pyStage* – This column must contain at least two values: Eligibility and Arbitration. You can leave the column empty when there are no actions for a customer after eligibility.
- *pyPropensity* – This column must contain the propensities as obtained from the adaptive model.

- Creating simulation tests in Pega Customer Decision Hub

Value Finder requirements for data sets

If your organization is currently still on a Pega Customer Decision Hub version earlier than 8.6 that does not include Value Finder, you can import a data set with strategy results and customer attributes to an 8.6 environment and analyze the data there by running a Value Finder simulation test.

Pega Customer Decision Hub

Ensure that your data set meets the import requirements so that Value Finder can analyze the data. For example, the data set must contain all strategy results and appropriate customer attributes that Value Finder can use to describe under-served customers. The data set must also contain the data for at least the eligibility and arbitration stages, as well as the propensities from adaptive models.

Since the structure of a strategy framework can vary and propensity values are not always available at the eligibility stage, you might have to create a new strategy to obtain the necessary eligibility and arbitration stage results from your specific strategy framework, and to attach the necessary propensities.

Best practices

When creating a data set for Value Finder analysis, follow these guidelines:

- Run a simulation and save the strategy results for all customers. Then, join these strategy results with customer data. Use a data flow to merge these sources and to ensure that there is at least one entry for every customer in the original audience.
- Do not mix inbound and outbound results as propensity ranges can vary.
- If applicable, include multiple records for the same customer ID. There may be multiple records for the same customer ID because:
  - The Next-Best-Action Designer strategy consists of different stages: eligibility, applicability, suitability, and arbitration. The data set must include a *pyStage* column that contains at least the Eligibility and Arbitration values. The other stages are optional. If a stage is empty, this indicates that all actions were filtered out because there are no strategy results for that customer after eligibility.
  - A customer can have one or more strategy results (actions or treatments) at each of these stages.
- Include a *pyPropensity* column that indicates the propensity for each of the strategy results obtained from the adaptive model.

Required fields

Include the following fields in the data set:

- *CustomerID*
- *pyStage* – In this column, include at least two values: Eligibility and Arbitration. You can also leave it empty when there are no actions for a customer after eligibility.
- *pyPropensity* – In this column, include the propensities as obtained from the adaptive model.
- Customer attributes:
  - Include any fields that are useful in describing groups of under-served customers, for example, age, gender, or current product holdings.
  - You can exclude customer attributes such as names, addresses, or other uniquely identifiable data. A Value Finder analysis does not require this data.

Optional fields

If you use the following fields in the prioritization, you can include them in the data set. Value Finder does not use these fields for analysis, but they can be useful for further insights:

- *pyChannel*
- *pyDirection*
- *pyGroup*
- *pyName*
- *pyTreatment*
- *pyValue*

Tabular data example

The following table shows a sample data set which Value Finder can analyze to identify gaps in customer engagement. A data set must include all customers, so that Value Finder knows how large the original population is and can identify how many customers are left without actions after eligibility.

If the decision strategy returns a result for a customer at every stage of the decision strategy funnel, include all the results in the table. If a customer does not pass the eligibility stage, you can leave the result for the stage empty, so that Value Finder knows how many customers are left without actions after eligibility.

There are multiple records for some customer IDs. For example, there are three records for customer 1, which represent three strategy results. This customer is eligible for two actions (OfferX and OfferY), one of which was selected for the customer at the arbitration stage (OfferX).

CustomerID	Age	Gender	hasPremiumProductX	pyStage	pyDirection	pyChannel	pyIssue	pyGroup	pyName	pyPropensity
1	30	F	true	Eligibility	Outbound	Email	Sales	MobileData	OfferX	0.25
1	30	F	true	Eligibility	Outbound	Email	Sales	ChannelPackages	OfferY	0.31
1	30	F	true	Arbitration	Outbound	Email	Sales	..	OfferX	0.25

CustomerID	Age	Gender	hasPremiumProductX	pyStage	pyDirection	pyChannel	pyIssue	pyGroup	pyName	pyPropensity
2	27	M	false	Eligibility	Outbound	Email	Sales	..	OfferZ	0.05
2	27	M	false	Arbitration	Outbound	Email	Sales	..	OfferZ	0.05
3	42	F	true	Eligibility	Outbound	Email	Sales	ChannelPackages	OfferY	0.10
3	42	F	true	Arbitration	Outbound	Email	Sales	ChannelPackages	OfferY	0.10
4	55	M	true					NULL	NULL	NULL

- Creating simulation tests in Pega Customer Decision Hub

Running simulation tests

From Pega Customer Decision Hub, you can start a simulation test that you already configured. Additionally, you can start a new simulation test or restart an already completed one.

Pega Customer Decision Hub

1. In the Pega Customer Decision Hub portal, click **Simulation Testing**.
2. Locate the ID of the simulation test that you want to start and click it.
   Use the filter function to browse through simulations. For more information, see Filtering simulations.
3. Start the simulation by performing one of the following steps:
   - To start a new simulation, click **Start**.
   - To start an already completed simulation, click **Actions Restart simulation** .

   Caution: Restarting a simulation always clears the previous results for the simulation test.
4. After the simulation has finished running, review the results by clicking **Review reports**.
   For more information, see Viewing and managing reports assigned to a simulation and Viewing additional simulation details.
5. **Optional:** If required, you can edit and re-run the simulation after it has finished running, by clicking **Actions Edit simulation** .
   Editing the simulation settings removes data about the previous simulation run, and changes the simulation status back to In Build.
6. To mark the simulation as completed, click **Wrap up**.
   After you wrap up a simulation, it is considered resolved and you can no longer rerun it. However, you can still review any reports created for the simulation.**Tip:** On Pega Cloud, you can configure an automatic archiving job for simulation-related case data. For more information, see Creating an archival policy.

- Simulation testing
- Managing simulation tests in Pega Customer Decision Hub

Running simulation tests from the strategy form

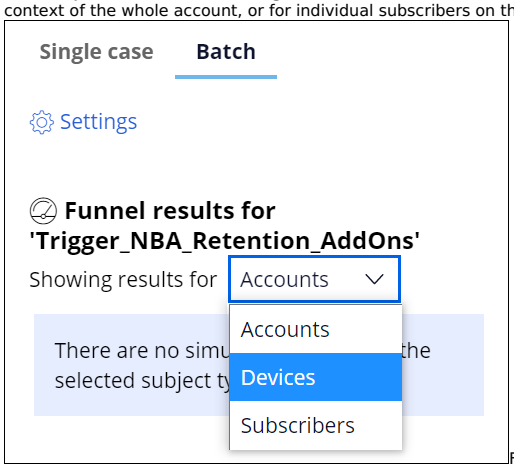
You can simulate a strategy directly from the rule form. This option simplifies strategy design process because it enables on the fly tests to investigate whether the strategy configuration contains any flaws or whether it produces the results that you expect.

Pega Customer Decision Hub

1. Access the strategy that you want to simulate. **For example:** You can access a Strategy rule from Dev Studio, Pega Customer Decision Hub, or from a change request in revision management.
2. On the right side of the Strategy form, click **Toggle side panel** to open the **Test run** panel.
3. In the **Test run** panel, click the **Batch** tab.
4. Click **Settings**.
5. In the **Analyze** section, select an option, and then configure the necessary settings:

Choices	Configuration
Decisions	Select the simulation test that you want to run from the following options: ◦ Select **An existing simulation test**, and then select the simulation ID. **Tip:** To check the configuration of the simulation test, click **Open**. ◦ Select **A new simulation test** and specify the data flow, data set, or report definition that you want to use as the input data of the simulation test.
Performance	a. Select the source and the subject of the test that you want to run from the following options: ▪ Select **Data set** and specify a data set instance that you want to use as the source of the test run. ▪ Select **Data flow** and specify a data flow instance that you want to use as the source of the test run. b. Specify the number of records on which you want to run the test by selecting one of the following options: ▪ To test the strategy on all available records in the specified data source, select **All records**. ▪ To test the strategy on a specific number of records, select **A limited number of records** and specify the number of records to use in the test run. The system uses that number of records from the top of the data set or data flow that you selected as the source.

6. Click **Run**.
7. Analyze the results.
   Decision simulation tests do not support all components. This includes adaptive and predictive models, scorecards, decision trees, decision tables, and others. If a decision simulation test does not include a component, the canvas does not display results for that component.
8. **Optional:** For decision simulations of multilevel strategies (that is, strategies that decision on various levels of customer contacts, such as account owner or device owned), you can use the **Showing results for** list to view the results of the strategy for a particular context. **For example:** You can see the results of the strategy in the context of the whole account, or for individual subscribers on the account.



   For more information about multilevel decisioning, see Understanding multilevel decisioning and the Context Dictionary in the *Pega Customer Decision Hub User Guide*.
9. **Optional:** To see if a component is optimized, at the top of the strategy canvas, click **Show optimization**. **Result:** The following symbols appear on strategy components:
   - A check mark for optimization
   - A warning sign for a lack of optimization
10. **Optional:** To see the results for a specific non-optimized strategy component, click that component.
    The **Test run** panel displays different performance results for components depending on optimization. Non-optimized components display values for all performance statistics. For example, if you click a non-optimized component, the results display such information as processing speed and throughput. Optimized components do not display any performance statistics in the results.

- Simulation testing
- Managing simulation tests in Pega Customer Decision Hub

Managing simulation tests in Pega Customer Decision Hub

You can perform various maintenance tasks on the simulation tests that you created in Pega Customer Decision Hub to quickly and effectively manage simulations in your application.

Pega Customer Decision Hub

- Filtering simulation tests
- Viewing additional simulation test details
- Configuring reports assigned to simulation test outputs
- Comparing simulation tests
- Assigning custom reports to simulation tests
- Scheduling simulation tests

Filtering simulation tests

You can filter through simulation tests in Pega Customer Decision Hub to quickly find the simulation tests that you need, for example, to create a duplicate of an existing simulation test. You can filter simulation tests by input, application revision, or operator.

Pega Customer Decision Hub

1. In the Pega Customer Decision Hub portal, click **Simulation Testing**.
2. Above the simulations list, use the **Search** field to look for simulation tests by name, ID, type, strategy, or owner.
   Note: By default, simulations that were marked as resolved are not included in the output. Use the **Include resolved simulations** toggle to view them.

Viewing additional simulation test details

You can view additional details for each simulation test in the Execute stage. Additional details can help you correctly assess performance of target strategy on the component level as well as view record distribution across all decision data nodes in your application.

Pega Customer Decision Hub
Before you begin: Additional simulation test details are not visible for operators in the default Pega 1:1 Operations Manager access groups. If required, log out and log in as an operator in a different access group.

1. In the Pega Customer Decision Hub portal, click **Simulation Testing**.
2. Click the ID of the simulation test whose details you want to inspect.
3. Under the progress bar, click **More details**.
   You can view the following details:

   Component statistics
   : View the number of successful and failed records per data flow component and the average processing time (in milliseconds). You can also view the percentage of the total processing time that your application took to process each component.

   Distribution details
   : View the number of Data Flow nodes that were assigned to process the data. You can also view the number of partitions that were created to process the data in each decision data node. The statistics display the number of records processed by each node, the number of failed records, and the current status of the decision data node.

   Run details
   : The default parameters of the simulation test run. You cannot change these parameters. In Pega Customer Decision Hub, simulations are always run in Batch mode. If a simulation test encounters at least one error on one of the nodes that are assigned to process the data, the simulation test fails on that node. The data is processed on the remaining nodes, starting from the last successfully created data snapshot.

Configuring reports assigned to simulation test outputs

You can review simulation tests by inspecting a variety of reports that can be assigned to a simulation output. Additionally, you can quickly respond to changes in business requirements or obtain additional data by adding a new report to the simulation.

Pega Customer Decision Hub
Note: You can configure reports only for simulations whose status is not Resolved, that is, simulations that you have not yet marked as completed.

1. In the Pega Customer Decision Hub portal, click **Simulation Testing**.
2. Click the ID of the simulation test whose reports you want to access.
3. In the **Assigned reports** section, view a report assigned to an output by clicking the report name in the **Report** column.
4. **Optional:** To view additional data about the simulation, in the **Assigned reports** section, click **Add report**.
5. Click **Submit**.

- Simulation testing
- Managing simulation tests in Pega Customer Decision Hub
- Assigning custom reports to simulation tests

Comparing simulation tests

You can compare outputs of two simulation tests in Scenario Planner. For example, by comparing different strategies, you can determine the strategy that best fulfills your business requirements. By simulating different strategies you can also assess how modifications in your product offering can affect product sales.

Pega Customer Decision Hub
For more information about using Scenario Planner, see Seeing the effect of your business changes with Scenario Planner.

Assigning custom reports to simulation tests

You can create custom reports and assign them to simulation tests. By using this feature, you can adjust simulation reports to your business needs, for example, by configuring a report to show additional or more detailed data.

Important: This is an advanced task. Perform this task if your operator profile has access to Dev Studio, for example, you are a system architect.
Pega Customer Decision Hub

1. Create an output that is the source for your report or use an existing output by performing the following actions:
   - To create an output, go to step 2.
   - To assign a report to an existing output, skip to step 3.
2. Create a new output for your simulation test by performing the following steps:
   a. In the Pega Customer Decision Hub portal, click **Simulation Testing**.
   b. Open a simulation test and assign it a new output by clicking **Actions Edit simulation** .
      You can add outputs to simulation tests that are in the In Build status. Editing the simulation settings removes data about the previous simulation run, and changes the simulation status back to In Build.
   c. In the **Assign output destinations** section, click **Create New**.
   d. In the **Create new output** window, enter the name of your output, for example, MyDataTable, and select the output type, for example, **Database Table**.
   e. Click **Done**.
   A new output is created in your application together with a new class *Data- Output_Name*, for example, *Data-MyDataTable*.
3. In Dev Studio, create a report definition.
   You must create a report definition in the same class as the output that you want to assign to this rule. For example, if you want to assign a report to *MyDataTable* that is in

class *Data-MyDataTable*, you must create a report definition in the same class (*Data-MyDataTable*). For more information, see Creating advanced reports.
4. Enable the report definition as a simulation report by performing the following actions:
   a. On the Report Definition form, click the **Report Viewer** tab to open it.
   b. In the **Report header** section, complete the **Report title** field.
   c. Create the report shortcut.
   For more information, see Creating a report shortcut.
5. Save the rule.
The report is now available in Pega Customer Decision Hub to assign to an output that you created in step 2.

- Simulation testing
- Running simulation tests
- Managing simulation tests in Pega Customer Decision Hub
- Viewing additional simulation test details
- Configuring reports assigned to simulation test outputs

Scheduling simulation tests

You can schedule a one-off simulation to run at a specific time. This option is useful when you expect a third-party system to populate customer data at a certain time or you want to simulate on large amounts of customer data during off-peak hours to minimize memory consumption in your application.

Pega Customer Decision Hub

1. In the Pega Customer Decision Hub portal, click **Simulation Testing**.
2. Open the simulation that you want to schedule and click **Actions Schedule simulation** .
The simulation must be in the Execute or Review stage.
3. In the **Schedule run** window, click the **Calendar** icon and select the year, month, day, and hour for the simulation to start.
4. Click **Apply**.
The simulation status changes to Open-Scheduled. You can cancel any scheduled run by clicking **Cancel schedule**.

- Simulation testing
- Managing simulation tests in Pega Customer Decision Hub

Testing Next-Best-Action configuration with audience simulations

You can improve the performance of your Next-Best-Action Strategy by testing the configuration of engagement policies against simulated audiences. In this way, you can check how many potential actions are filtered out by each component of the policy, and discover if a particular criterion is too broad or too narrow for your requirements.

Pega Customer Decision Hub

1. On the **Engagement policy** tab, click the name of any group, or click **All groups** to run the simulation for all issues and groups in your business structure.
2. Click **Actions Audience simulation** .
3. In the **Audience simulation** section, select or create a simulation with which you want to test the proposition filter:
   - To use an existing simulation, select the simulation in the **Simulation** list.
   - To create a new simulation, click **Create simulation**, and then select the audience to use for the simulation.
   Important: If you are creating your simulation for Scenario Planner purposes, you must select **Engagement policies and arbitration** as the **Next-best-Action scope**. Only the simulations run with arbitration will work with Scenario Planner.
4. After the simulation test finishes, analyze the results to see what percentage of the audience would receive each action according to the current engagement policies.
Result: For each customer context in your application, the simulation drills down to individual components of the engagement policy. The results show numerical or percentage values that indicate the population of the selected audience that receives the proposition based on the current criteria. For example, if the result for a criterion that checks if the proposition is active returns 100.00%, the condition did not filter out any audience members.

- Seeing the effect of your business changes with Scenario Planner

Seeing the effect of your business changes with Scenario Planner

Understand the effect of changes to your Next-Best-Action Designer configuration by using the planning capability of Pega Customer Decision Hub. Use scenario planning to examine the impact of adding new actions, changing lever values, or modifying next-best-action engagement policies.

Browse the results of next-best-action audience simulations and monitor progress towards business priorities in a simple, transparent interface. View the outcomes and compare two simulations to understand the trade-offs across all issues and groups, or drill down to the details of a single action and channel.

Pega Customer Decision Hub

With Scenario Planner you can easily simulate “what-if” scenarios, so that you can forecast results and align your next-best-action strategies to hit specific goals.

For example, you can use Scenario Planner to view how your actions are performing, and whether you can expect to achieve your KPIs. You can also see the opportunity cost of adjusting your strategies, that is the benefit that would have been derived by not choosing an option.

By using Scenario Planner, included as a Limited Availability release in Pega Platform version 8.6, you can:

- Simulate changes across actions, treatments, issues, groups, and channels.
- Project reach, accepts, and value capture.
- Adjust levers to better align with business objectives.
- Compare two scenarios and see the potential trade-offs.
- Use a representative sample of your customer population for a simulation.

Tip:

Watch the below video to get to know Scenario Planner:

If you want to learn more about Scenario Planner simulations, visit Pega Academy.

- Viewing outcomes in Scenario Planner
- Scenario planning results

Viewing outcomes in Scenario Planner

Use scenario planning to understand the impact of changes made in Next-Best-Action Designer. Browse the results of next-best-action audience simulations and monitor progress towards business objectives in a simple, transparent interface. View the outcomes and compare two simulations to understand the trade-offs across all issues and groups, or drill down to the details of a single action and channel.

Important: Because Pega Customer Decision Hub follows best practice guidelines for business change management, you should only run simulation testing in a dedicated Simulations or Business Operations Environment. For more information, see Deployment architecture for Pega Customer Decision Hub. As such, the Scenario Planner menu item is inactive on any environment that is designated a Level 5 Production environment.
Pega Customer Decision Hub

1. In Next-Best-Action Designer, create and run an audience simulation that corresponds to a new next-best-action configuration that you want to test.
For more information, see Testing Next-Best-Action configuration with audience simulations.
Important: Make sure to configure the following simulation settings:
   - In the **Next-Best-Action scope** section, click the **Engagement policies and arbitration** radio button.
   - Select the **Include a second-pass simulation without business levers (for Scenario Planner)** check box.
2. Run the simulation that you created. **Result:** Two simulations will run in parallel: one based on your Next-Best-Action Designer configuration with levers, and one based on your Next-Best-Action Designer configuration without any levers. The two simulations are run in order to display the opportunity cost that is associated when applying the levers. For more information, see Comparison simulation.

3. In the Pega Customer Decision Hub, click **Discovery Scenario Planner** .
4. Use the **Scope** list to define the range of the context in which you view the data.
5. In the **Simulation**, select the simulation that you created before.
6. **Optional:** Measure the difference between two simulations by selecting **Comparison simulation**. For more information, see Comparison simulation.
7. Click **Apply**.

What to do next:

- Review the simulation results. For more information, see Scenario planning results.
- Click **Export Excel file**, to save the simulation results as an .xlsx file.

Tip:

If you would like to have more hands-on experience with Scenario Planner simulations, click here to try it out in Pega Academy environment: Analyzing simulation results using Scenario Planner.

Scenario planning results

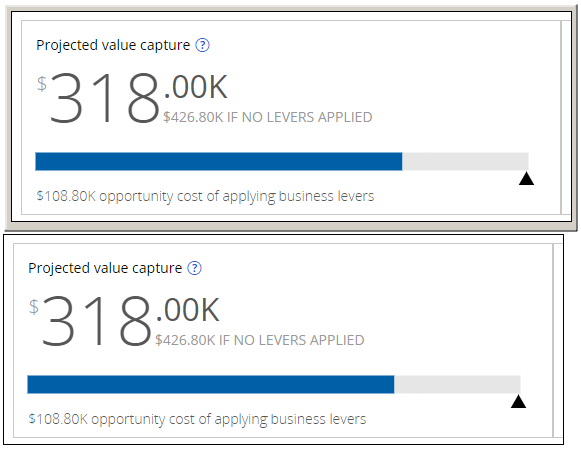
The Scenario Planner landing page shows a number of useful key performance indicators (KPIs), including audience simulation counts and projected reach, projected accepts based on adaptive models, projected value capture, as well as historical accepts, sends, and impressions.

Depending on the selected filters, the Scenario Planner view can display the results for a specific channel. You can easily navigate down the hierarchy by clicking on the specific issue, group or action in the details table. The breadcrumbs functionality allows you to conveniently move backwards within the business structure.

Pega Customer Decision Hub

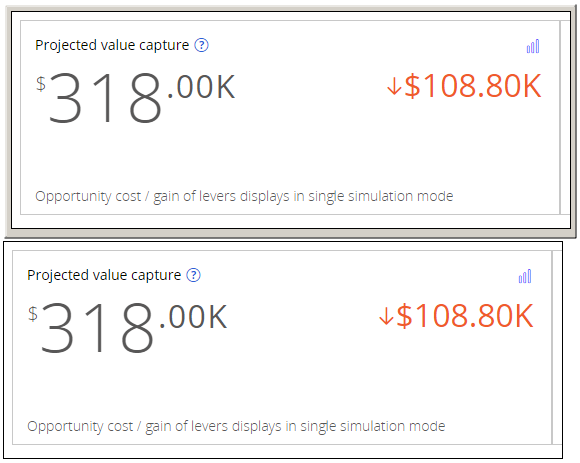
Projected value capture

The **Projected value capture** tile from the Scenario Planner results screen shows the estimated value that the upcoming next best actions (outbound runs and inbound impressions) will bring with the currently selected configuration.



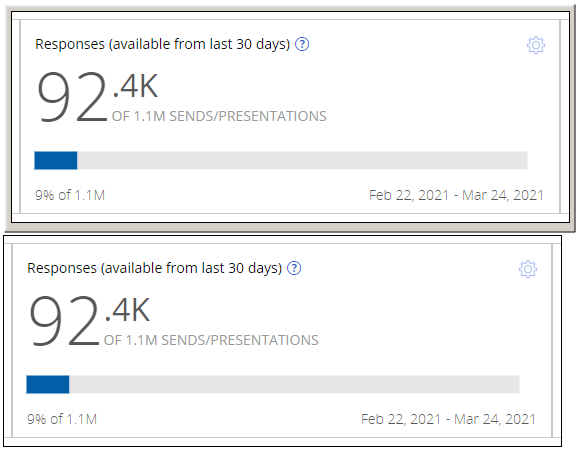
When you run a single simulation, the **Projected value capture** tile displays a bar. The blue part of the bar represents the projected value captured from all of the engagement policies and levers that you configure in Next-Best-Action Designer.

The black triangle and the gray part of the bar represent the opportunity cost of the levers that you apply, that is, the benefit that would have been derived by not choosing an option. If you do not see a gray part, this means that with the applied levers you are gaining the opportunity. The bar is not displayed when comparing two scenarios, as in the following figure:



Responses

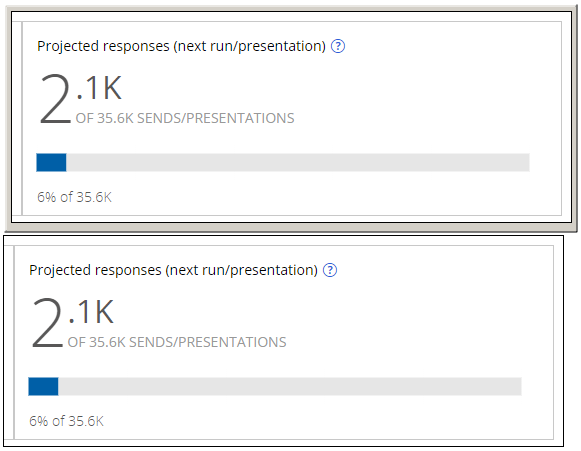
The **Responses** tile shows the number of accepted actions versus the aggregate number of outbound sends and inbound impressions over a certain period of time, as in the following figure:



By clicking the **Open** icon in the upper right corner of the **Responses** tile, you can change the time frame for the historical data from the last 30 days to the last 60, 90, 180, or 365 days. You can also aggregate data, from the last full quarter up to a full year.

Projected responses (next run/presentation)

The **Projected responses** tile shows the aggregate number of sends or impressions with the projected accepts that the upcoming next-best-actions (outbound run and inbound impressions) will bring with the currently selected configuration, as in the following figure:



Details table

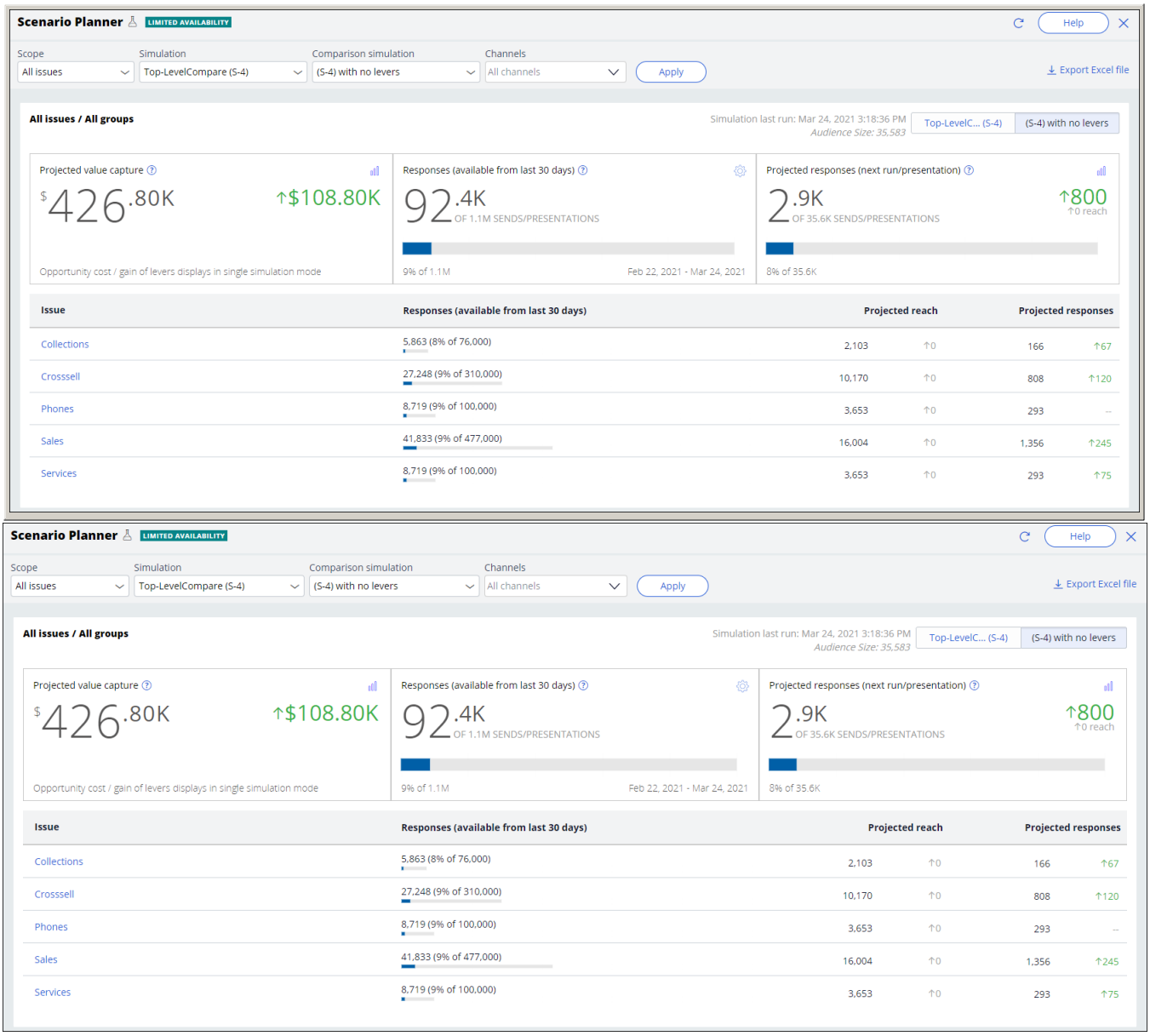
The **Details** table shows results by selected simulation, issue, group, action, and channel. Click on an issue or group in the table to easily drill down to the details of a specific action or treatment, or see a broad comparison across the whole issue or group.

Comparison simulation

To compare two simulations, you first choose one simulation from the **Simulation** list in Scenario Planner. In the **Comparison simulation** list, the first simulation in the list (assuming that you enabled the second-pass run) will then be your current simulation, but without any levers.

The tiles and table show the deltas, that is, the differences between the results of the two simulations. The deltas are indicated by arrows and color-coding.

You can click on any issue to drill down to the next level of your simulation. You can also switch between the tabs, to see the results from the perspective of one or the other simulation. The following figure shows a simulation results in Scenario Planner:**Note:** Currently, Scenario Planner does not support simulations in which more than one action or placement is requested for one customer (primary context). Scenario Planner will count only the top action for each customer in projected KPIs for simulations with requests for multiple actions.



Important: For more information on how to configure the two simulations to run in parallel: one based on your Next-Best-Action Designer configuration with levers, and one based on your Next-Best-Action Designer configuration without any levers, see Viewing outcomes in Scenario Planner. The two simulations will run in order to display the opportunity cost that is associated when applying the levers.**Note:**

Watch the video below, to enhance your knowledge about the topic:

To learn more about Scenario Planner, visit Pega Academy: Analyzing the effect of business changes using Scenario Planner.

Complying with policies or regulations by detecting unwanted bias

Test your strategies for unwanted bias. For example, you can test whether your strategies generate biased results by sending more actions to female rather than male customers.

Pega Customer Decision Hub
Before you begin:

To edit the ethical bias policies, your access group must have the *pzBiasPolicyConfiguration* privilege. For more information, see the *Pega Customer Decision Hub Implementation Guide* on Community

Use the Ethical Bias Policy landing page to configure the fields that are used to measure bias.

1. In Dev Studio, click **Configure Decisioning Infrastructure Ethical Bias Policy** .
2. On the **Bias fields** tab, click **Add bias**.
   You can select any property from your customer class. For example, you can use gender, age, and ethnicity-related properties for bias testing.
3. If the property value that you selected is a number, in the **Add bias field** window, specify whether to represent that value as a category or as an ordinal number.
   Categorical values represent customer properties such as gender or ethnicity. If there are many categorical values, only the 20 most frequent values are checked for bias. Do not classify numerical values such as age as categories.
4. On the **Bias threshold** tab, review and configure the bias threshold settings for each issue in your business structure.
   The bias threshold measurement depends on the type of field that you selected. For more information, see Bias measurement.
5. Confirm your settings by clicking **Save**.

What to do next:

To use the bias policy to test the behavior of your strategies, create a new simulation test with the purpose **Ethical bias**. For more information, see Simulation testing.

- Bias measurement

Bias measurement

Understand how to properly measure unwanted bias in offering products to your customers to comply with your company policies and regulations.

The bias threshold measurement depends on the type of field that you selected. Depending on your settings, you can select the rate ratio or Gini coefficient.
Pega Customer Decision Hub

Rate ratio
: Use this ratio to determine bias for categorical fields by comparing the number of customers who were selected for an action to those not selected for an action, and correlating that to the selected bias field. For example, the rate ratio that is represented in the following table indicates that actions are sent more often to male rather than female customers:

Example rate ratio

	Female customers	Male customers
selected for the action	500	1000
not selected for the action	20000	18000
rate ratio	[500 / (500+20,000)] / [1000 /(1000+18000] = 0.46	[1000 / (1000+18,000)] / [500 /(20,000+500] = 2.16

A rate ratio of 1 represents perfect distribution equality. You can select a warning threshold between 1 (warn if any bias is detected) and 0.50 - 2.00 (warn only if very high bias is detected). You can also choose to ignore this bias field for a particular issue in your business structure.

Gini coefficient
: Use the Gini coefficient calculate bias for numerical fields. This is a method of measuring the statistical inequality of value distribution, for example, the distribution of actions to customers based on their age. A Gini coefficient of 0 represents perfect distribution equality. You can select a warning threshold between 0 (warn if any bias is detected) and 0.7 (warn only if very high bias is detected). You can also choose to ignore this bias field for a particular issue in your business structure.

Optional: Comparing decision results in the Visual Business Director planner

Check the performance of the champion and challenger strategies across all channels, products, or lines of business in the 3-D graphical view to see how the challenger strategy compares to the champion strategy.

Pega Customer Decision Hub

1. Access VBD planner.
2. In the right side panel, click the **Data** tab.
3. Select a primary data source.
4. Select a reference data source.
5. If the primary data source is not the same as the reference data source, select a data mode:

   Regular
   : shows the primary data source versus the reference data source.

   Delta
   : provides comparison between the reference data source and the primary data source by showing the delta between the results of two strategies. The red cone means that the value of the reference data source is higher than the primary source.

6. Click **Update data** to update the selected data sources.
7. Select the **Automatically update data** check box.
8. Click the **Time** tab and specify the time range for showing the data.
   a. **Optional:** In the Show list, select a predefined time range.
9. Click the **Filter** tab and select dimension levels to define customer interactions.
   a. Select check boxes to include specific dimensions or click **Select all**.
   b. Click **Apply changes**.
   Together with KPIs, dimensions are used in VBD to construct the business view. The visualization of dimensions is determined by the dimension filter and the Set Y Axis and Set X Axis dialog boxes.
10. In the scene, move the mouse cursor to a particular bar in the chart to see its details.
11. **Optional:** To refresh the scene with data of the selected KPI, click a highlighted KPI line chart on the wall.
12. **Optional:** Configure the scene by performing the following actions:
    a. Select a KPI to display on the wall by clicking the KPI title and in the **Select KPI** dialog box, selecting a KPI to display.
    VBD planner displays up to six of the most recent KPI line charts. If more than six KPIs are defined, you can select one to display on the wall.
    b. Click a KPI chart to display it on the grid.
    c. Change the dimension that is displayed on the y-axis by clicking the axis label on the left and in the **Set Y Axis** dialog box, selecting a different dimension or a level in the dimension.
    d. Use the sliders in the Timeline console below the scene to change the time range for showing the data.
    e. Change the dimension that is displayed in the x-axis by clicking the axis label on the right and in the **Set X Axis** dialog box, selecting a different dimension or a level in the dimension.
    For example, you can perform time-based analysis for outcome by setting the y-axis to outcome, and setting the x-axis to the time period for showing customer behavior.
    f. Manipulate the scene with the buttons in the top right corner.
13. **Optional:** To reset all the settings in the right side panel, click **Refresh**.

- Accessing Visual Business Director (VBD) planner
- Customizing Visual Business Director (VBD) planner
- The Data Sources tab
- The Key Performance Indicators tab
- Visual Business Director planner

- Visual Business Director planner

Accessing Visual Business Director (VBD) planner

Open VBD planner to display decision results with a 3-D graphical view.

Pega Customer Decision Hub
Before you begin: Make sure that there is at least one KPI on the KPI list. If the list is empty, add at least one KPI.

1. Configure the Real Time Data Grid if you have not already done so.
2. In the header of Dev Studio, click **Configure Decisioning Monitoring Visual Business Director Key performance indicators** .
3. Click the **Data Sources** tab.
4. In the **Name** column, click a link to open VBD planner for a specific data source.

- Visual Business Director planner
- Optional: Comparing decision results in the Visual Business Director planner

Customizing Visual Business Director (VBD) planner

When you open the VBD planner, the *pyGetConfiguration* activity under the *Data-Decision-VBD-Configuration* class gathers the information required to render the VBD planner. This information (dimensions, properties, and KPIs) is retrieved from Interaction History and forms the basis for visualizing decision results. The *pyGetDimensions* activity under the *Data-Decision-VBD-DimensionDefinition* class provides a number of customization points.

Pega Customer Decision Hub

The description of the properties that represent the dimension determines the labels in VBD planner. If you want to change the default text (label), change the description of the corresponding property under the *Data-Decision-IH-Dimension-<DimensionName>* class.

You can also customize how dimensions are displayed in theVBD planner. The *pySetupDimension* activity under the *Data-Decision-VBD-DimensionDefinition* class can be circumstanced by dimension name. You can override the *pyLevels* value list to define a different sequence of properties for a given dimension, for example:

- Circumstance the *pySetupDimension* activity by property when the *pyName* is Application.
- Use the *Property-Set* activity to set the sequence of level in the application dimension to put the application information at the end of the sequence.
  - Set *.pyLevels(1)* to pyInteration
  - Set *.pyLevels(2)* to pyStrategy
  - Set *.pyLevels(3)* to pyComponent
  - Set *.pyLevels(4)* to pyApplication

You can set the default level to be displayed for a dimension by overriding the *pyDefaultLevel* property for a given dimension, for example:

- Circumstance the *pySetupDimension* activity by property: when the *pyName* activity is Action.
- Use the *Property-Set* activity to set the default level for the action dimension as group by setting *.pyDefaultLevel* to pyGroup.

- Visual Business Director planner

The Data Sources tab

The Data Sources tab displays data sources that represent the contents of the Interaction History (Actuals) or the records that you want to visualize in the VBD Planner. These data sources are generated by running a data flow that generates simulation data.

Pega Customer Decision Hub

You can perform the following actions to manage the selected data source:

- Delete a data set and information associated with it.
- Clear a data set to delete the records in it.
- Expand a data set, define its start date, and use the date to monitor the data set.
- Click a data source to launch VBD planner.

- Visual Business Director planner

The Key Performance Indicators tab

The Key Performance Indicators (KPIs) tab allows you to view and manage the available key performance indicators. Once defined, the KPIs are calculated every time the system writes results to Interaction History, Visual Business Director, and database tables.

Pega Customer Decision Hub

For each KPI, you can check its name, description, the time stamp, and the user name corresponding to the last change.

- Adding a key performance indicator
- Editing a key performance indicator

Adding a key performance indicator

The key performance indicators (KPIs) can be used to track, compare, and monitor business performance across the defined areas of interest. You create KPIs based on outcomes previously defined in interactions.

Pega Customer Decision Hub
Before you begin: Before you can monitor any data, specify VBD server host and port within the Infrastructure Services landing page.

1. In the header of Dev Studio, click **Configure Decisioning Monitoring Visual Business Director Key performance indicators** .
2. Click **New** and provide a name for the new KPI.
3. In the **Selected outcomes** list, select type of operation that defines the KPI formula.
4. In the **Available outcomes** section, check the list of possible values in the outcome dimension.
5. Click on the available outcomes or click **Add All** to select all the available outcomes.
   The **Selected outcomes** section lists the possible values selected from the **Available outcomes** section.
6. In the **Description** section, select one of the following options:
   - Use generated
     Automatically generates KPI description.
   - Use custom
     Requires entering your own KPI description.
7. In the **Display data in Visual Business Director** section, select one of the following options:
   - Cumulative
     VBD displays values accumulated over time.
   - Non-cumulative
     VBD does not display values accumulated over time.
8. In the **Compare data sources in Visual Business Director** section, select one of the following options:
   - Higher values are better
     The values below a reference value are displayed in red and green when above.
   - Higher values are worse
     The values below a reference value are displayed in green and red when above.
9. Click **Submit**.

Result: The new key performance indicator is added to the list in the **Key performance indicators** tab.

- Editing a key performance indicator
- The Key Performance Indicators tab

Editing a key performance indicator

The key performance indicators (KPIs) can be used to track, compare, and monitor business performance across the defined areas of interest.

Pega Customer Decision Hub

1. In the **Key performance indicators** tab of the **Visual Business Director** landing page, click **New**.
2. Click the link in the **Name** column to change an existing KPI.
3. In the **Edit KPI** dialog, make your changes.
   For details about particular settings, see Adding a key performance indicator.
4. Click **Submit**.

- The Key Performance Indicators tab

Visual Business Director planner

The Visual Business Director (VBD) planner is an HTML5 web-based application that helps you assess the success of your business strategy after you modify it. Use the planner to check how a new challenger strategy compares to the existing champion strategy.

Pega Customer Decision Hub

To compare two strategies, you run a simulation before and after you modify a strategy. The strategy that you currently use is called a champion strategy. When you run a simulation on it, the strategy results that you get can be used as a reference data source in the VBD planner. Then you create a new strategy or modify the challenger strategy and run a simulation on it. This is a challenger strategy and its results can be used as a primary data source in the VBD planner. When you have the two data sources, you can open them for visual comparison in the three-dimensional (3-D) graphical view of the VBD planner.

The 3-D graphical view displays different dimensions and key performance indicators (KPIs), such as accept rate, conversion rate, average price, volume, number accepted, or number of processed responses. This information is retrieved from Interaction History and forms the basis for visualizing decision results and monitoring KPIs with the graphical view.

The VBD planner includes these elements:

- Scene - Displays a 3-D graphical view of different dimensions and key performance indicators (KPIs).
- Timeline - A collapsible panel where you can browse the recorded historical performance and predict the future performance of the business strategy.
- Settings panel - A side panel where you configure VBD planner settings to visualize decision results.

Note: If you previously used the applet mode of the VBD planner, you can enable it in the Visual Business Director nodes tab.

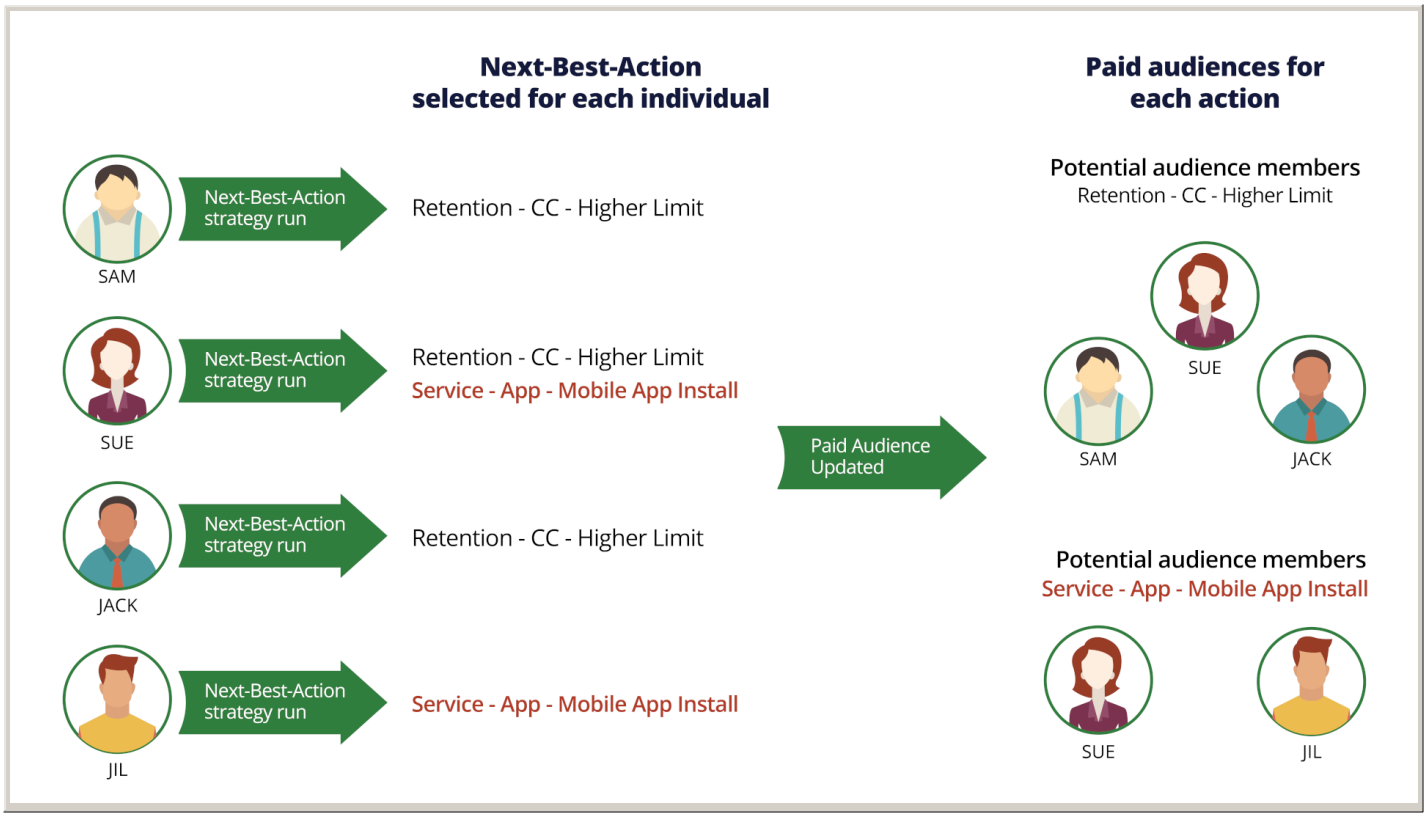
Extending next-best-action to digital advertising platforms with Paid Media Manager

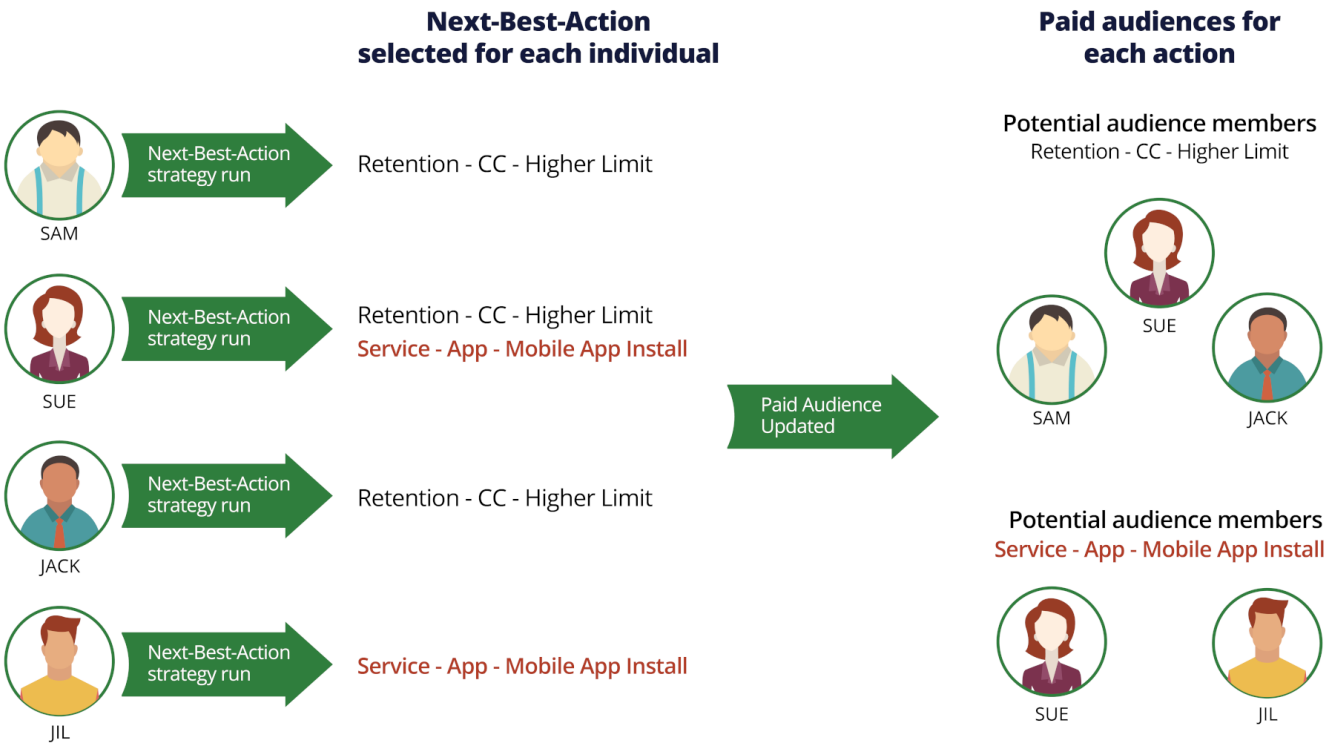
Paid Media Manager extends Next-Best-Action to modern digital advertising platforms, such as Facebook Ads and Google Ads. Each individual's Next-Best-Action is communicated through audiences on ad platforms.

Pega Customer Decision Hub

Traditional advertising campaign management tools rely on audience-based mechanisms and anonymized data segments which can be shared outside the enterprise firewall to determine what ad each individual should see. With Paid Media Manager, action, eligibility, and priority decisions are made with the full fidelity of enterprise data and intelligence within the enterprise firewall, in the same way as with other Pega Customer Decision Hub-enabled channels.

Paid Media Manager uses the latest Next-Best-Action strategy execution to place individuals into audiences and leverages these audiences on ad platforms to target the individual with the right action at the right priority. Each paid run results in the audiences being updated according to the latest next-best-action for each individual, as in the following figure:





In this way each known individual, including customers and prospects, can be targeted with ads based on individual-level decisioning rather than audience-based rules. These decisions and the model-based learning which powers them can run on the full fidelity of enterprise data within the firewall. Only the action-based decision is communicated to the advertising platforms in the form of individual audience updates.

Example scenario

For example, Next-Best-Action may determine that based on an interaction in the call center, an individual named Troy Murphy has a high propensity to accept a UPlusBank Travel Credit Card offer from his bank, the UPlus Bank. Paid Media Manager adds Troy to an audience, for example, UPlusBank Travel Credit Card – High Priority, on each ad platform. Within each ad platform these audiences are configured to be targeted with the appropriate advertising creative for the action. The maximum bid amount for the add click or impression can also be aligned with the priority of that action for each individual.

The result is that soon after the interaction in the call center, Troy's Next-Best-Action is updated across major ad platforms. He is targeted with Travel Credit Card advertisements across major media properties that he interacts with regularly, such as Facebook, Instagram, YouTube, and Google Search.

- Learning about paid destinations
- Enabling segment updates on paid destinations
- Communicating conversions to destination platforms to optimize return on ad spend
- Using action eligibility to reduce wasted ad spend on people-based destinations
- Responding to individual engagement on paid channels with real-time events
- Reducing wasted ad spend based on action priority
- Reducing wasted ad spend based on destination-specific interaction results
- Reduce wasted ad spend based on the predicted value of a click
- Prospecting with Next-Best-Action paid audiences
- Reference

Learning about paid destinations

A paid destination holds the configuration of how an ad platform receives audiences using a specific integration mechanism. You can have multiple paid destinations for each ad platform. For example, separate Google Ads accounts may be used for Google Search and for YouTube. After a paid destination is set up, it can be configured to receive paid audiences based on Next-Best-Action Designer or intelligent segments.

Pega Customer Decision Hub

Paid Media Manager supports modern people-based destinations, such as LinkedIn Ads, Facebook Ads, or Google Ads. You can also use it with legacy integration methods supported by traditional website banner advertisements.

People-based paid destinations

Paid Media Manager uses the people-based matching APIs for Facebook Ads, Google Ads, and LinkedIn Ads to synchronize paid audiences and target individuals based on their personally identifiable information (PII). **Note:** *People-based APIs* is the general term for APIs that are intended to operate on known individuals, typically by using hashed PII such as emails and phone numbers as the common ID linking the individual in the two systems.

To set up a connection, you must configure both Paid Media Manager and the paid destination to which you want to connect. For more information, see the following topics in the *Pega Customer Decision Hub Implementation Guide*:

- Mapping individuals to paid destinations with CRM attributes
- Configuring Facebook Ads destinations
- Configuring Google Ads destinations
- Configuring LinkedIn Ads destinations

After the connection is established, a one-way hashed PII is sent directly from Pega Customer Decision Hub to the ad platforms through a secure connection. If the one-way hashed PII matches what the ad platform knows about the individual, the audience is updated. The hashed PII values are then discarded. The one-way hash cannot be reversed to expose PII values to the paid destination. In this way, the individuals' PII is not exposed and the one-way hash is only used to update the audience membership on the paid destination.

After an anonymous link is established, Paid Media Manager can add and remove individuals based on the actions that have been assigned to them by Next Best Action in near real-time. Because people-based updates do not rely on website cookies, they can be triggered by scheduled runs, real-time container calls, or action responses.

Updating PII values

For updates to individuals' PII values, such as a new name or phone number, the PaidCustomerResyncMaster agent periodically scans for changes. If a PII update is detected, for example, if there is a new phone number, the agent updates the information used for subsequent paid syncs across destinations which use that information. The individual is also removed from existing audiences which use the old PII and added back with the updated PII. By default, this is configured to run every 7 days. For more information, see Paid Media Agents.

Updating PII mappings

If you update a PII mapping, for example, by mapping the destination platform email to a different property in Pega Customer Decision Hub, each audience for the updated destination will be rebuilt with new PII values. Rebuilding audiences based on updated mappings may take a similar amount of time as an initial scheduled run. The ProcessPaidMediaReSync agent scans for changes in PII mappings. If a change is detected, it triggers a resynchronization for that destination. By default, the agent is configured to run every 5 seconds. For more information, see Paid Media Agents.

Web-based and DMP connections to paid destinations

In addition to modern, people-based platforms, Paid Media Manager also supports pixel-based APIs provided by advertising technology platforms, including demand-side platforms (DSPs), data management platforms (DMPs), and advertising networks. Instead of people-based APIs, these platforms generally rely on third-party cookies to build anonymous profiles and target individuals with ads. For more information on how Paid Media Manager integrates with these mechanisms, see the topic Configuring web-based and DMP connections to paid destinations in the *Pega Customer Decision Hub Implementation Guide*.

- Understanding paid media artifacts

Understanding paid media artifacts

Paid Media Manager generates and continuously updates a number of artifacts on ad platforms to enable targeting individuals with next-best-action decisions.

Pega Customer Decision Hub

Audiences

Audiences list the individuals that are targeted by specific actions, for example, a Platinum credit card offer or a mortgage refinance. Paid Media Manager generates the following types of audiences:

Next-best-action audiences
List of individuals to target for a particular action. If using a paid strategy, this can also indicate the priority to target an individual with a particular action.
For destination-specific information, refer to the following table:

Destination type	Audience type in the paid destination	Naming schema	Description
Facebook Ads	Custom audience	*Issue-Group-Action*, for example, Sales-Card-Platinum	Generated for every action that has been enabled for paid synchronization.
Google Ads	Customer Match with customer contact information	*Issue-Group-Action*, for example, Sales-Card-Platinum	Generated for every action that has been enabled for paid synchronization.
	Customer Match with mobile device IDs	*Issue-Group-Action-Device ID*, for example, Sales-Card-Platinum-dc336b91	Generated for every action that has been enabled for paid synchronization, if you enabled device synchronization in the destination settings.
LinkedIn Ads	Matched audiences	*Issue-Group-Action*, for example, Sales-Card-Platinum	Generated for every action that has been enabled for paid synchronization.
Web destinations	Positive audience	*Issue-Group-Action*, for example, Sales-Card-Platinum	Generated and stored in Pega Customer Decision Hub.
DMP destinations	Audience	Audience key	Added and removed to destination platforms in the following format: *<<customerid>> add=<<audienceid>> remove=<<audienceid>>*, for example, CUST-1022 add=af7b511d,add=364fd700,remove=184960e2,remove=554307fd

Next-best-action lookalike and similar audiences
Lookalikes generated by the ad platform based on the list of individuals to target for a particular action and priority.
For destination-specific information, refer to the following table:

Destination type	Audience type in the paid destination	Naming schema	Description
Facebook Ads	Value-based custom audience	*Issue-Group-Action-LAL*, for example, Sales-Card-Platinum-LAL	Generated for every action that has been enabled for paid synchronization, if you enabled the creation of lookalike audiences in Facebook Ads. This type of custom audience is used as the basis to create the matching lookalike audience.
	Lookalike audience	*Issue-Group-Action-LAL*-Lookalike audience ratio, for example, Sales-Card-Platinum-LAL-0.01	Generated from the matching value-based custom audience.
Google Ads	Similar audience	*Similar to <Customer Match audience name>*, for example, Similar to Sales-Card-Platinum	Automatically generated for every Customer Match audience that has been created in the paid destination.

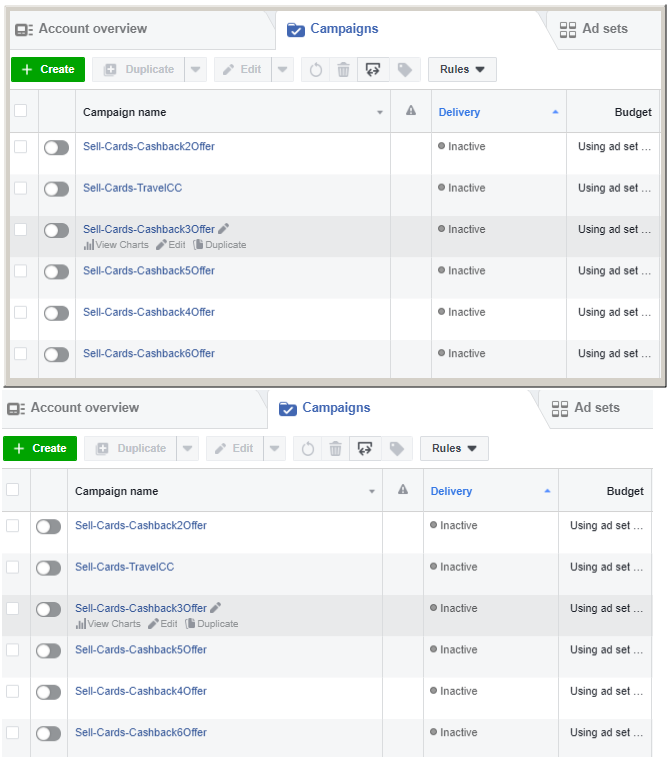
Next-best-action negative audience
List of individuals who are not eligible for an action due to next-best-action eligibility rules such as minimum age, or because they have already accepted or rejected an action. For more information, see Reducing wasted ad spend with negative audiences and next-best-action.
For destination-specific information, refer to the following table:

Destination type	Audience type in the paid destination	Naming schema	Description
Facebook Ads	Custom audience	*Issue-Group-Action-N*, for example, Sales-Card-Platinum-N	Generated for every action that has been enabled for paid synchronization.
Google Ads	Customer Match with customer contact information	*Issue-Group-Action-N*, for example, Sales-Card-Platinum-N	Generated for every action that has been enabled for paid synchronization.
	Customer Match with mobile device IDs	*Issue-Group-Action-Device ID-N*, for example, Sales-Card-Platinum-dc336b91-N	Generated for every action that has been enabled for paid synchronization, if you enabled device synchronization in the destination settings.
LinkedIn Ads	Matched audiences	*Issue-Group-Action-N*, for example, Sales-Card-Platinum-N	Generated for every action that has been enabled for paid synchronization.

Campaigns

On ad platforms, a campaign is used to specify an objective, budget, and sometimes targeting. Paid Media Manager can generate campaigns in the following ad platforms:

- Facebook Ads
- Google Ads

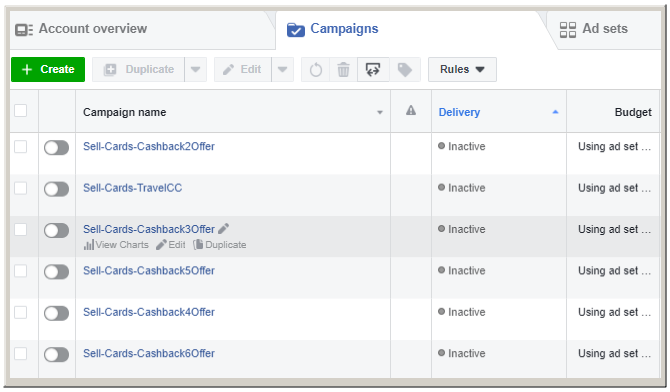


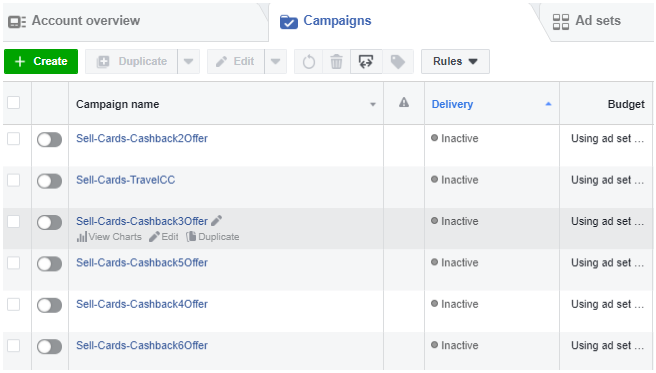
By default, campaigns are created with the following default settings that you should edit as needed before activating the campaign:

Destination	Default campaign settings
Facebook Ads	• Campaign objective - LINK_CLICK • Campaign status - PAUSED • Special ad category - NONE • Spend cap for campaign - NONE
Google Ads	• Bidding strategy - CPC (enhanced) • Campaign status - PAUSED • Daily budget - 1, with the currency set to the account currency • Campaign type for **Customer list with customer contact information** audiences - Search • Campaign type for **Customer list with mobile device IDs** audiences - Display

Ad sets and ad groups

Most ad platforms use a sub-campaign entity to specify how a group of ads within a campaign are delivered. On Google Ads this entity is called an ad group, and on Facebook Ads it is called an ad set. Paid Media Manager generates an ad set or ad group for every action that has been enabled for paid synchronization. If a Paid Strategy is configured to deliver prioritized audiences for each action, then the audiences it creates in the destination platform indicate the priority, as seen in the following figure:



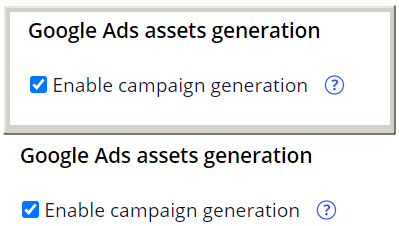


By default, ad sets and ad groups are created with the following default settings that you should edit as needed before activating the campaign:

Destination	Default ad set/group settings
Facebook Ads	• Ad set status - PAUSED • Billing event - IMPRESSIONS • Daily budget - 100, with the currency set to the account currency • Bid strategy - LOWEST_COST_WITHOUT_CAP
Google Ads	• Ad group status - PAUSED

Enabling ad set, ad group, and campaign generation in Paid Media Manager

Paid Media Manager automatically generates campaigns and ad sets or ad groups if you select the **Enable campaign generation** checkbox in the destination settings, as shown in the following figure:



Google Ads assets generation
Enable campaign generation

For more information, see the following topics in the Pega Customer Decision Hub Implementation Guide:

- Configuring Facebook Ads destinations
- Configuring Google Ads destinations

Overview of the creation process for campaigns, ad sets, and ad groups

The process of creating campaigns, ad sets and ad groups in the destination platform consists of the following stages:

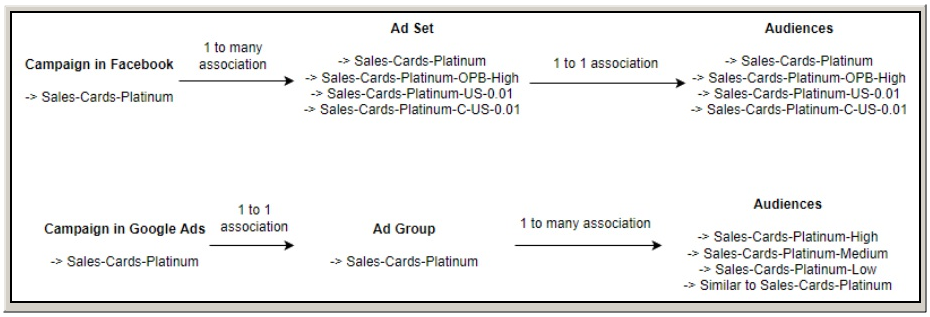
1. The *PaidMediaArtifactsMaster* agent runs at scheduled intervals to create the campaign, ad sets, and ad groups. By default, the agent runs every 30 minutes.
2. The agent creates one campaign, one ad set, and one ad group for each audience identifier, with the exception of device-based audiences in Google Ads, where all device-based audiences for the same action are associated with the same ad group.

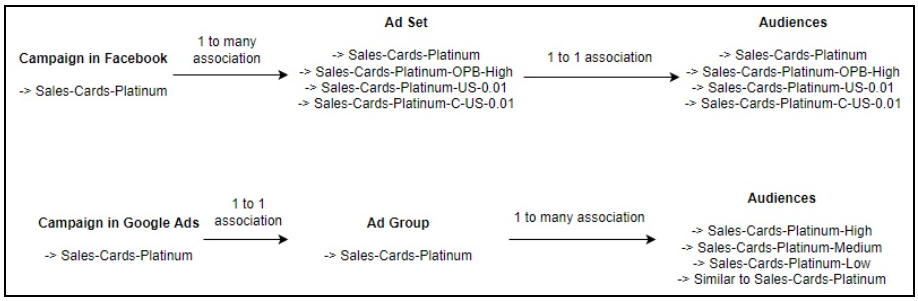
   Paid Media Manager creates one campaign for each audience identifier. Audience identifiers are derived from the business issue, business group, and name of an action that has been enabled for paid synchronization. For example, if the business issue is Sales, the group is Cards, and the action is called Platinum, the generated audience identifier is Sales/Cards/Platinum. The campaign generated in the destination platform for the audience identifier is named Sales-Cards-Platinum, with the exception of device-based audiences in Google Ads, where the name of the generated campaign is slightly different and contains the suffix Device, for example, Sales-Cards-Platinum-Device.

3. The agent associates the campaigns, ad sets, ad groups, and audiences.
4. The agent can associate bid modifiers. By default, bid modifier is set to 1%. This value is configurable manually from the Google Ads network along with setting a shared budget.

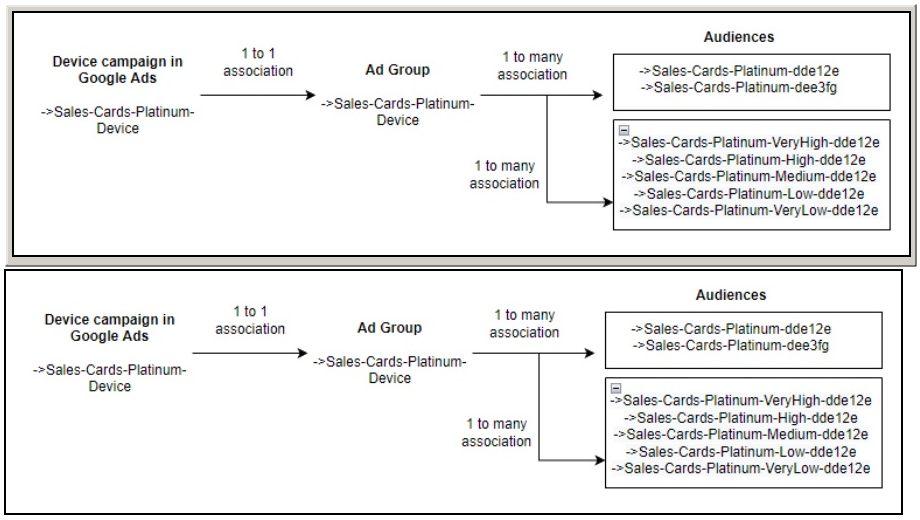
   For more information, see following documentation about bid modifiers.

One campaign can be associated with several ad sets or ad groups, as in the following figure:





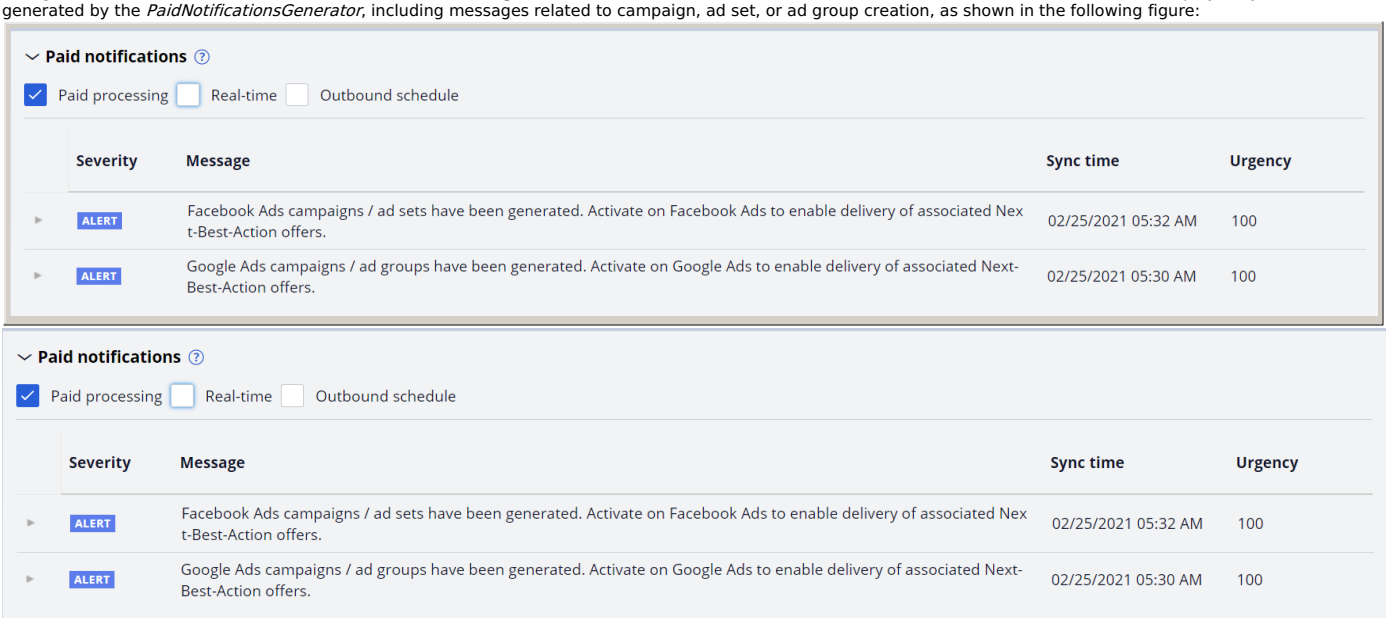
One ad set or ad group can only be associated with one audience, except for device-based audiences in Google Ads, where one ad group can be associated with multiple device-based audiences, as in the following figure:



Frequently asked questions about Paid Media artifacts

How can I verify whether campaigns, ad sets, or ad groups were generated correctly?

In Pega Customer Decision Hub, click **Next-Best-Action Designer Channels** , and scroll down to the **Paid notifications** section. This section displays any notifications generated by the *PaidNotificationsGenerator*, including messages related to campaign, ad set, or ad group creation, as shown in the following figure:



By default, *PaidNotificationsGenerator* runs every hour.**Tip:** Operators with access to the Pega Customer Decision Hub database can also check the results directly in the *mkt_paidmedia_campaigns* and *mkt_paidmedia_adsets* tables.

What are the next steps?

After Paid Media Manager has generated the campaigns and ad set or ad group scaffolding with default settings, adjust them as needed to suit your business requirements. When all the settings are configured to your requirements, run the campaign in the destination platform.

Enabling segment updates on paid destinations

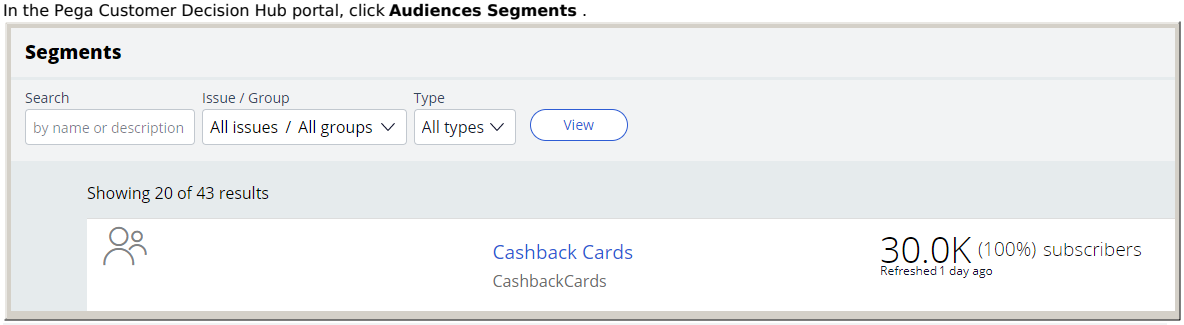
After you configure the connection with a paid destination, you can use your existing marketing segments to target individuals on the ad platform. Create paid audiences that are based on existing segments by synchronizing your List or Criteria Segments with a paid destination. Since segments can be based on multiple factors, such as traits or AI analytics, they are a powerful tool to build audiences in the paid destination.

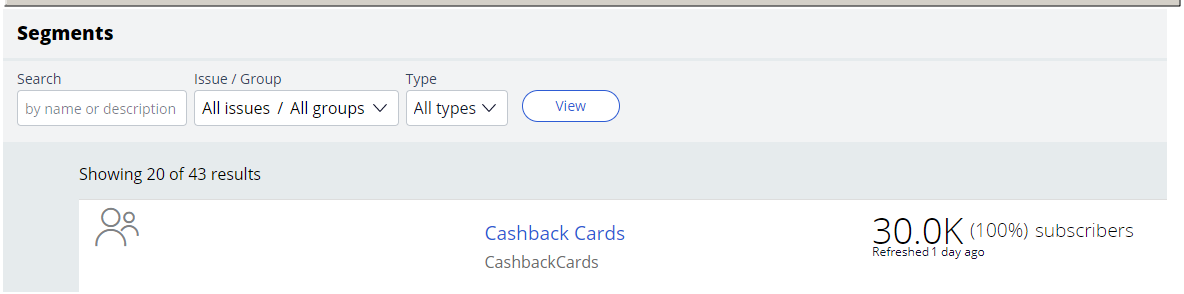
Pega Customer Decision Hub

For example, UPlus Bank, which has recently implemented Pega Customer Decision Hub, wants to extend the segments to paid destinations such as Facebook. A segment which includes individuals who have agreed to receive information about new products and promotions is enabled for paid synchronization. The segment includes an individual called Troy Murphy. Since Facebook now recognizes Troy as an individual to whom UPlus Bank wants to advertise their product offers, Troy starts seeing ads from UPlus Bank in his Facebook feed.

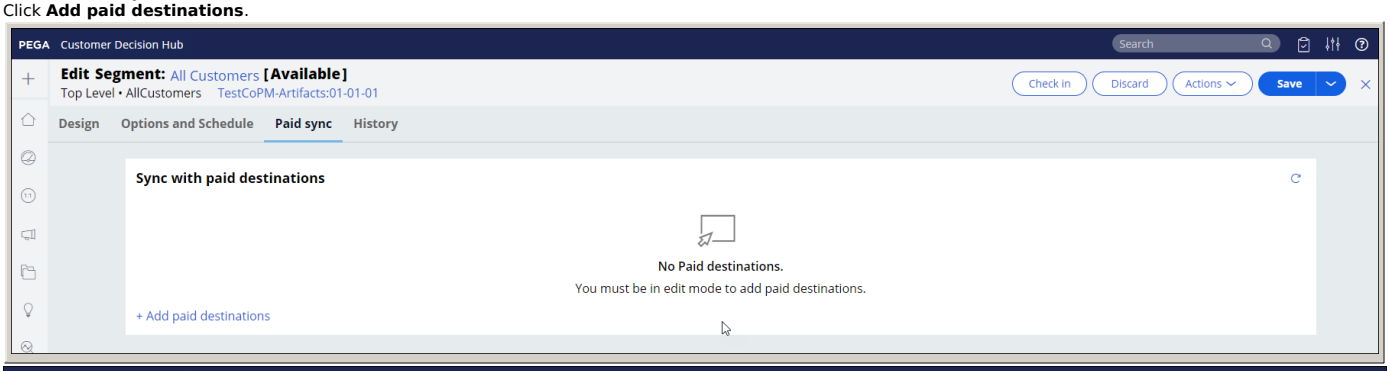
Do the following steps for each segment which you want to use to create audiences in a paid destination:

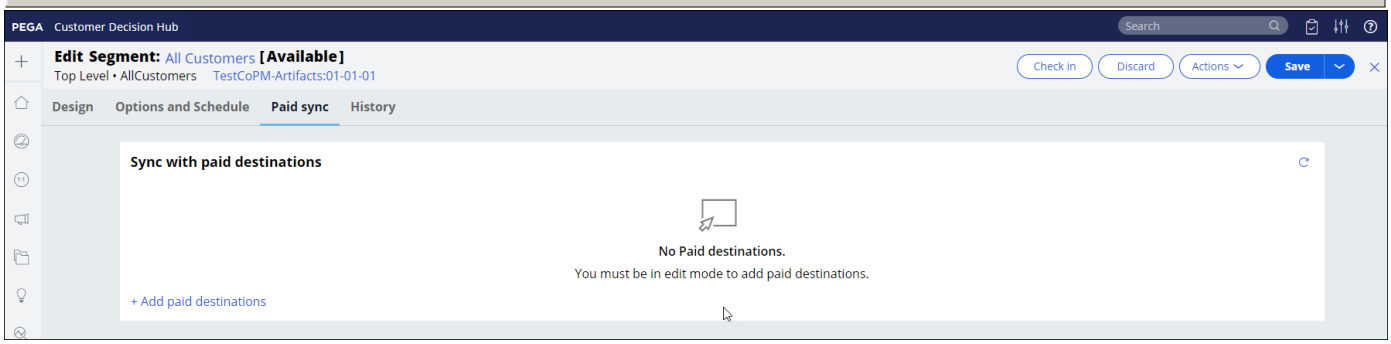
1. In the Pega Customer Decision Hub portal, click **Audiences Segments** .





2. Open a segment.
3. Click the **Paid sync** tab.
4. Click **Add paid destinations**.



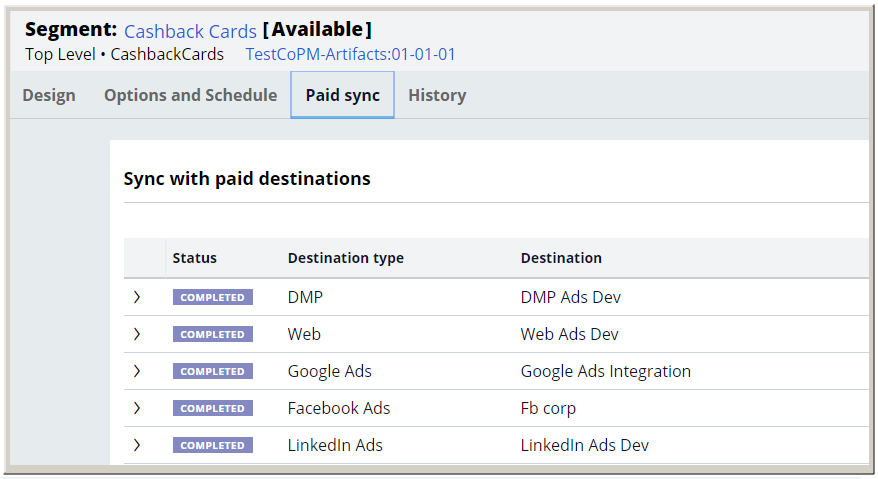


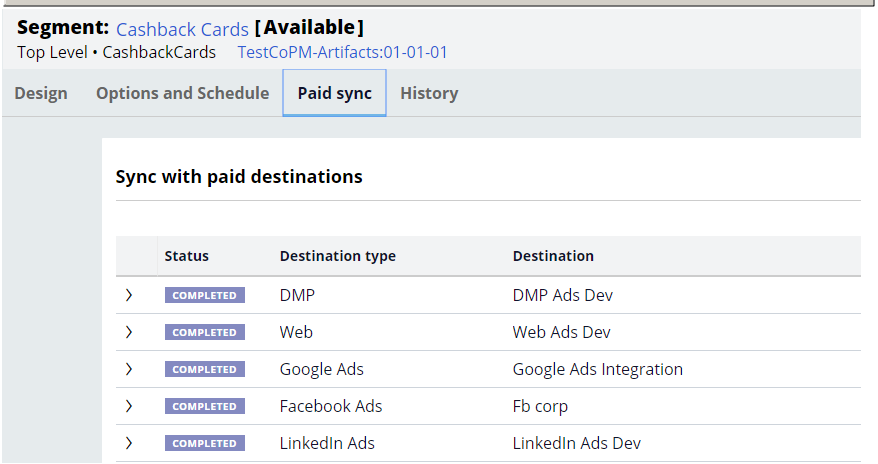
5. Select one or more paid channels to synchronize with this segment.

Result:

For CRM S2S-based destinations, such as Facebook Ads, Google Ads, and LinkedIn Ads, each segment update triggers a corresponding audience update. For web-based destinations, the updated segment information is available when the Paid Audience Web service is called. For Data Management Platform destinations, scheduled runs trigger the updates.

You can view the synchronization status for each destination in the Status column. Click a row for more details.





Errors in Paid Segment processing are recorded as Paid Notification work objects. For more information, see Errors and Warnings.

If you disable Paid Sync for a Segment, audiences created from that segment are deleted from the Paid destination during the next synchronization.

Communicating conversions to destination platforms to optimize return on ad spend

Paid Media Manager enables the same conversion events that drive internal decisioning and model-based learnings to help drive the performance of ad spend by optimizing the advertising targeted at known individuals.

Pega Customer Decision Hub

People-based offline conversions (that is, information about individuals who have completed an action (e.g. 50K Platinum Credit Card) provide valuable data and signals to advertising spend. Organizations in industries where the vast majority of conversions are to individuals in a known state (for example, retail banking, insurance, or communications) are uniquely positioned to take advantage of people-based conversion functionality provided on modern ad platforms. For example, when an individual completes a 10,000 online bonus points offer travel credit card action, that event can be communicated to an ad platform to optimize bids and targeting for associating ad spend. Paid Media Manager can do so through people-based APIs.

Paid Media Manager provides the option to integrate with the people-based conversion API provided by Facebook Ads and Google Ads to continuously deliver data about conversion events, including offline conversions, directly to the advertising platform. This data can then be used to optimize bid strategies towards actual conversion, as opposed to commonly used intermediary metrics like clicks and impressions. The event information is captured by using the same CaptureResponse API that is used to drive adaptive and predictive models in Pega Customer Decision Hub.

Offline conversions and the CaptureResponse service

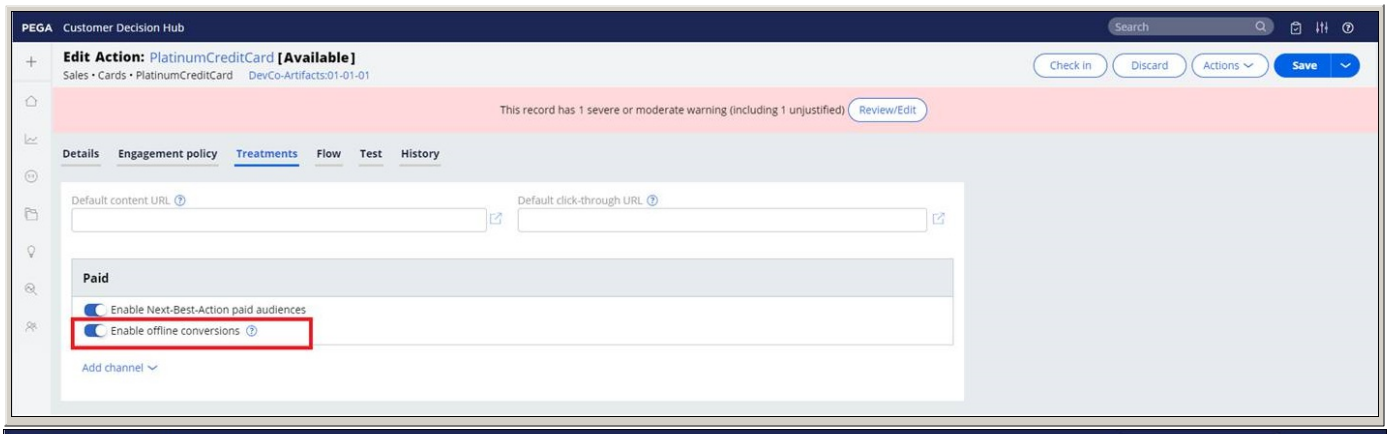
To learn how known individuals respond to actions across owned and paid channels, the AI adaptive and predictive models that can drive the decisioning process in Pega Customer Decision Hub use events captured by the CaptureResponse service. Information obtained from capturing individuals' responses to actions can be used, for example, to increase the efficiency of your advertising by enabling bids for associated ads to optimize for actual conversion as opposed to intermediary metrics like impressions and clicks. **Tip:** Any action where the outcome is Accepted is considered to be a conversion. Since all communications with known individuals are recorded in interaction history, conversions that happen offline are automatically recognized and associated with the original action.

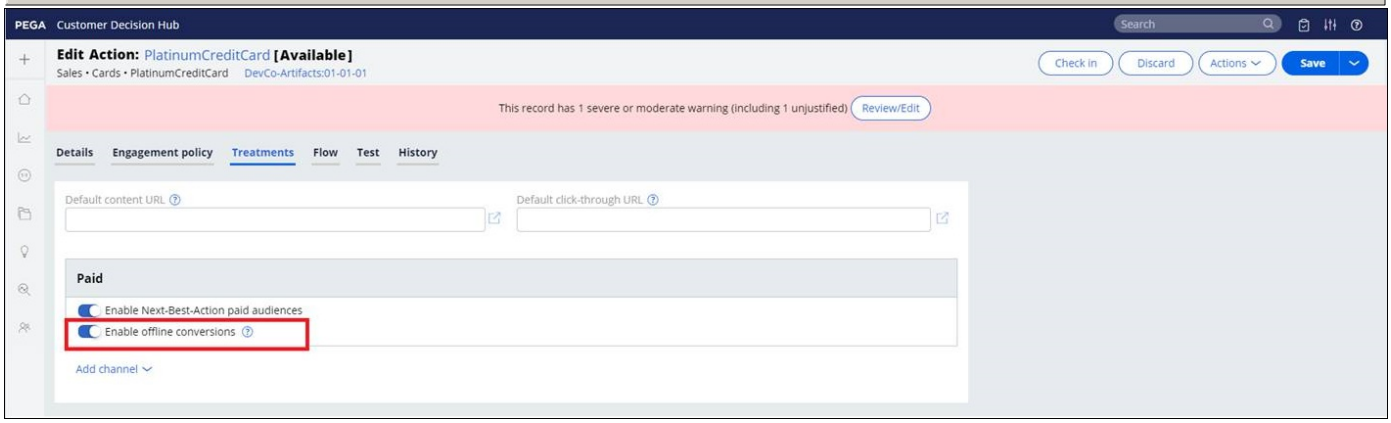
Paid Media Manager makes it possible to bring information about conversions outside of the enterprise firewall and securely communicate it to destination ad platforms. This is done through the same Next-Best-Action framework governing owned and paid channels, and the adaptive and predictive models informing both.

Understanding how offline conversion data from the CaptureResponse service is sent to paid destinations

The conversion synchronization process consists of the following stages:

1. After the offline conversion sync has been enabled and at least some actions have been enabled for offline conversions, an individual receives and accepts an action that has been enabled for offline conversions. The acceptance is the factor that triggers the conversion capture. **Tip:** By default, all actions that are enabled for paid synchronization are also enabled for offline conversions. If you do not want a particular action (for example, a service action) to be included in conversion s, you can set the Enable offline conversion toggle on the Treatments tab of the action to disabled.





2. The CaptureResponse service is invoked and stores interaction details such as the ID of the actions presented to the customer, and the customer's response in the *pr_data_ih_fact* Interaction History database table.
3. The *PrepareOfflineEventData* activity iterates over the CaptureResponse results and prepares conversion-related event data for sending to a paid destination.
4. The *OfflineConversionRTTrigger* job scheduler runs at specific intervals (by default, once every hour) and triggers the synchronization *RealTimeOfflineConversionTriggerSync* activity that sends the event data to a destination platform.**Note:** The *OfflineConversionRTTrigger* job scheduler is disabled by default. To synchronize your offline conversion data, you must enable it in the **Resources Jobs** menu of Admin Studio.
5. Conversion event data is communicated to the destination platform and associated with a conversion event. Paid Media Manager generates an audience containing all individuals who have converted for this action, with the following naming pattern: <Issue-Group-Action>-C, for example, Sales-Cards-PlatinumCreditCard-C.**Note:**

   On Google Ads, you must configure the offline event set before you can synchronize event data. For more information, see Enabling offline conversion push for Google Ads.

   On Facebook Ads, the offline event set can be automatically created by the conversion push, but it is also possible to associate the conversions with a previously created offline event set. For more information, see Optional: Synchronizing the conversion data with an existing offline event set in Facebook Ads.

6. You can then use the data tooptimize bids, create custom lookalike audiences in the destination platform, or measure the effectiveness of your ads through attribution reporting.

Using people-based conversions to optimize bids

Offline conversion data can be used to optimize bidding by configuring advertising campaigns to be optimized for conversions instead of clicks or impressions. For more information about the process, refer to the following destination-specific topics:

- Facebook Ads: Ad Account, Ad Campaigns
- Google Ads: About bid adjustments

Using conversions audiences to target lookalike audiences based on individuals who converted

After the conversion data is communicated to a destination platform in the form of conversion audiences, you can use it to create lookalike audiences based on individuals who have converted, and use them to better target your ads.

For more information about the process, refer to the following destination-specific topics:

- Facebook Ads: Create a Lookalike Audience**Note:** For Facebook Ads, to use lookalike audiences, you must first enable the **Lookalike audience** option in the destination settings. For more information, see Prospecting with Next-Best-Action paid audiences.
- Google Ads: Add similar audiences to your targeting

Using people-based conversions with attribution reporting

Offline and online conversion data helps measure the real effectiveness of your ads by providing you with data from a variety of channels. After conversion information is communicated to the platform, you can use it to measure the effectiveness of your advertising keywords. For more information about the process, refer to the following destination-specific topics:

- Facebook Ads: Offline Conversions API
- Google Ads: About attribution reports

- Enabling offline conversions

Enabling offline conversions

To enable offline conversions, you must configure relevant settings in Paid Media Manager and in the destination platform.

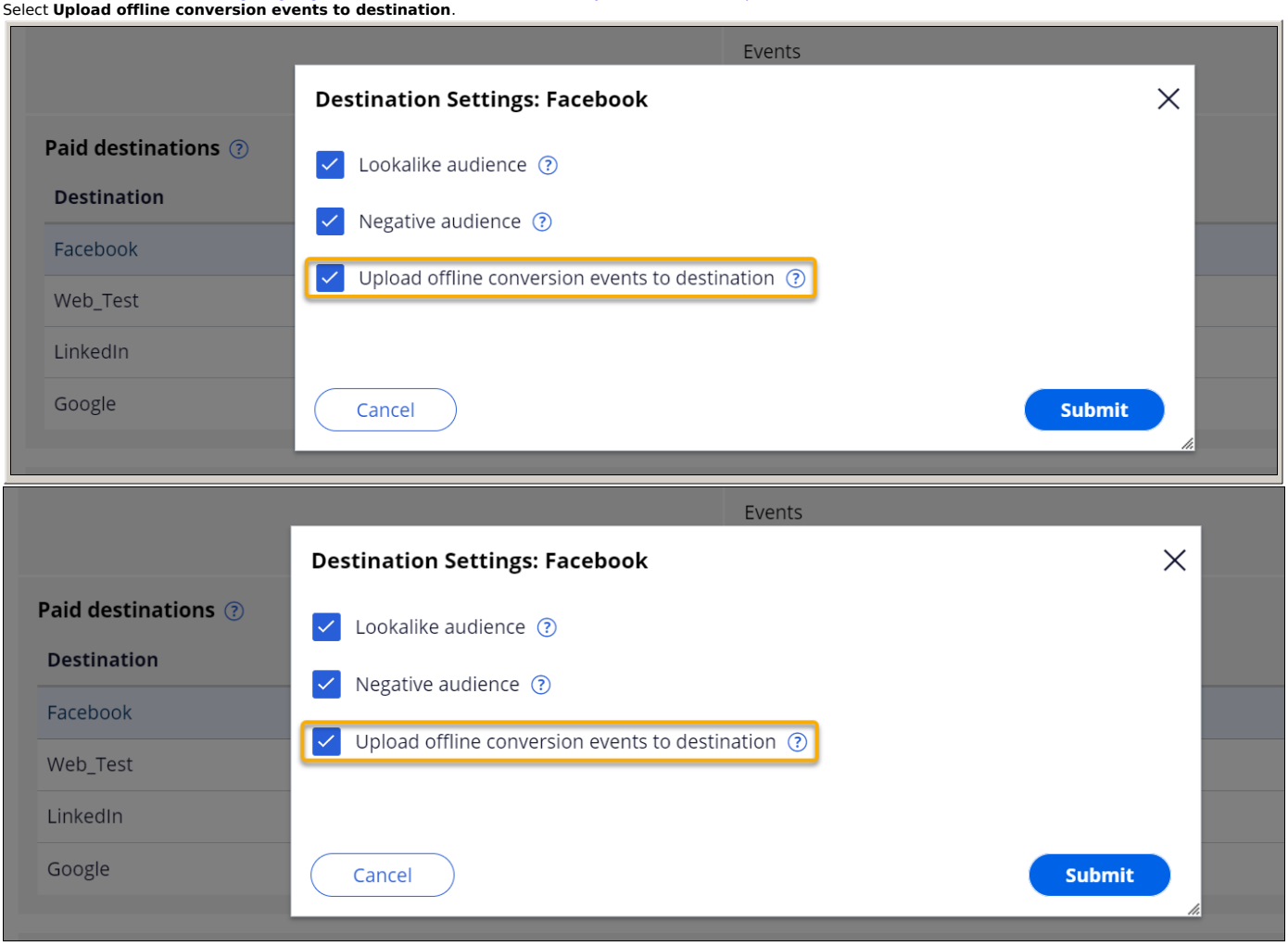
Pega Customer Decision Hub
Before you begin: Work with your implementation specialist to ensure that the *OfflineConversionRTTrigger* job scheduler is set to enabled. For more information, see Enabling the offline conversion job scheduler for Paid Media Manager in the *Pega Customer Decision Hub Implementation Guide*. For destination-specific instructions, refer to the following sections:

- Enabling offline conversion push for Facebook Ads
  - Optional: Synchronizing the conversion data with an existing offline event set in Facebook Ads

- Optional: Synchronizing historical conversion data in Facebook Ads
- Enabling offline conversion push for Google Ads

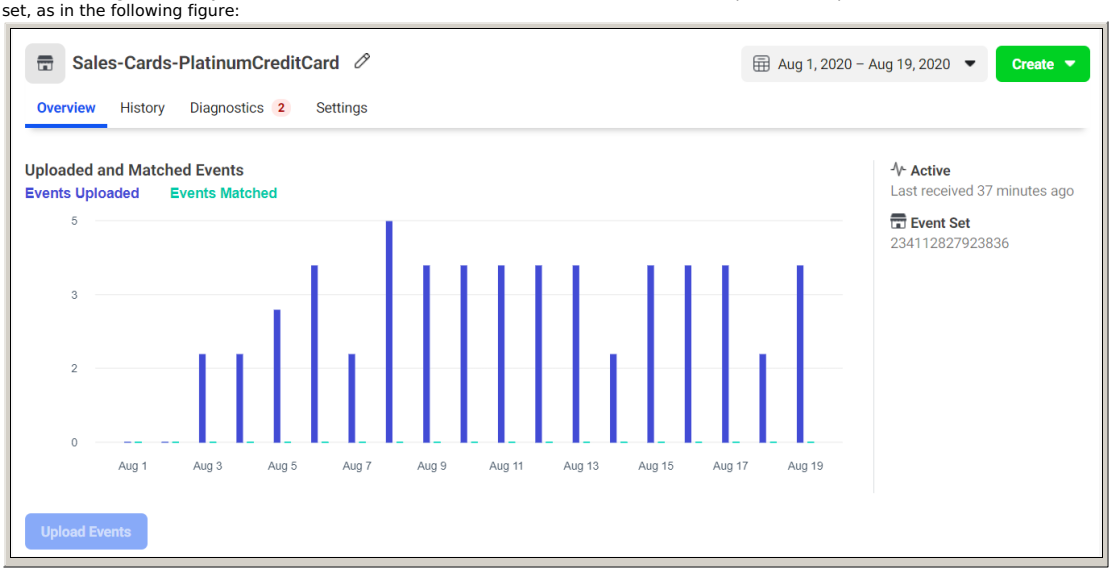
To automatically push live offline conversion data to Facebook Ads whenever a conversion is captured through the Capture Response REST service, do the following steps:

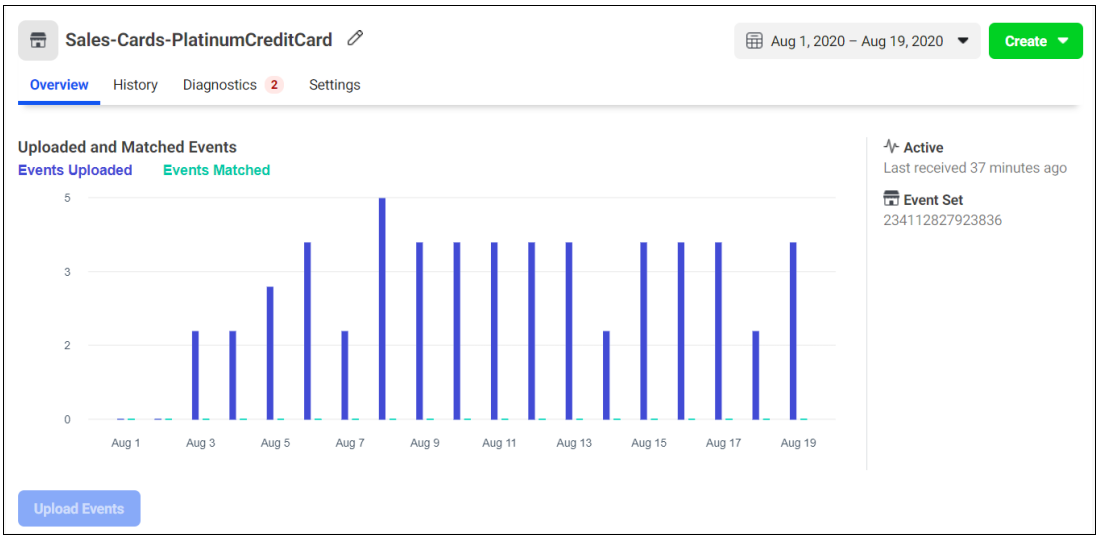
1. In App Studio, click **Next-Best-Action Designer Channels** .
2. In the Paid Media section, edit or create a destination.
   For more information, see Configuring Pega Customer Decision Hub to enable synchronization with paid destinations.
3. Select **Upload offline conversion events to destination**.



4. Click **Submit**.

Result: During the next synchronization for this destination, metrics related to action acceptance are uploaded to Facebook Business Manager and linked to the offline event set, as in the following figure:





If no offline even set existed in Facebook Business Manager, the synchronization creates it with the following naming convention: <Issue-Group-Offer>, for example, Sales-Cards-PlatinumCreditCard.

You can now use the conversion data for use cases such as attribution reporting, bid optimization, and lookalike audiences. For more information, see Communicating conversions to destination platforms to optimize return on ad spend.

If you want to synchronize the conversion data with an offline event set that you previously created in Facebook Business Manager, make sure that the name of the event set has the format <Issue-Group-Offer>.

For example: If the CaptureResponse service is triggered for when the action name is PlatinumCreditCard and the outcome is Accepted, the name of the event set must be Sales-Cards-PlatinumCreditCard. For more information, see Offline Conversions.

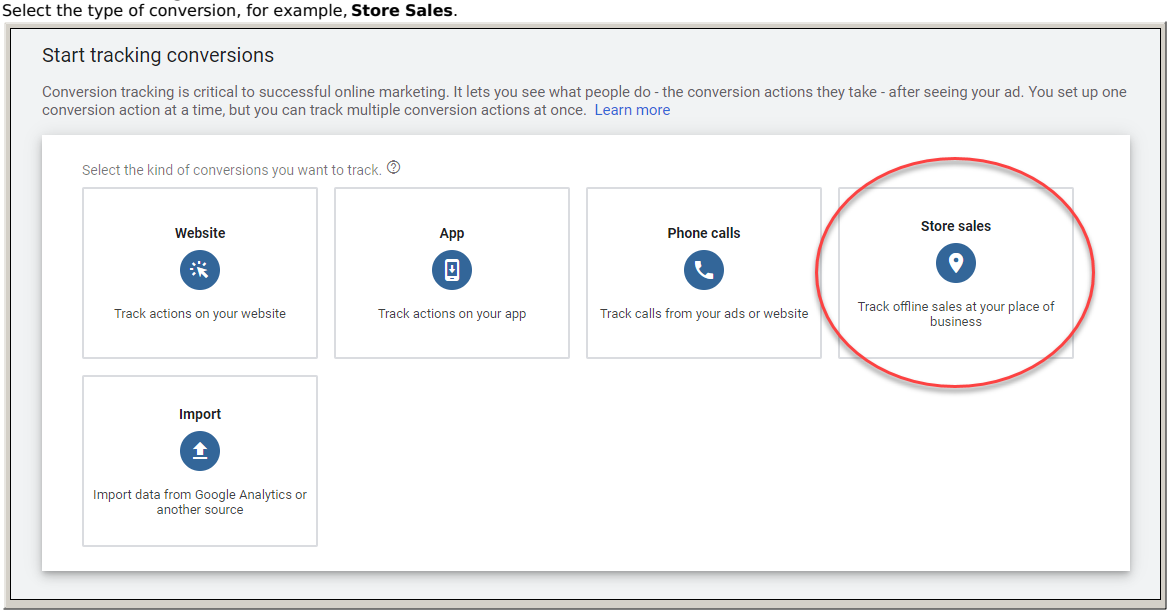
In addition to sending data about new interactions to Facebook Ads, you can also manually trigger a synchronization of historical data, previously collected by the CaptureResponse service for a specific time period.

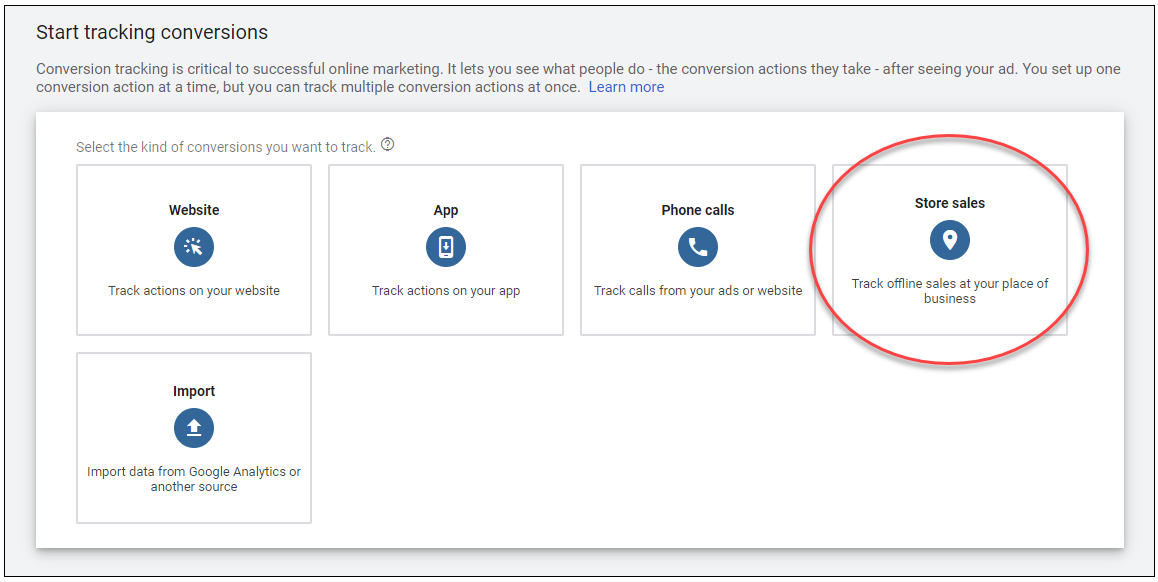
For example, if you are starting to implement the offline conversions functionality, and you have already implemented a CaptureResponse pixel on your website, you can do a one-time synchronization that will send the captured data to Facebook.

1. In App Studio, click **Settings Channels Paid media Facebook Ads** .
2. In the **Historical conversions** section, select **Push historical conversions**.
3. In the **Look-back** window, select the time frame for historical conversion data.
4. Click **Run** to send the conversion data to Facebook Ads.
   Note: If historical conversion push times out when reading from the *pr_data_ih_fact* interaction history fact table, the cause of the timeout is likely the large number of records present in the table. To solve the issue, create an index on the *pzoutcomeid* property.

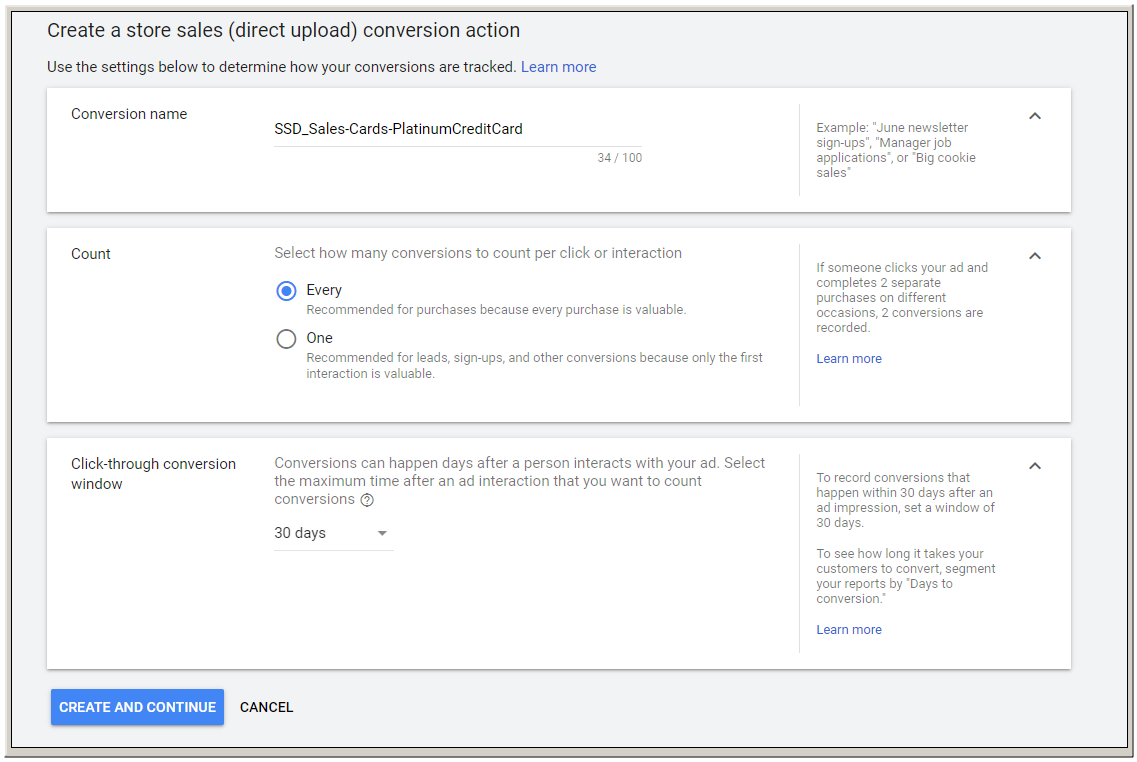
To automatically push live offline conversion data to Google Ads whenever a conversion is captured through the Capture Response REST service, you must first configure a conversion action in Google Ads, then enable the offline conversion push in Next-Best-Action Designer.

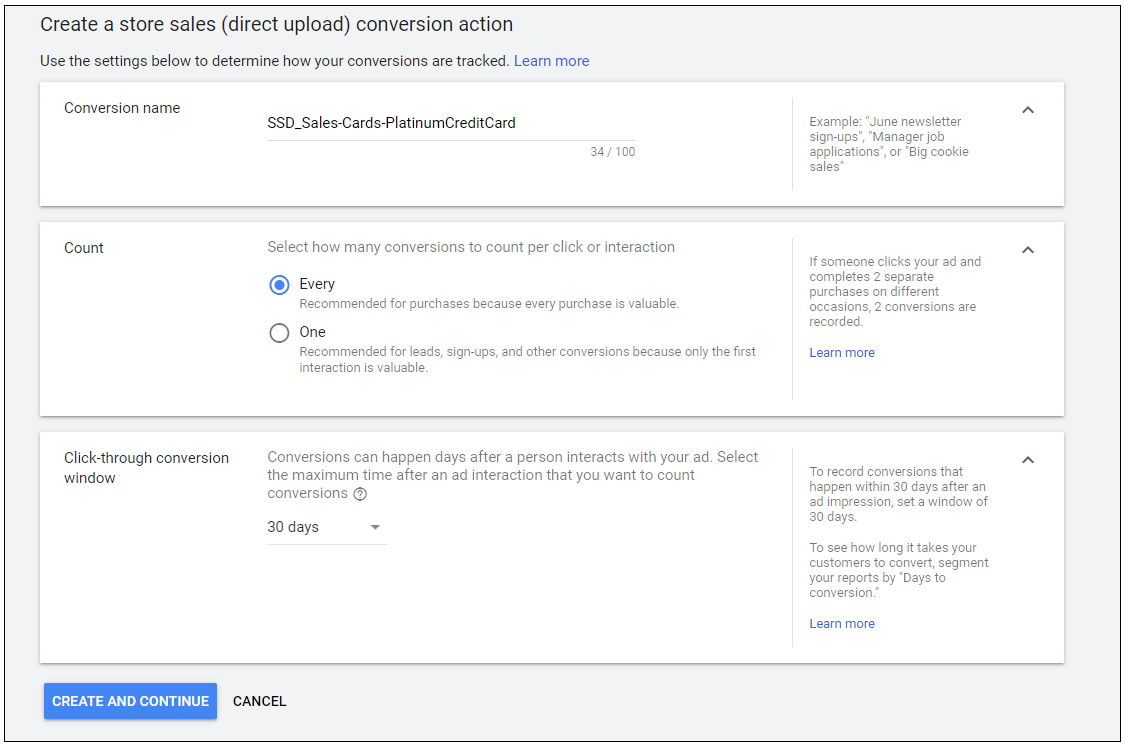
1. Log in to your Google Ads account.
2. Select an account that has SSD transactions enabled.
   Tip: To enable SSD transactions for your account, contact your Google representative.
3. Click **Tools & Settings Conversions Add** .
4. Select the type of conversion, for example, **Store Sales**.





5. 5. Fill out the action details. The action name must have the format *SSD_<Issue-Group-Offer>*. **For example:** If the CaptureResponse service is triggered for the PlatinumCreditCard action in the Sales issue and Cards group, the name of the event set must be *SSD_Sales-Cards-PlatinumCreditCard*.
   For more information, see What are conversion action sets?.
6. Click **Create and Continue**.

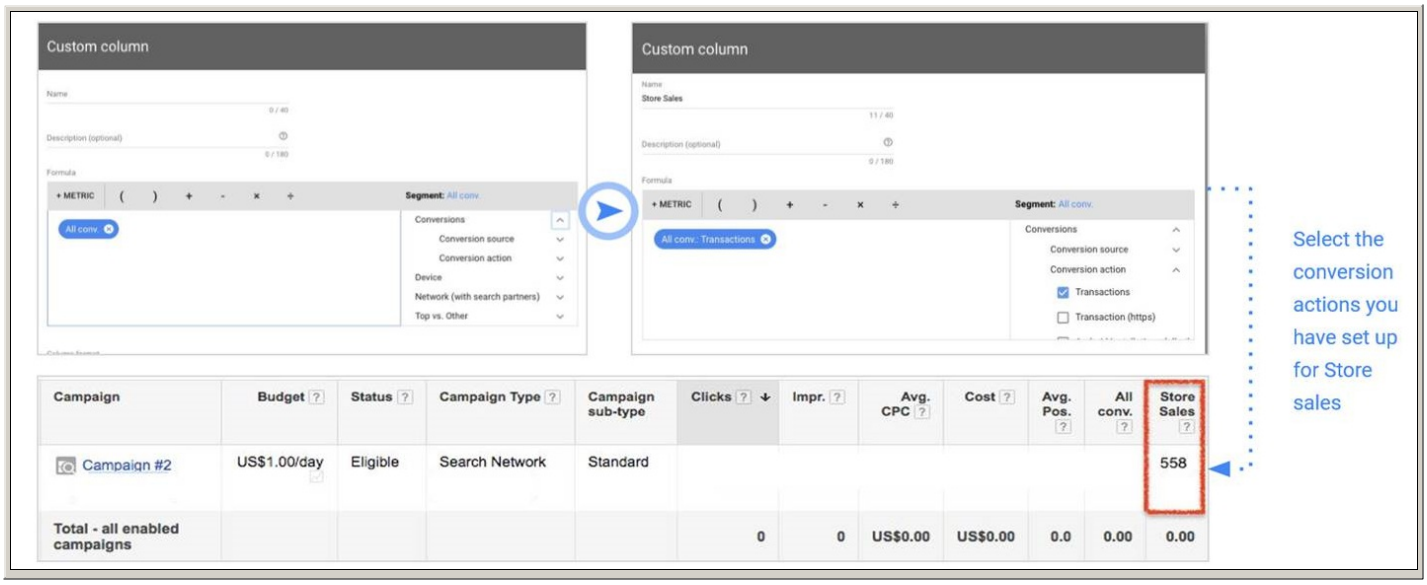


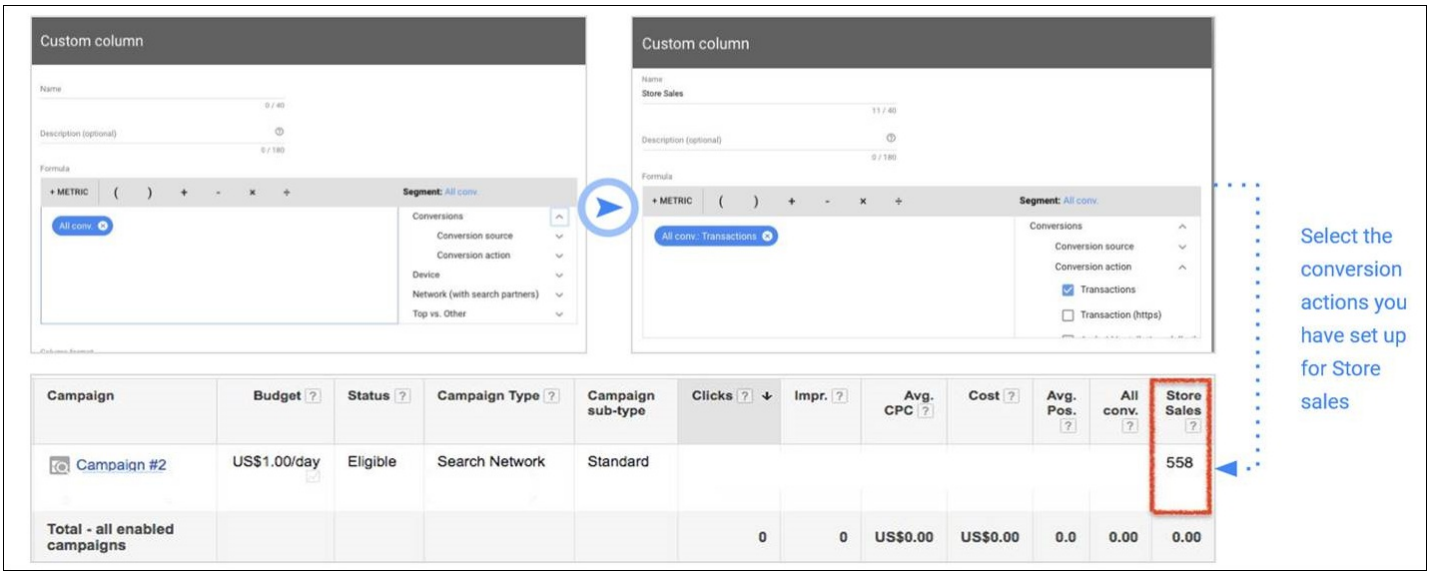


7. In App Studio, click **Next-Best-Action Designer Channels** .
8. In the Paid Media section, edit or create a destination.
   For more information, see Configuring Pega Customer Decision Hub to enable synchronization with paid destinations.
9. Select **Upload offline conversion events to destination**.
10. Click **Submit**.

Result:

During the next synchronization for this destination, metrics related to action acceptance are uploaded to a conversion action created in Google Ads. To view the metrics, create a custom column to show the aggregated value of conversions, as shown in the following figure:





You can now use the conversion data for use cases such as attribution reporting, bid optimization, and lookalike audiences. For more information, see Communicating conversions to destination platforms to optimize return on ad spend.

Using action eligibility to reduce wasted ad spend on people-based destinations

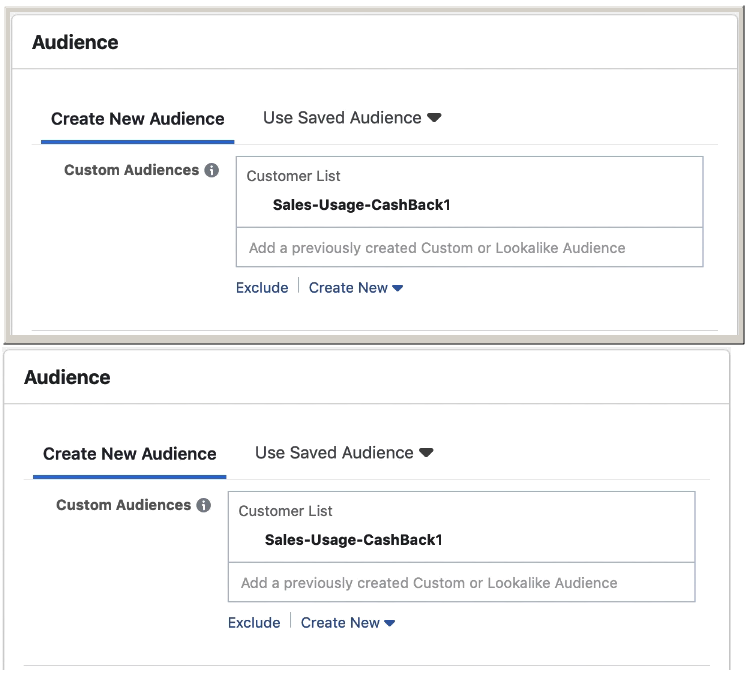
Advertising on paid channels without a centralized Next-Best-Action strategy has the risk of wasting ad spend on ads which are not relevant to the individuals whom you want to target. For example, an individual may see an ad for a product which he or she already has, because the information is not communicated from your Pega Customer Decision Hub application to the paid destination where you are advertising. The cost of that impression is likely wasted and would be better spent on showing the ad to an individual for whom the product offer is more relevant.

Pega Customer Decision Hub

Paid Media Manager addresses this issue by communicating relevant Next-Best-Action decisions to the destination platform, according to the engagement policies specified in the Eligibility section of the Next-Best-Action Designer. This means that after individuals with accounts on the destination platform are matched based on a one-way hash of their personally identifiable information (PII), such as an email address, phone number, or device ID, they are placed in audiences only for products for which they are eligible. This ensures that ad spend is not wasted by showing the individual offers which are not relevant to them.

For example, while browsing his Facebook feed, Troy may see ads from UPlusBank, but before UPlusBank enabled Paid Media Manager to communicate Next-Best-Action decision to paid channels, Troy sometimes saw ads which were not relevant to him, for example, ads for the same type of checking account which he already had.

After UPlusBank extends their Next-Best-Best-Action to paid channels, information about each individual's Next-Best-Action is communicated to Facebook Ads. Individuals are placed into custom audiences, with each audience consisting only of individuals who are eligible for a particular offer. Each audience is associated with an ad set, which contains one or more ads for the offer. One of Troy's Next-Best-Actions is determined to be a cashback credit card offer, so he is added to the Sales-Usage-CashBack1 audience set, which is targeted with an ad set configured with associated advertisements.



As a result, Troy now sees an ad for the cashback credit card in his Facebook feed.

The below tasks describe how to configure Paid Media Manager so that it can create audiences based on the offer eligibility data you have collected in your Pega Customer Decision Hub application.

- Triggering paid updates through Next-Best-Action

Triggering paid updates through Next-Best-Action

After you configure the connection to a paid destination and specify how individuals are matched to their accounts on that destination, use the Next-Best-Action Designer to enable Next-Best-Action decisions for paid channels.

Pega Customer Decision Hub

The Next-Best-Action Designer allows you to define the Next-Best-Action decisions and requirements which apply to the whole organization, or to specific business issues or groups. You can use it to select the channels on which actions, such as product offers, are communicated to individuals. In addition, you can specify the triggers which cause the actions to be communicated to individuals, for example, scheduled outbound runs or real-time container updates.

To extend your Next-Best-Action to paid channels:

1. Log in to the Pega Customer Decision Hub portal as an administrator.
2. Click **Next-Best-Action Designer Channels** .
3. In the **Paid Media > Paid Destinations** section, click **Configure** to select one or more paid destinations.
4. Specify the trigger or triggers for the audience update.
   Tip: In Pega Customer Decision Hub 8.5, you can define triggers for both positive and negative audiences, that is, the individuals who should or should not receive a specific action. Keep the following considerations in mind when defining triggers for each audience type:
   - Positive audiences - As a best practice, always define the trigger at the top level of your business hierarchy. Real-time events calls at lower levels of the hierarchy can produce undesired results, for example, removing an individual from all eligible audiences other than those at the same hierarchy level as the event.
   - Negative audiences - You can define the container or event at any level of the hierarchy. When the container or event is triggered, Paid Media Manager updates negative audiences only for the actions on that level of the hierarchy. For example, if you define a real-time container at the Sales-Cards level, only audiences for actions from the Cards group of the Sales issue will be updated.

   For more examples on how to configure and implement this capability, see Enabling channels.
5. **Optional:** For Google Ads and Facebook Ads destinations, in the **Paid destinations section**, click **Configure Negative audience** to increase the accuracy of your targeting by excluding individuals who are ineligible for an action. For more information, see
6. **Optional:** For Google Ads and Facebook Ads destinations, in the **Paid destinations section**, click **Configure Lookalike audience** . This creates an audience of customers who share the traits of your existing customers. For more information, see Configuring Facebook Lookalike Audiences and Generating Google Similar Audiences.
7. **Optional:** As a best practice, you should use a single, primary paid configuration to manage all paid actions. However, if required, you can define additional paid configurations, for example, to use different paid strategies for different levels of your business hierarchy:
   a. In the **Channels** tab of the Next-Best-Action Designer, click **Actions Add paid configuration** .
   b. In the new **Additional paid configuration** section, in the **Issue / Group** list, select the issue and group or groups to which the schedule applies.
   c. Select the paid strategy, triggers and destinations for this hierarchy level, as described in steps 1-3.
8. Click **Save**.

- Reducing wasted ad spend with negative audiences and next-best-action
- Enabling actions for paid channels
- Streamlining Campaign Generation for Next-Best-Action Paid Audiences

Reducing wasted ad spend with negative audiences and next-best-action

Negative audiences, introduced in Pega Customer Decision Hub 8.5, contain all individuals that were not selected for a specific action due to eligibility, low propensity, and so on. You can now reduce wasted ad spend by using negative audiences to negative target ad campaigns regardless of whether you are also targeting the positive audience.

Pega Customer Decision Hub

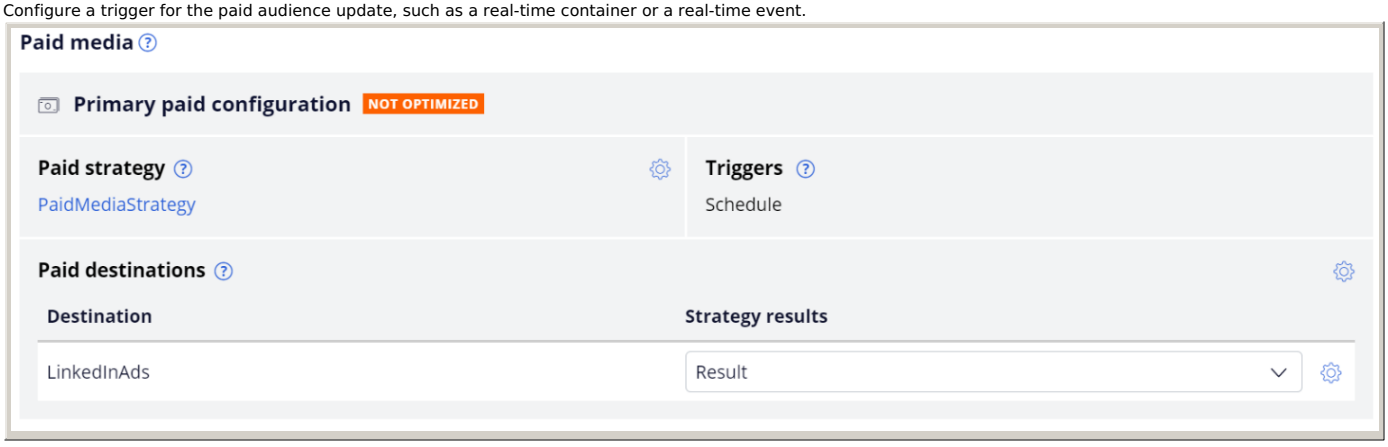
Advertisers spend the bulk of their advertising budgets on anonymously targeted campaigns – generally to acquire new customers. That results in wasted advertising to individuals that are ineligible or that have low propensity for the underlying action. The advertising spend to these individuals is not only wasteful but also reduces the performance of acquisition advertising budgets. Paid Media Manager can limit this waste on major ad platforms, and in doing so increase the performance of acquisition advertising campaigns that drive the vast majority of advertising today.

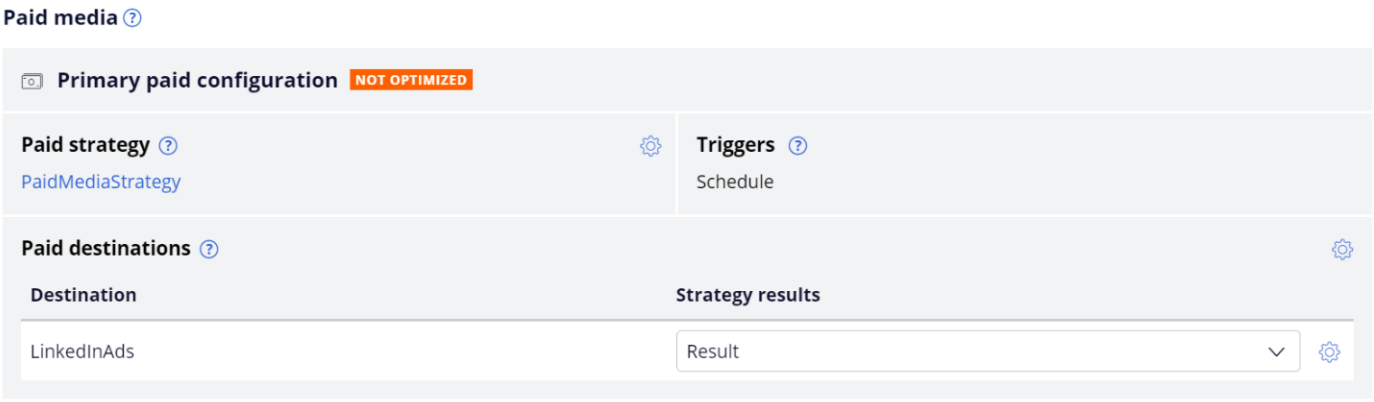
Advertising campaigns for a particular action can then be configured to exclude these individuals from being served ads. These negative audiences are updated each time an individual triggers Next-Best-Action on any channel. Also, they are updated in nightly batch runs across all known individuals.

Enabling negative audiences

To enable negative audiences for a destination:

1. Log in to the Pega Customer Decision Hub portal as an administrator.
2. Click **Next-Best-Action Designer Channels** .
3. In the **Paid Media > Paid Destinations** section, click the **Configure** icon, and then select the **Negative audience** check box to increase the accuracy of your targeting by excluding individuals who are ineligible for an action.
4. Click **Save**.
5. Configure a trigger for the paid audience update, such as a real-time container or a real-time event.





Tip: You can define the container or event at any level of the hierarchy. When the container or event is triggered, Paid Media Manager updates negative audiences only for the actions on that level of the hierarchy that are enabled for paid synchronization. For example, if you define a real-time container at the Sales-Cards level, only audiences for actions from the Cards group of the Sales issue will be updated.

Viewing negative audiences in Pega Customer Decision Hub

You can view current negative audiences on the **Audiences > Paid Audiences** landing page in the Pega Customer Decision Hub portal. Each negative audience has the naming convention Issue-Group-Action-N, for example, Acquisition-Credit Cards-Platinum Credit Card 2020-N.

Adding negative audiences to a Facebook ad set or Google ad group

After a paid run or real-time container update, these audiences are communicated to the destination platform. Negative audiences can then be excluded from targeting by using platform-specific mechanisms on platforms such as Facebook Ads or Google Ads, demand-side platforms (DSPs), and advertising networks.

For more information about configuring the mechanism for Facebook Ads and Google Ads, see the following topics:

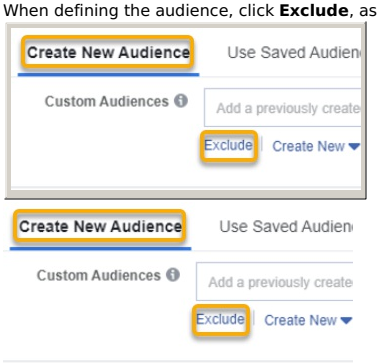
Excluding negative audiences from Facebook Ads targeting

To exclude a negative audience from being targeted by an ad set on Facebook, add it to the ad set after it is communicated to Facebook Ads.

Tip:

This is a one-time configuration. The audience membership on the platform is continuously updated once the audience is created.

1. Log in to Facebook Ads Manager.
2. Click **Create** to create a new ad set. For more information, see Create an Ad Set in Ads Manager.
3. When defining the audience, click **Exclude**, as in the following figure:



4. Select the negative audience created by Paid Media Manager.

Excluding negative audiences from Google Ads targeting

To exclude a negative audience from being targeted by a Google Ads ad group, follow the steps described in the topic Exclude specific audiences from your targeting.

Appendix: Deprecated negative audience behavior in previous releases

In previous releases of Pega Customer Decision Hub, the Next-Best-Action negative target audience was used for negative targeting. Its primary function was to exclude known individuals from being targeted by ads aimed at anonymous users.

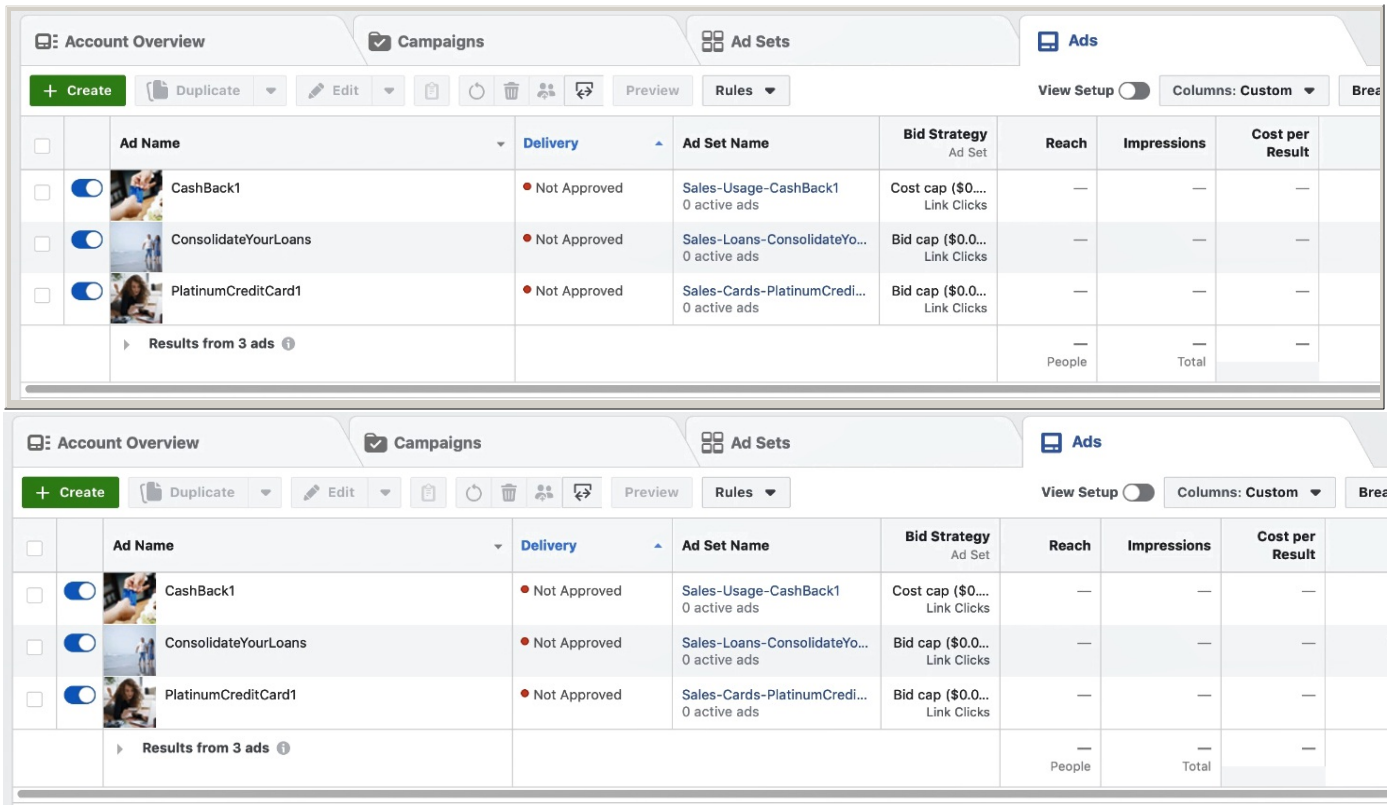
However, since Pega Customer Decision Hub 8.5, the Next-Best-Action negative target audience is superseded by negative audiences, which offer more granular results that indicate who should not be targeted for a specific action. The Next-Best-Action negative target audience is considered deprecated, and will be removed in a future version of the product.

Enabling actions for paid channels

Once you configure the paid destinations and enable it in the Next-Best-Action Designer, ensure that you have some customer actions which can be displayed in paid destinations.

Pega Customer Decision Hub

1. Create or edit an action, for example, a Credit Card action. For more information, see Managing actions.
2. On the Treatments tab of the action, select **Paid**. The action is now enabled for paid destinations.
3. In the paid destination, define the ad creative which should be displayed to individuals in your action's audience.



Result: Individuals like Troy, who belong to an audience linked to one of the ad sets, see the ads displayed, for example, while browsing their Facebook feed or watching videos on Youtube.

Streamlining Campaign Generation for Next-Best-Action Paid Audiences

To streamline the creation of advertising campaign on Facebook and Google destinations, enable the option to automatically create campaigns for each action and ad sets or groups for each paid audience.

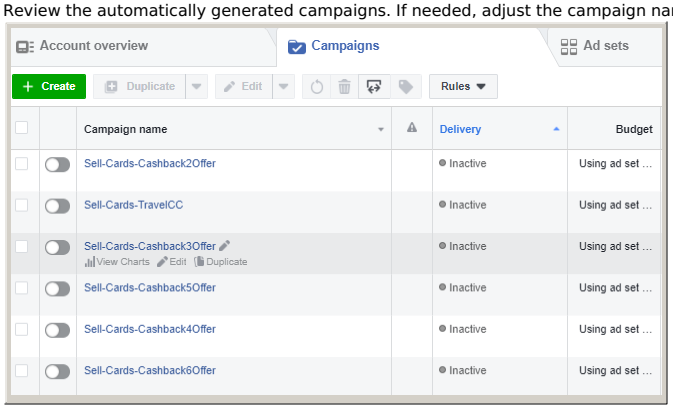
Pega Customer Decision Hub

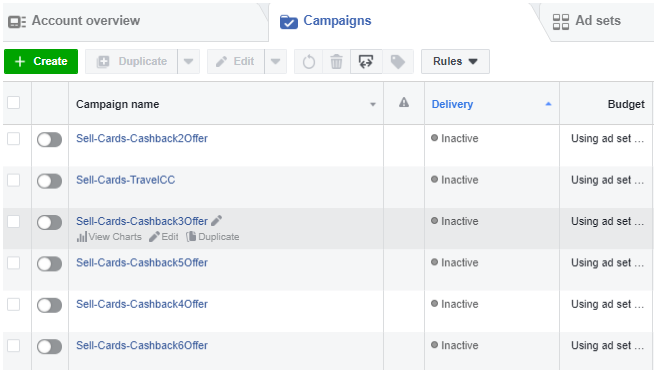
Advertisers must configure campaigns on ad platforms to use Next-Best-Action based paid audiences. For Google Ads and Facebook Ads, this task can be streamlined.

Enabling campaign generation for Facebook Ads and Google Ads creates a campaign for each offer. If a paid strategy is used, an ad set and ad group is generated for each priority level. These priority-specific ad groups/ad sets can be configured with a max bid aligned with the priority level.

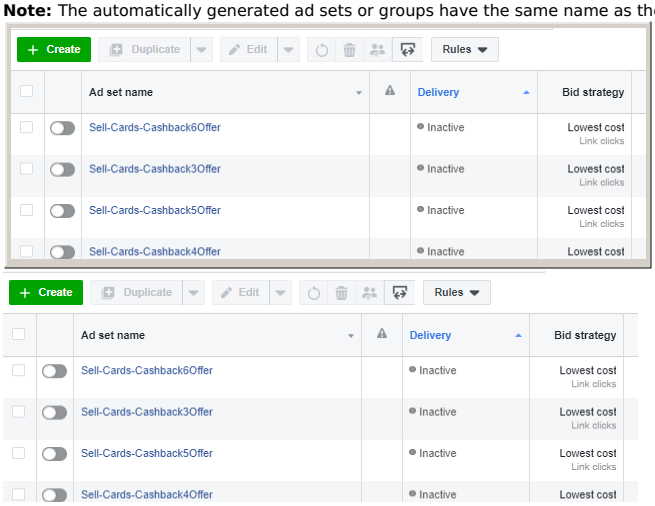
To enable this option for a specific Facebook Ads or Google Ads destination:

1. Log in to App Studio.
2. Click **Settings Channels Paid Media Facebook Ads** or **Google Ads**.
3. Edit or create a destination.
   Note: Do not use a manager account for Google Ads if you want to automatically create campaigns for this destination. Automatic campaign generation is not available for Google Ads manager accounts.
4. In the **Facebook/Google Ads assets generation** section, select **Enable campaign generation**.
5. Click **Apply**.
6. Enable the paid destination you created in the Next-Best-Action Designer.
   For more information, see Triggering paid updates through Next-Best-Action.
7. Once the individual data for this destination has been synchronized, for example, after a scheduled run, log in to Facebook Business Manager or to your Google AdWords account.
8. Review the automatically generated campaigns. If needed, adjust the campaign name, objective, budget, and other settings.





Note: The automatically generated ad sets or groups have the same name as the paid audience with which they are associated.



The association between the generated paid audience and the ad set enables the individual added to the audience to be targeted with the correct action at the correct willingness to pay. Altering this association prevents this mechanism from working.

9. Finish the configuration of the automatically generated advertising campaigns in the destination platform.

Responding to individual engagement on paid channels with real-time events

You can define real-time events that correspond to real-time engagement opportunities on paid destinations.

Pega Customer Decision Hub

Real-time events are entry points into the application which can be initiated by internal or external systems and can trigger the execution of outbound runs in Next-Best-Action Designer. For more information, see Responding to real-time customer engagement with real-time event triggers.

For example, you can configure a real-time event that triggers when an individual makes a large deposit into a savings account. If the source of the deposit is, for example, a pension payout or a lottery win, the deposit is a signal for the bank that they may take the opportunity to discuss investment options with the individual, or consider upgrading the individual to their Private Banking services. Paid Media Manager helps advertisers pursue this opportunity on paid channels by targeting the individual with relevant ads on Facebook Ads, Google Ads, and LinkedIn Ads destinations

1. Define a real-time event. For more information, see Responding to real-time customer engagement with real-time event triggers.
2. In the **Channels** tab of the Next-Best-Action Designer, in the **Triggered by Events** section, click **Configure**.
3. Select the event that you created.
4. In the **Business structure level** list, select the business issue and group with which the event is associated.
   Tip: As a best practice, always define the event trigger at the top level of your business hierarchy. Real-time events calls at lower levels of the hierarchy can produce undesired results, for example, removing an individual from all eligible audiences other than those at the same hierarchy level as the event.
5. Ensure that at least one Facebook Ads, Google Ads, or LinkedIn Ads destination is configured for the same issue and group.
   For more information, see Triggering paid updates through Next-Best-Action.
6. Configure an action and enable it for paid destinations.
   For more information, see Enabling actions for paid channels.
7. Click **Submit**.
8. Click **Save**.
9. Log in to your application as an operator with access to Dev Studio.
10. Save the *CDHEventSource* data set into your application ruleset.
11. On the **Settings** tab of the *CDHEventSource* data set, disable the authentication.
12. Create a real-time data flow run for the *ProcessEventFromSource* data flow of the class *PegaMKT-Data-Event*, as described in Triggering a real-time event with the Event Stream service.
    Example request payload:

    ```
    {

    "EventName" : "ApplicationAbandon",

    "CustomerID": "CID-5"

    }
    ```

Result: Audiences created after the event is triggered are synchronized to the paid destination or destinations selected in step 5. You can also view them on the **Paid Audiences** landing page in Pega Customer Decision Hub.

Reducing wasted ad spend based on action priority

To further optimize your advertising, use a paid strategy to convey offer priority to ad platforms which can then be aligned with the willingness to pay for each individual and action as a factor in targeting individuals with actions.

Pega Customer Decision Hub

Paid strategies let you place an individual in audience based on the action selection and the priority. You can then configure this action and priority-specific audience with the appropriate max bid.

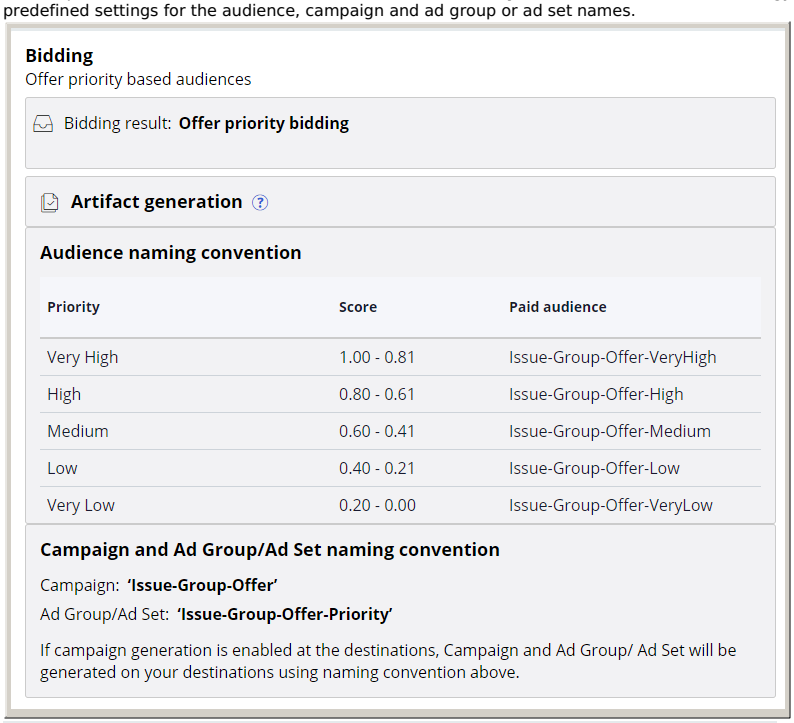
- Creating a priority-based strategy with the Strategy Builder

Creating a priority-based strategy with the Strategy Builder

Rather than create a paid strategy from scratch, you can use the Strategy Builder to generate it based on one or more optimization methods.

Pega Customer Decision Hub
The basic optimization method generates a strategy that groups individuals based on the priority of their action as determined by Next-Best-Action. The priority can be driven by the individual's omni-channel propensity and the associated value and margin for an action.

1. In the Pega Customer Decision Hub portal, click **Intelligence Strategies** .
2. Click **Create Guide me through it** .
3. Enter the strategy name, for example, Priority Based Targeting.
4. In the Purpose section, click **Configure**.
5. Select the **Paid optimization** objective.
6. Click **Apply**.
7. In the Optimization section, select the default value of **Basic optimization**. **Result:** The Strategy Builder automatically generates an optimization strategy with predefined settings for the audience, campaign and ad group or ad set names.

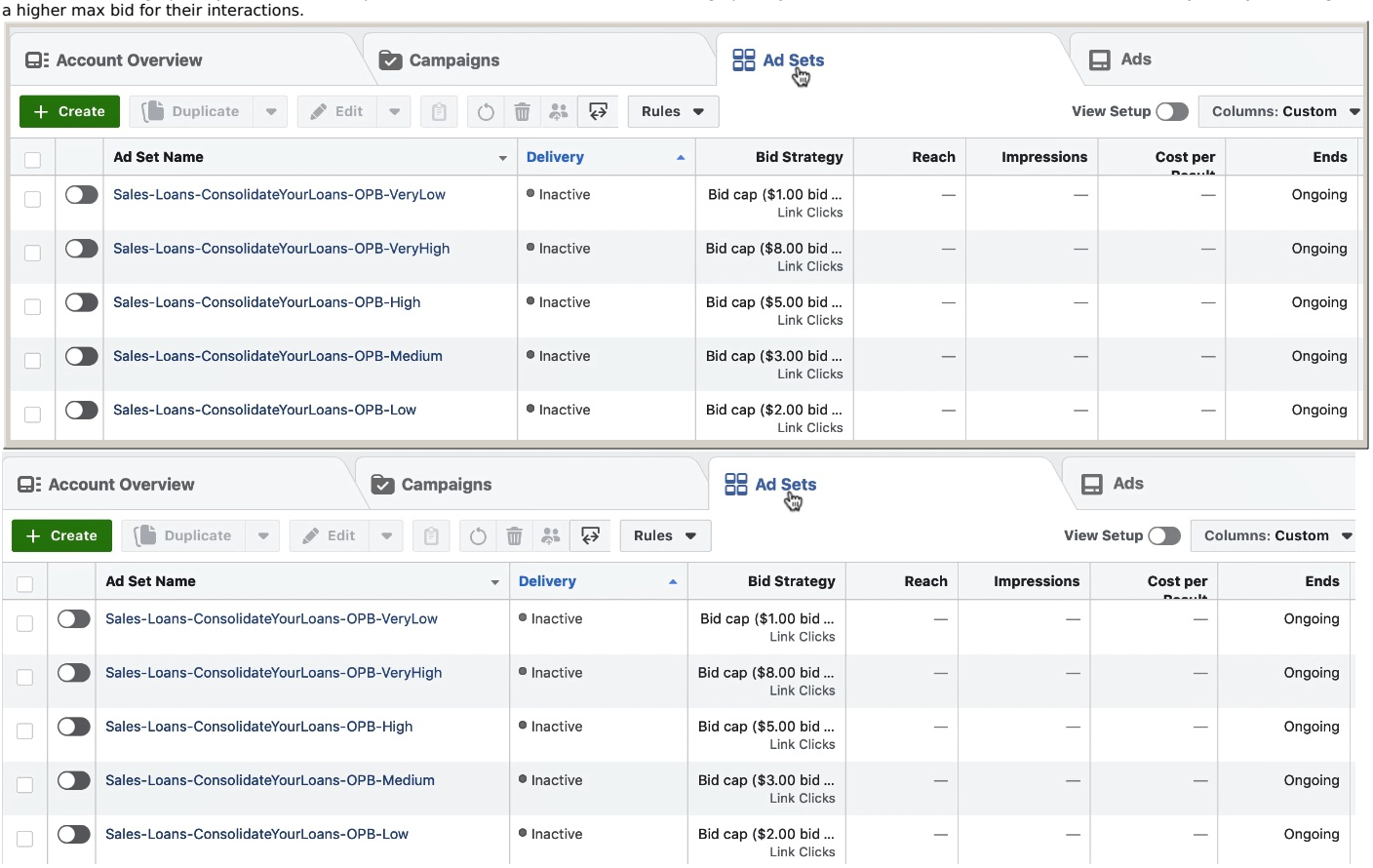


Bidding
Offer priority based audiences
Bidding result: Offer priority bidding
Artifact generation
Audience naming convention
Priority
Score
Paid audience
Very High
1.00 - 0.81
Issue-Group-Offer-VeryHigh
High
0.80 - 0.61
Issue-Group-Offer-High
Medium
0.60 - 0.41
Issue-Group-Offer-Medium
Low
0.40 - 0.21
Issue-Group-Offer-Low
Very Low
0.20 - 0.00
Issue-Group-Offer-VeryLow
Campaign and Ad Group/Ad Set naming convention
Campaign: 'Issue-Group-Offer'
Ad Group/Ad Set: 'Issue-Group-Offer-Priority'
If campaign generation is enabled at the destinations, Campaign and Ad Group/ Ad Set will be generated on your destinations using naming convention above.

   The prioritization is based on a **Priority lookup** decision table, which calculates the priority value based on each individual's score. Individuals are then assigned to audiences with matching priority, and ineligible individuals are filtered out.

8. To edit the default name for each audience, click **Edit naming**.
9. Select the strategy in the Next-Best-Action Designer.
   a. In the Pega Customer Decision Hub portal, click **Next-Best-Action Designer Channels** .
   b. In the Media strategy section, click **Configure**.
   c. Select the strategy which you created.
   d. In the Paid destinations section, for each paid destination, select the type of bidding results.
      For a strategy which uses basic optimization, only the Offer priority bidding result is available for all destinations. For information about specifying different bidding results per destination, see Reduce wasted ad spend based on the predicted value of a click.

Result: When the selected strategy runs, the audiences it creates in the destination platform indicate the priority, which can then be aligned with the willingness to pay for associated ads. High-priority audiences for a specific action indicate that the action has high priority for the individuals in that audience, and therefore you may be willing to set a higher max bid for their interactions.



Account Overview
Campaigns
Ad Sets
Ads
Create
Duplicate
Edit
Rules
View Setup
Columns: Custom
Ad Set Name
Delivery
Bid Strategy
Reach
Impressions
Cost per
Ends
Sales-Loans-ConsolidateYourLoans-OPB-VeryLow
Sales-Loans-ConsolidateYourLoans-OPB-VeryHigh
Sales-Loans-ConsolidateYourLoans-OPB-High
Sales-Loans-ConsolidateYourLoans-OPB-Medium
Sales-Loans-ConsolidateYourLoans-OPB-Low
Inactive
Bid cap ($1.00 bid ...
Bid cap ($8.00 bid ...
Bid cap ($5.00 bid ...
Bid cap ($3.00 bid ...
Bid cap ($2.00 bid ...
Link Clicks
—
Ongoing

For Troy Murphy, executing the media strategy communicates to Facebook that you are willing to pay more for his clicks on a loan consolidation offer than for the cashback credit card offer. When he accesses his Facebook feed, he is now more likely to see the loan consolidation offer instead.

Reducing wasted ad spend based on destination-specific interaction results

Individuals interact with your actions, for example, by clicking a link in the ad, accepting or rejecting the offer, or ignoring the action altogether. The individual's preferences captured in the results of these interactions can be used to increase the accuracy of adaptive models and leverage destination-specific responses.

Pega Customer Decision Hub

The CapturePaidResponse service is used by media strategies to drive downstream learning based on paid interactions such as paid clicks. Interactions captured from paid channels are recorded and used to improve the accuracy of adaptive models and drive decisioning. This intelligent learning allows a media strategy to make destination-specific audience decisions for the same action.

For example, if Troy Murphy is more likely to click an ad on Facebook than on LinkedIn, the media strategy can put that individual into a higher-priority audience on Facebook and a lower-priority audience on LinkedIn. In this way, you can bid more for the individual on the destination where they are more likely to interact with your message.

The service supports the POST HTTP method. The website to which an individual is redirected when they click the URL must contain JavaScript that fetches parameters from the click URL and passes them to the CapturePaidResponse service.

For the request, the service expects a JSON Object with the following attributes:

- *CustomerID* - Used to identify individuals based on information captured during their interactions with the destination. For individuals yet to be identified, a unique id is stored in a browser cookie. Once an individual has been identified as a customer, their encrypted customer id is stored. For more information, see Identifying customers with identity matching.
- *ExternalAudienceId* - For ad platforms other than Google Ads and Facebook Ads, this parameter is instead used to associate an ad to a specific audience and find the action corresponding to the ad. Configure the click URL of the ad to include the *ExternalAudienceId*. You can see the *ExternalAudienceId* for each audience in the AUDIENCE KEY field on the Paid Media Audiences landing page.
- *AccountId* - The ID of the paid destination.
- *ReferrerUrl* - The HTTP referer of the requested website.
- *Source* - The site which sent the request.
- *Utm_medium* - The type of link which was used.
- **Optional:** *AdSetID* - For Facebook Ads and Google Ads, include the *AdSetID* parameter in the click URL. For Facebook Ads, the *AdSetID* parameter corresponds to *adset_id*; for more information, see About URL parameters. For Google Ads, the parameter corresponds to *{adgroupid}*; for more information, see About ValueTrack parameters. The parameters identify the ad set for which the ad is created, and identify the action corresponding to the ad.
- **Optional:** *DestinationType* - For Facebook Ads and Google Ads, this parameter identifies the destination type.

An example POST request is below:

```
{ "CustomerID":"TestID31265",
"ExternalAudienceId":"90a6",
"AccountId" : "facebook_test_ac",
"ReferrerUrl": "facebook.com",
"Utm_medium":"Banner Ad" ,
"AdSetID":"TestAd85372",
"DestinationType":Facebook
```

In order to update which ads an individual is targeted with after they have accepted an offer, real-time container execution for paid media is triggered asynchronously if the request contains the container name and the response corresponds to a final stage, such as Accepted or Rejected.

Reduce wasted ad spend based on the predicted value of a click

Use the interactions recorded by the PaidCaptureResponse service to predict whether clicking on an ad would increase an individual's propensity to accept an offer, and therefore to align the willingness to pay with the predicted value of a click.

Pega Customer Decision Hub

Cost per click (CPC) is a billing method for ad campaigns where payment is charged per click on an ad. Paid Media Manager provides the option to optimize your CPC ad spend by assessing the value of a click. That is, you can use the CPC bidding model to generate strategies which calculate how much a click on an ad would increase the individual's propensity, and therefore how much value is added by that individual's click.

The priority for the CPC bidding model is determined by the out-of-the-box adaptive model *CPCIncrementalLift*. The model determines the individual's propensity to accept an offer if they clicked on that offer by comparing adaptive analytics data from individuals who clicked an ad for that offer to the omni-channel propensity determined by the full set of adaptive analytics. The difference in propensity is called the incremental lift. The CPC bidding model uses that lift to align the willingness to pay with the value which the ad spend generates per individual.

The *CPCIncrementalLift* model is based on data captured by the CapturePaidResponse service. You must implement CapturePaidResponse in order for the model to work correctly. For more information, see Reducing wasted ad spend based on destination-specific interaction results.

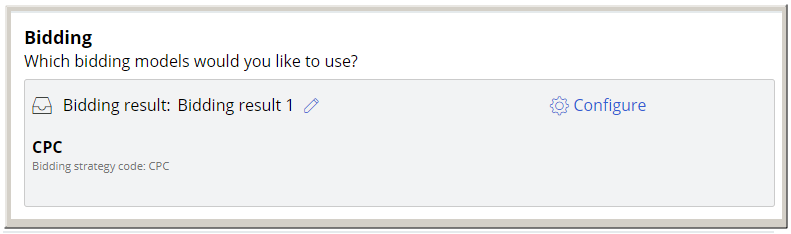
- Creating a CPC-based strategy with the Strategy Builder

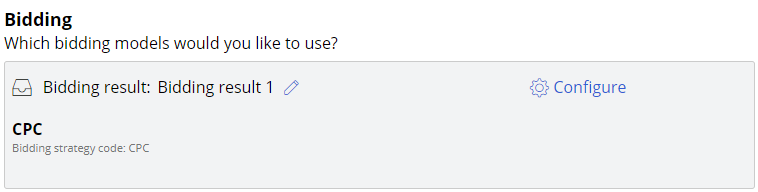
Creating a CPC-based strategy with the Strategy Builder

To create a media strategy which prioritizes paying for clicks based on how much they increase the individual's propensity to accept the offer, use the Strategy Builder.

Pega Customer Decision Hub

1. In the Pega Customer Decision Hub portal, click **Intelligence Strategies** .
2. Click **Create Guide me through it** .
3. Enter the strategy name, for example, CPC Based Targeting.
4. In the Purpose section, click **Configure**.
5. Select the **Paid optimization** objective.
6. Click **Apply**.
7. In the Optimization section, select **Destination specific optimization**.
8. Select **CPC**.





9. **Optional:** Click **Add bidding result** to specify another bidding model. This can be used to define different bidding models for different paid destinations.
10. To edit the default name for each audience, click **Edit naming**.
11. Select the strategy in the Next-Best-Action Designer.
    a. In the Pega Customer Decision Hub portal, click **Next-Best-Action Designer Channels** .
    b. In the Media strategy section, click **Configure**.
    c. Select the strategy which you created.
    d. In the Paid destinations section, for each paid destination, select the type of bidding results.

Result: When the selected strategy runs, the audiences it creates in the destination platform indicate how much the click from an individual in that audience would increase the individual's priority to accept the offer. High-priority audiences have higher expected business value, and therefore you may be willing to set a higher max bid for their interactions.

Ad Set Name	Delivery	Bid Strategy	Reach	Impressions
Sales-Cards-PlatinumCreditCard-CPC-VeryLow	Inactive	Bid cap ($1.00 bid control) Link Clicks	—	—
Sales-Cards-PlatinumCreditCard-CPC-Medium	Inactive	Bid cap ($3.00 bid control) Link Clicks	—	—
Sales-Cards-PlatinumCreditCard-CPC-Low	Inactive	Bid cap ($2.00 bid control) Link Clicks	—	—
Sales-Cards-PlatinumCreditCard-CPC-High	Inactive	Bid cap ($5.00 bid control) Link Clicks	—	—
Sales-Cards-PlatinumCreditCard-CPC-VeryHigh	Inactive	Bid cap ($8.00 bid control) Link Clicks	—	—

Ad Set Name	Delivery	Bid Strategy	Reach	Impressions
Sales-Cards-PlatinumCreditCard-CPC-VeryLow	Inactive	Bid cap ($1.00 bid control) Link Clicks	—	—
Sales-Cards-PlatinumCreditCard-CPC-Medium	Inactive	Bid cap ($3.00 bid control) Link Clicks	—	—
Sales-Cards-PlatinumCreditCard-CPC-Low	Inactive	Bid cap ($2.00 bid control) Link Clicks	—	—
Sales-Cards-PlatinumCreditCard-CPC-High	Inactive	Bid cap ($5.00 bid control) Link Clicks	—	—
Sales-Cards-PlatinumCreditCard-CPC-VeryHigh	Inactive	Bid cap ($8.00 bid control) Link Clicks	—	—

For Troy Murphy, executing the media strategy communicates to Facebook that if he clicks an ad for the Platinum credit card, he has a very high priority to accept that offer. When he accesses his Facebook feed, he is now more likely to see ads for the Platinum credit card.

Prospecting with Next-Best-Action paid audiences

Paid Media Manager enables organizations to leverage their proprietary individual level data, decisioning, intelligence for prospecting by utilizing the lookalike capabilities of media platforms. You can also prospect for new customers using lookalikes of Next-Best-Action audiences.

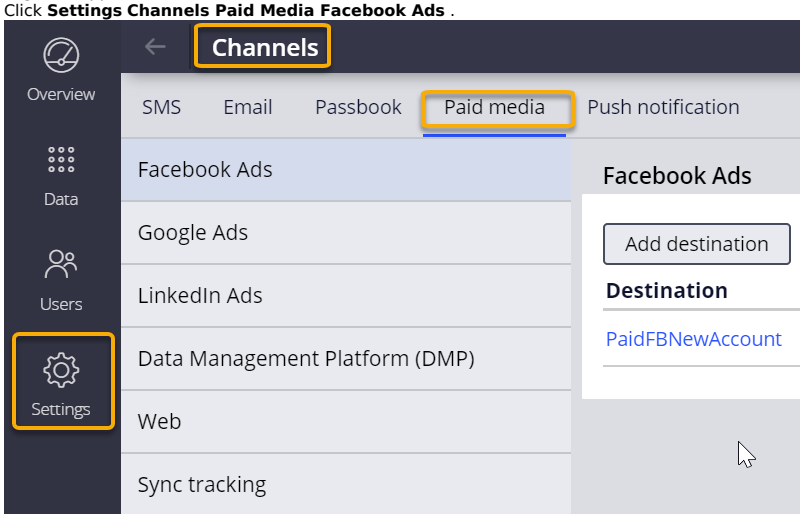
Pega Customer Decision Hub
Tip: Lookalike audiences, also known as similar audiences in Google Ads, are audiences whom you want to target because they resemble individuals whom you already target in some way, and may also be interested in your product or service.You can generate lookalike/similar audiences for Facebook Ads and Google Ads destinations. For destination-specific instructions, refer to the following sections:

- Configuring lookalike audiences in Facebook Ads
- Generating similar audiences in Google Ads
- Prospecting with Facebook lead ads

For Facebook Ads, lookalike audiences are created from a selected percentage of the population of a specified country. To configure them, first specify the lookalike audience country and ratio, and then enable the audience in the destination settings on the **Channels** tab of Next-Best-Action Designer.

Tip: For more information about Facebook lookalike audiences, see About Lookalike Audiences.

1. Log in to App Studio.
2. Click **Settings Channels Paid Media Facebook Ads** .



3. Edit or create a destination.
4. Configure the following settings in the **Facebook Ads assets generation** section:
    - **Lookalike audience country** - Enter the country of residence for audience members whom you want to target. Facebook Ads lookalike audiences must be specific to a single country.
    - **Lookalike audience ratio** - Enter the percentage of the population of the lookalike audience country whom you want to target. For example, a lookalike audience ratio of 0.01 targets 1% of the population of the selected country. Larger audience ratios target more people, but the targeted audience members may have less resemblance to individuals whom you already target.
5. Click **Apply**.
6. In the Pega Customer Decision Hub portal, click **Next-Best-Action Designer Channels** .
7. In the **Paid media Paid destination** section, edit or create a Facebook Ads destination.
8. In the **Destination settings** window, select **Lookalike audience**, and then click **Submit**.

For Google Ads, similar audiences are generated automatically for all destinations where you enabled automatic campaign generation. The generation may take up to 48 hours. After an audience is generated, Pega Customer Decision Hub generates the associated ad group within the Next-Best-Action-driven campaign ad group structure.

What to do next: For more information about Google similar audiences, see About similar audiences on the Display Network.

Facebook lead ads provides a mechanism to drive individuals from an anonymous to a known state by running lead generation campaigns on Facebook and Instagram. They include a contact form for individuals that allows them to get in touch with a business. Leads generated with Facebook Lead Ads are captured directly in Pega Customer Decision Hub. For more information about Facebook Lead Ads, see Lead ads.

To start using Facebook lead ads to collect contact information from prospects:

1. Ensure that your application can both send and receive data from Facebook.
2. Obtain a Facebook Ads access token with the following permissions:
    - *manage_pages*
    - *leads_retrieval*
3. Request the *leads_retrieval* and *manage_pages* permissions for Facebook Ads Lead Retrieval API and submit your app to App Review.
   These permissions are necessary to retrieve lead information once a lead ID is received from Webhooks.
4. In the Facebook Ads destination configuration, enter the access token and select **Enable Lead Generation**. **Result:** Auto-generated webhook URL and token appear.
5. In Facebook Ads, configure the webhook URL and token, and subscribe to lead generation.
   For more information, see Lead Ads.
6. **Optional:** If you changed the field names in the Facebook lead form, edit the data transform **PegaMKT-Data-PaidMedia-LeadAdResponse.FacebookFieldMapping**.
7. **Optional:** Modify the default activity that saves captured leads to better integrate it with your lead handling process. The name of the activity is **PegaMKT-Data-PaidMedia-LeadAdResponse.ProcessProspect**.

What to do next: Captured leads are saved in the database table **mkt_paidmedia_leads** in the Pega Customer Decision Hub database schema. Ensure that expired prospects are removed from the table to prevent it from growing over time.

Reference

This section contains information for administrators.

Pega Customer Decision Hub

- Monitoring Outbound Runs and Real-time Container Updates
- Monitoring paid synchronization jobs
- Error and Warning Handling
- Paid Media Agents
- Paid Media Manager Configuration Settings
- Paid destination setup
- Paid artifacts landing page

Monitoring Outbound Runs and Real-time Container Updates

Use the Data Flows landing page to monitor the progress of each ongoing outbound run or real-time container update. The Data Flows landing page provides developers and application users with an overview of the current state of each run or update.

Pega Customer Decision Hub

1. Click **Data Data Flows** .
2. To review the ongoing outbound runs list, click the **Batch processing** tab.
3. To review the real-time container updates list, click the **Real-time processing** tab.
4. To see the details, click an item ID.

Monitoring paid synchronization jobs

Use the **Sync Tracking** tab on the **Settings Channels Paid Media** landing page to monitor the status of paid synchronization jobs across segments, real-time, and scheduled runs. The Sync Tracking tab is used as a debugging tool for the Paid Media functionality.

Pega Customer Decision Hub

Understanding synchronization jobs

Synchronization jobs represent end-to-end processing of paid media runs. Each sync job consists of the following stages:

- Identifying the payload, that is, hashed PII information about the customer
- Identifying the audiences to which the customer already belongs, as well as the audiences from which the customer should be removed
- Writing to staging tables
- Creating queue items
- Creating and updating audiences in the ad networks
- Adding and removing customers from the ad networks.

Synchronization jobs are created when a paid run is executed by any of the following methods:

- When a real-time container or real-time event gets a request for the Paid channel
- When a Paid outbound run is triggered by the Next-Best-Action Designer outbound schedule
- When historical conversion push is triggered
- When *CapturePaidResponse* REST service is invoked and Offline conversion is enabled at Primary paid configuration in Next-Best-Action Designer
- When a user runs a segment that has been enabled for paid synchronization

Understanding the Sync Tracking tab layout

The following information is displayed for each synchronization job:

Status
: The status of the synchronization job. Possible statuses include Pending, Awaiting response, Running, Completed, Withdrawn, and Failed. If the synchronization job completed with errors, expand the row to view more information. For information about troubleshooting failed sync jobs, see Troubleshooting Paid Media Manager.**Note:** The *Awaiting response* status has been introduced to track the asynchronous Google Ads API.

Destination type
: The type of paid destination configured. The following destination types are available:

  - Facebook Ads
  - Google Ads
  - LinkedIn Ads
  - DMP
  - Web

Destination
: The name of the paid destination of the synchronization job.

Start
: The time when the synchronization job started.

End
: The time when the synchronization job completed or failed.

Sync type
: Indicates whether the synchronization job was triggered by Next-Best-Action, segment update, background processing by an agent that updates the customers' PII, or offline conversion synchronization (which also automatically creates associated conversion audience sync jobs). To filter the job table by a specific synchronization type, use the drop-down list above the table.

Triggered by
: Depending on the type of the synchronization job, this column shows the name of the Next-Best-Action hierarchy level (that is, the issue and group where the outbound run was triggered, or the issue and group of the actions initiated by a real-time run), customer segment, or agent which triggered the synchronization. For offline conversions and conversion audiences, this field is left blank.

Action
: For failed synchronization jobs, this column displays a **Retry** link. If needed, you can use this link to restart a synchronization job that failed.

Restarting a failed synchronization job

The **Retry** link is visible only in the following cases:

- The synchronization job must fail in the synchronization layer. That is, the error must have occurred while interacting with the paid destination, for example, because the access token was invalid, or because the paid destination was unavailable during the synchronization attempt.
- For all synchronization types apart from offline conversion jobs, the job must be the latest failed job of a given type, issue, and group. For example, if an error occurs during two segment synchronization jobs, the **Retry** link will only be visible for the latest failed job. Similarly, if an outbound run and a real-time container synchronization jobs configured at the same level of the business hierarchy both fail, the **Retry** link will only be available for the last failed job. This is to prevent customers being needlessly removed from audiences if an old synchronization job is restarted.**Note:** Offline conversion and conversion audience synchronization jobs display the **Retry** link for all failed jobs, because, as part of these jobs, customers are only added and not removed, so there is no risk of customers being needlessly removed from audiences..

When you click **Retry**, the synchronization job status changes from Failed to Running. Paid Media Manager does not restart the complete synchronization process of the failed synchronization job. Instead, only the processing of failed queue items is restarted.

Error and Warning Handling

The Next-Best-Action Designer displays errors and warnings resulting from each scheduled and real-time paid run. Errors and warnings are displayed chronologically for the Next-Best-Action hierarchy level and paid destination on which they occurred.

Pega Customer Decision Hub

If multiple errors are generated for the same destination, they are added to the existing work object. Paid Media Manager maintains a history of errors and warnings, as well as information about the number of records affected.

Errors that occur during Segment-based runs are logged as work objects of the type PegaMKT-Work-PaidSync. The paid sync errors are displayed in the Paid Sync tab of the Segment for which they were recorded.

If an error occurs, confirm that your account in the integrated platform is properly configured and activated, then resolve the error work object before triggering another paid run.

Errors displayed in the Next-Best-Action Designer come directly from the integrated external system. If you cannot resolve the problem yourself, contact the product team for your advertising platform.

To generate Paid Notification work objects, update the following Data transform rules:

- Data-Party-Operator – MarketingAdministrator
- Data-Party-Operator – MarketingAnalyst

For both rules, set the property **pyWorkPartyUri** to the operator ID used by your administrator and analyst.

Paid Media Agents

For information about paid media agents and job schedulers, see Pega Customer Decision Hub agents.

Pega Customer Decision Hub

Paid Media Manager Configuration Settings

Tune the configuration of Paid Audience updates by adjusting the following settings:

Pega Customer Decision Hub

- PaidMediaCustomerEventsForAPeriod event strategy - This event strategy accumulates events for a given individual for a given period of time. The default time is 5 minutes. Adjust this value if you want the updates to happen more or less frequently. Less frequent but larger API calls will raise the memory consumption.
- PaidMediaEventsByVolume event strategy - This event strategy accumulates all the events (that is, Real-time container and Real-time event requests) initiated for a particular time period and group them together to make batch API calls to networks such as Google or Facebook. The default value is set to 1 hour. Adjust this value based on the frequency of events.
- paidmedia/syncService/requestBatchSize dynamic system setting - This setting defines the number of records which Paid Media Manager batched within a Paid Media Sync queue item before it prepares an API call to supported networks such as Google or Facebook. The default number of records is 10000. The more records there are in a queue item, the less API calls there will be. Adjust this value depending on the memory capacity available in the cluster.
- paidmedia/ResponseWaitingTime dynamic system setting - This setting defines the timeout period for actions in paid channels. If an individual does not respond to an action within the specified time period, the system automatically records a negative response. The default timeout period is 604800 seconds (1 week).

Optional: Modify any of the following dynamic system settings for Adobe Audience Manager. For more information, contact your Adobe Audience Manager implementation team.

- paidmedia/adobe/api/authserver/url - This setting specifies the Auth Server URL for Adobe Audience Manager. By default, the URL is set to https://api.demdex.com/oauth/token.
- paidmedia/adobe/api/resourceserver/url - This setting specifies the Resource Server Base URL for Adobe Audience Manager. By default, the URL is set to https://api.demdex.com:443/v1.
- paidmedia/adobe/traitsegment/toplevel/folder/name - This setting defines the name of the top level segment and trait folder created under the parent folder in Adobe Audience Manager. The default name of this folder is Paid Audiences.
- paidmedia/adobe/file/push/maxfilesizeinbytes - The maximum size in bytes of synchronization files sent to Adobe Audience Manager. The default file size in bytes is 1000000000.
- paidmedia/adobe/file/push/retrycount - The number of times Paid Media Manager automatically tries to resend the synchronization file to Adobe Audience Manager in case of any synchronization problems. The default number of tries is 3.

Paid destination setup

App Studio Settings Channels Paid media

The **Channels Paid media** landing page contains the following tabs:

Pega Customer Decision Hub

- Facebook Ads
- Google Ads
- LinkedIn Ads
- Data Management Platform (DMP)
- Web
- Sync Tracking

Using the Facebook Ads tab

The **Facebook Ads** tab allows you to configure one or more paid destinations for Facebook Ads. For more information, see "Configuring Facebook Ads destinations" in the *Pega Customer Decision Hub* Implementation Guide on the Pega Customer Decision Hub product page.

Using the Google Ads tab

The **Google Ads** tab allows you to configure one or more paid destinations for Google Ads. For more information, see "Configuring Google Ads destinations" in the *Pega Customer Decision Hub* Implementation Guide on the Pega Customer Decision Hub product page.

Using the LinkedIn tab

The **LinkedIn Ads** tab allows you to configure one or more paid destinations for LinkedIn Ads. For more information, see "Configuring LinkedIn Ads destinations" in the *Pega Customer Decision Hub* Implementation Guide on the Pega Customer Decision Hub product page.

Using the Data Management Platform (DMP) tab

The **Data Management Platform (DMP)** tab allows you to configure one or more paid destinations for data management platforms (DMPs). You can configure paid destinations for Adobe Audience Manager or a generic DMP. For more information, see "Configuring DMP destinations" in the *Pega Customer Decision Hub* Implementation Guide on the Pega Customer Decision Hub product page.

Using the Web tab

The **Web** tab allows you to configure one or more web-based paid destinations. Legacy web-based ad platforms, such as demand-side platforms (DSPs), use pixels, also known as remarketing tags, to access their cookie-based profiles and target ads accordingly. These pixels are typically called when an individual visits a website, and the ads which they display are based on the individual's current website activity.

Paid Media Manager allows you to include the power of Next-Best-Action decisioning into the pixel calls. You can extend the pixel call with a call to the Paid Media web service to determine ad targeting based on the enterprise decisioning and intelligence embedded in Next-Best-Action. For more information, see "Paid Media web service" in the *Pega Customer Decision Hub* Implementation Guide on the Pega Customer Decision Hub product page.

Using the Sync Tracking tab

The **Sync Tracking** tab allows you to monitor the status of paid synchronization jobs across segments, real-time, and scheduled runs. You can use it to find failed synchronization jobs and restart them without the necessity to restart the whole outbound run.

The following information is displayed for each synchronization job:

Status
: The status of the synchronization job. Possible statuses include Pending, Running, Completed, and Failed. If the synchronization job completed with errors, expand the row to view more information.

Destination
: The name of the paid destination of the synchronization job.

Start
: The time when the synchronization job started.

End
: The time when the synchronization job completed or failed.

Type
: Indicates whether the synchronization job was triggered by Next-Best-Action, segment update, or background procesing by an agent. To filter the job table by a specific synchronization type, use the drop-down list above the table.

Details
: Depending on the type of the synchronization job, this column shows the name of the Next-Best-Action hierarchy level, customer segment, or agent which triggered the synchronization.

Action
: For the latest failed synchronization job, this column shows the **Retry** option. Click the link to restart the synchronization job without restarting the associated scheduled run.

Paid artifacts landing page

Pega Customer Decision Hub portal Audiences Paid Media

The **Audiences Paid Media** landing page lists the paid audiences created after a paid run.

Pega Customer Decision Hub

Use this landing page to:

- Review high-level paid audience details
- Filter the list of audiences by audience type
- Check for trends in audience sizes

Review high-level paid audience details

The following information is displayed for each audience:

- Platform icon
- Audience name
- Audience key
- Issue and group name
- Destination type and destination name
- Number of customers in the audience
- Number of matched customers and match rate for all Google audiences, excluding conversion sets
- Number of matched customers and match rate for Facebook conversion sets

Filter the list of audiences by audience type

You can use the **Audience type** filter to display audiences based on their type. Paid Media Manager creates the following types of audiences in a paid destination:

- **Positive** - These audiences are used to target individuals with ads based on the priority of an ad for an individual as determined by eligibility and relevance rules, as well as AI-driven analysis.**Note:** These audiences also include the conversion audiences for Facebook Ads and Google Ads.
- **Negative** - These audiences represent the inverse of priority-based audiences – that is, individuals that are not eligible and have low priority. Negative audiences are excluded from targeting to eliminate wasted ad spend.
- **Conversion** - These conversion sets represent the individuals who have completed a conversion, that is, accepted an action after viewing an online ad from the marketer.

Check for trends in audience sizes

A spark line graph shows the trends of population and matched count over the last thirty days. To see the audience count for a specific date, hover your mouse over a point on the spark line.

Reporting and monitoring Pega Customer Decision Hub key performance indicators

Monitor and report on key performance indicators using dashboard widgets, out-of-the-box and customized reports.

Pega Customer Decision Hub

Pega Customer Decision Hub provides numerous reporting tools to monitor the performance of actions. These tools include out-of-the-box dashboard widgets, which provide managers and analysts with quick insights into the status of various marketing metrics. For more in-depth reporting, Pega Customer Decision Hub provides several reports which you can further customize to get real-time insights into the performance of your action portfolio. During the implementation phase, make sure to review the available reporting and monitoring tools, and configure them to meet your organizational requirements.

Reporting options in Pega Customer Decision Hub

Use the following tools to monitor and report on your marketing portfolio:

Dashboard
: The dashboard contains a set of widgets, which display the performance of various marketing metrics, such as owned media spend or churn rate. Once configured, the dashboard is a tool for marketing analysts and managers to quickly assess how their business is doing.

Reports
: Reports retrieve and sort information from the Pega Customer Decision Hub database. In addition to reports inherited from the Pega Platform report library, Pega Customer Decision Hub also contains marketing-specific reports, which you can customize.

Adaptive model reports
: Adaptive models predict customer behavior. Use Prediction Studio to monitor how successful the predictions are.

- Customizing the Pega Customer Decision Hub dashboard
- Customizing the Pega Customer Decision Hub reports

Customizing the Pega Customer Decision Hub dashboard

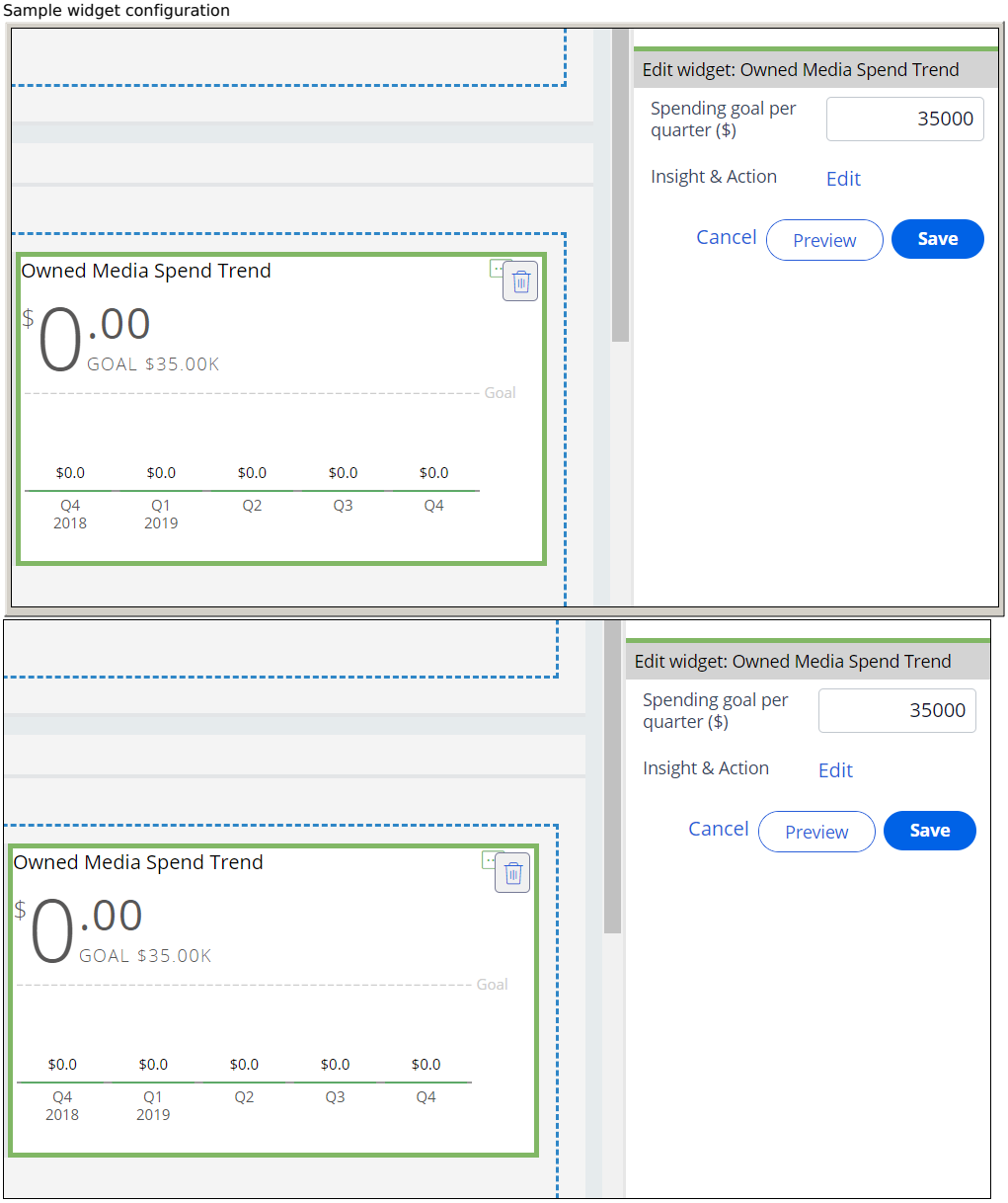
Customize the dashboard to include the most important widgets and configure the desired goals for the metrics that the widgets monitor.

Pega Customer Decision Hub

1. In the Pega Customer Decision Hub portal, click **Dashboard**.
2. Click the **Customize** icon in the top right corner of the dashboard.
3. To add a widget to a slot on the dashboard, click **Add Widget**.
4. Configure the widget.
    a. Hover your mouse over the widget. If there are additional configuration options, the **Click to edit** message appears.

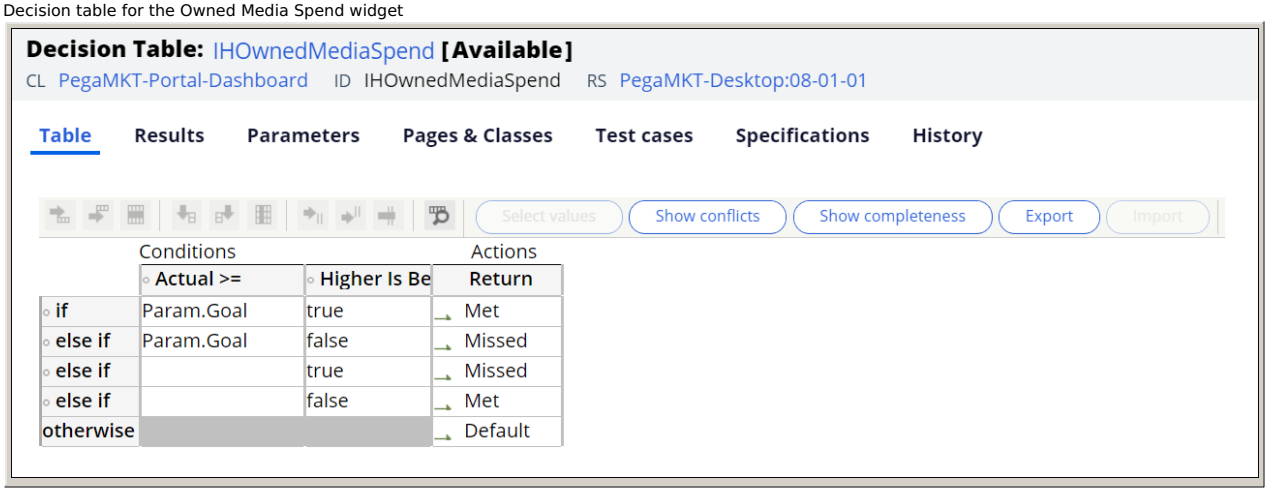
b. Click the widget.
c. In the **Edit dashboard** panel, configure the desired goals for the metrics that the widget monitors.

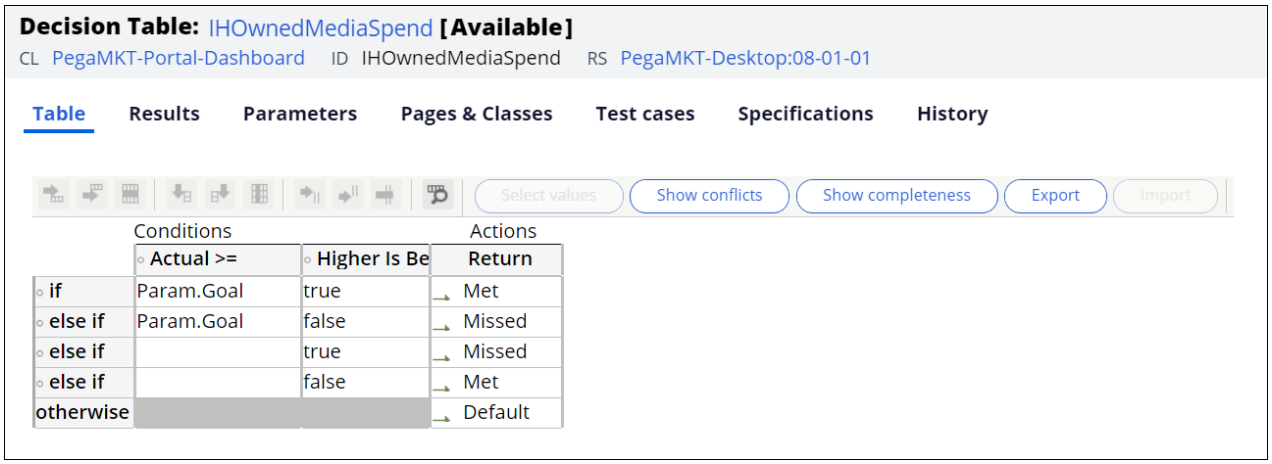
Sample widget configuration



d. **Optional:** To configure the message that indicates whether the goals for the metric are met, in the **Insight & Action** section, click **Edit**, and then configure the underlying decision table, as in the following figure:

Decision table for the Owned Media Spend widget

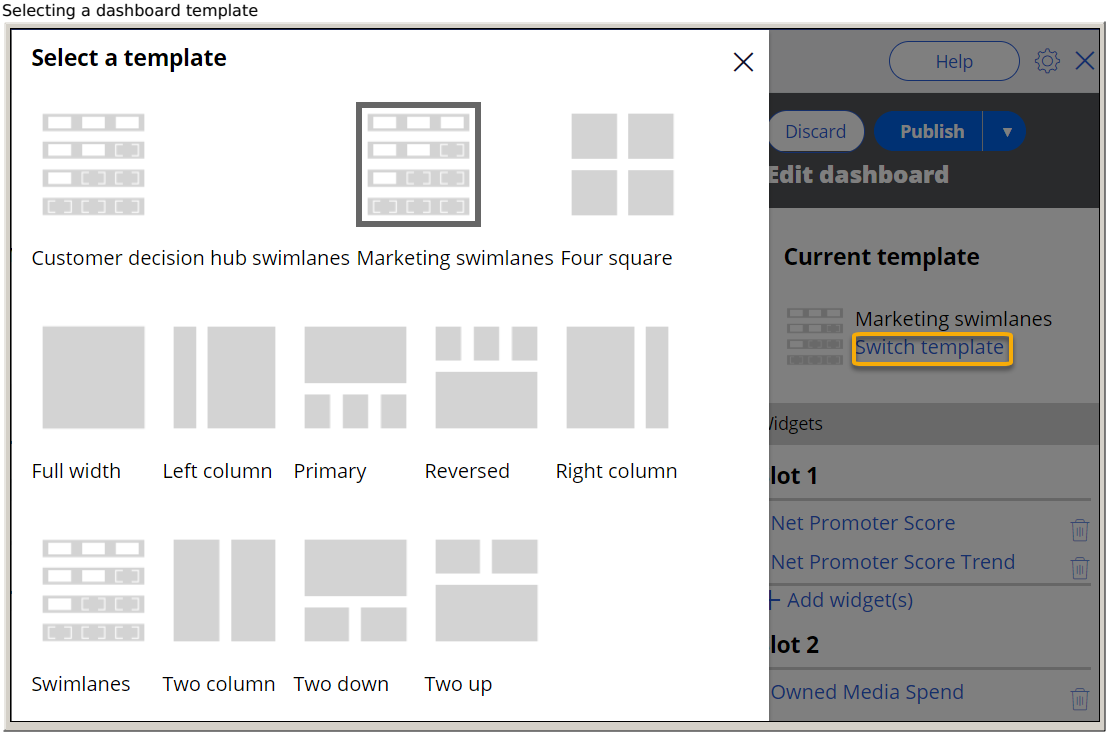


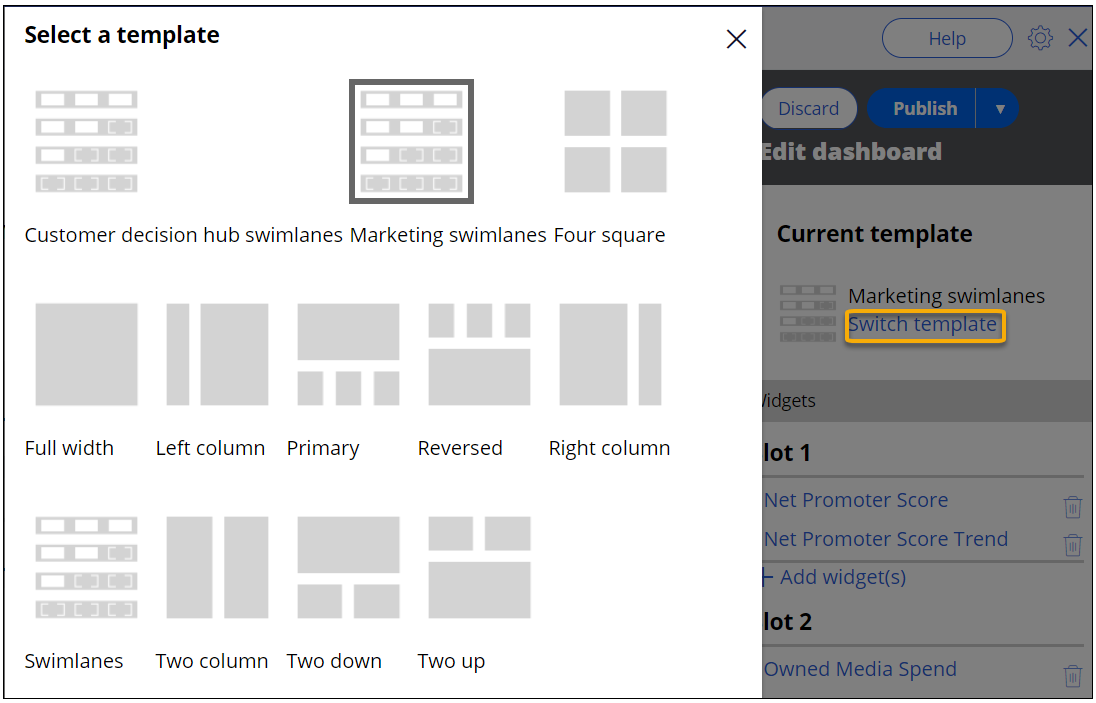


e. Click **Save**.

5. To rearrange the widgets on the dashboard, hover your mouse over a widget, and then drag it into a new position.
Note: You can also change the dashboard template, that is, the general layout of the widgets on the dashboard, as in the following figure. However, bear in mind that the default **Marketing swimlanes** template is optimized for Pega Customer Decision Hub widgets and is recommended in most cases.

Selecting a dashboard template





6. After making the required changes to the dashboard, perform one of the following actions to publish them:
   - To update only your own dashboard, click **Publish**.

     Your changes are not visible to other users.
   - To make the updated dashboard the default for the entire application, for example as part of implementing Pega Customer Decision Hub, click **Publish Publish to default** .**Note:** You can only update the default dashboard configuration if your access group has the PegaMarketing_FW:PortalAdministrator access role.

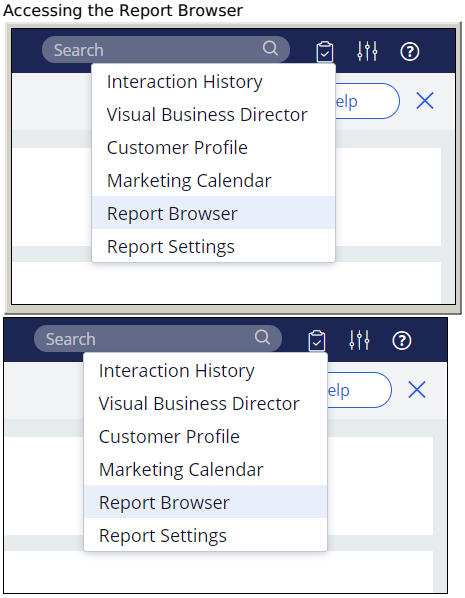
Customizing the Pega Customer Decision Hub reports

Customize the Pega Customer Decision Hub reports to monitor key performance indicators specific to your business.

Pega Customer Decision Hub
Pega Customer Decision Hub contains several marketing-specific reports, grouped together under the Interaction History category in the Report Browser. Review the reports and configure them to suit your business needs.

1. In the Pega Customer Decision Hub portal, click **Reports Report Browser** , as in the following figure:

   Accessing the Report Browser



2. In the **Public categories** list, select **Interaction History**.
3. Review the available reports:

   Acceptance Rate
   : This chart shows the percentage of accepted actions for each group in the business structure of your organization over the last 30 days. Use it to quickly compare the acceptance rate across different business groups.

   Channel Distribution
   : This chart shows the number of initiated actions by channel over the last 30 days. Use it to see which channels have the highest or lowest volume of interactions.

   Interaction History
   : This table provides detailed insights into individual interactions recorded in the Interaction History. Use it to drill down to the details of responses to actions which are offered to customers.

   Proposition Distribution
   : This chart shows the number of initiated actions over the last 30 days. Use it to see which actions are being offered to your customers.

4. Configure the reports to meet your business requirements.

For a list of best-practice configurations, see the following topics:
- Configuring the Proposition Distribution report to reflect portfolio coverage
- Configuring the Interaction History report to show responses from actions offered to customers
- Configuring the Interaction History report to rank accepted actions

- Configuring the Proposition Distribution report to reflect portfolio coverage
- Configuring the Interaction History report to show responses from actions offered to customers
- Configuring the Interaction History report to rank accepted actions
- Accessing Interaction History data for external reports
- Exporting reports
- Capturing negotiation outcomes from Pega Next-Best-Action Advisor

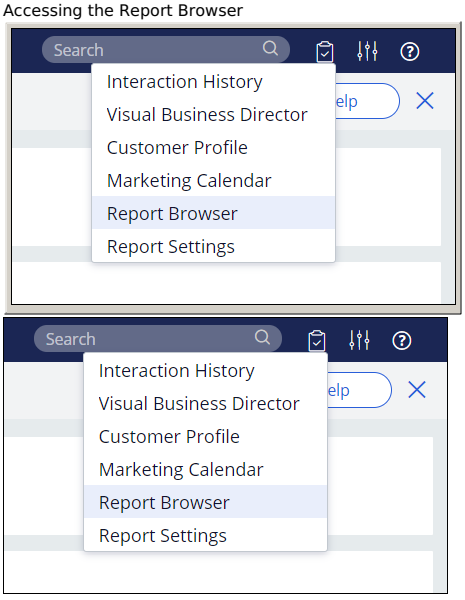
Configuring the Proposition Distribution report to reflect portfolio coverage

Configure the Proposition Distribution reports to show how the business is performing across your portfolio.

Pega Customer Decision Hub
The Proposition Distribution chart shows the portfolio coverage, which is the number of actions that were initiated for your customers over a specific time period. Edit this report to change the reporting period, adjust the grouping of data in the report, or filter out data about specific action groups.

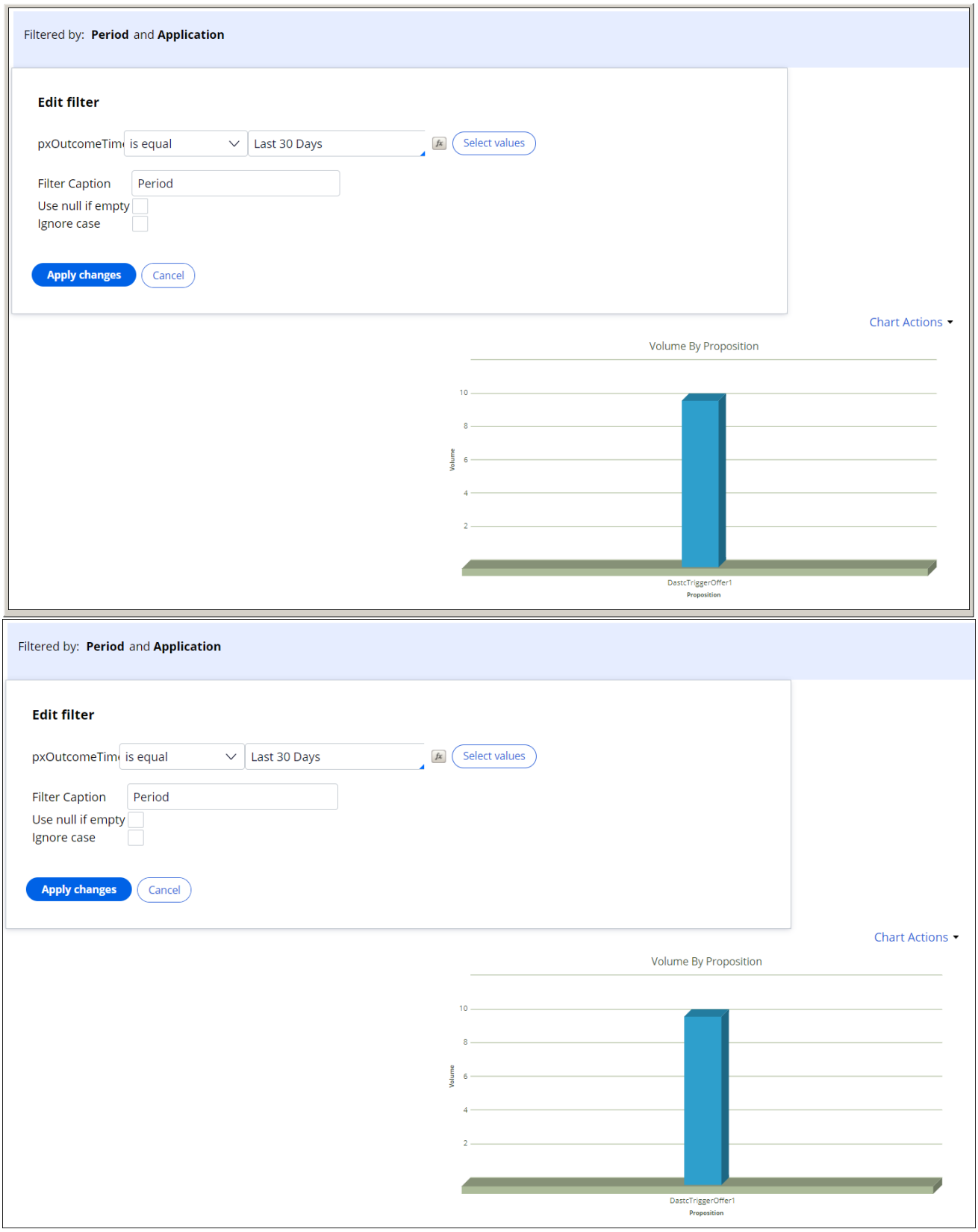
1. In the Pega Customer Decision Hub portal, click **Reports Report Browser** , as in the following figure:

   Accessing the Report Browser

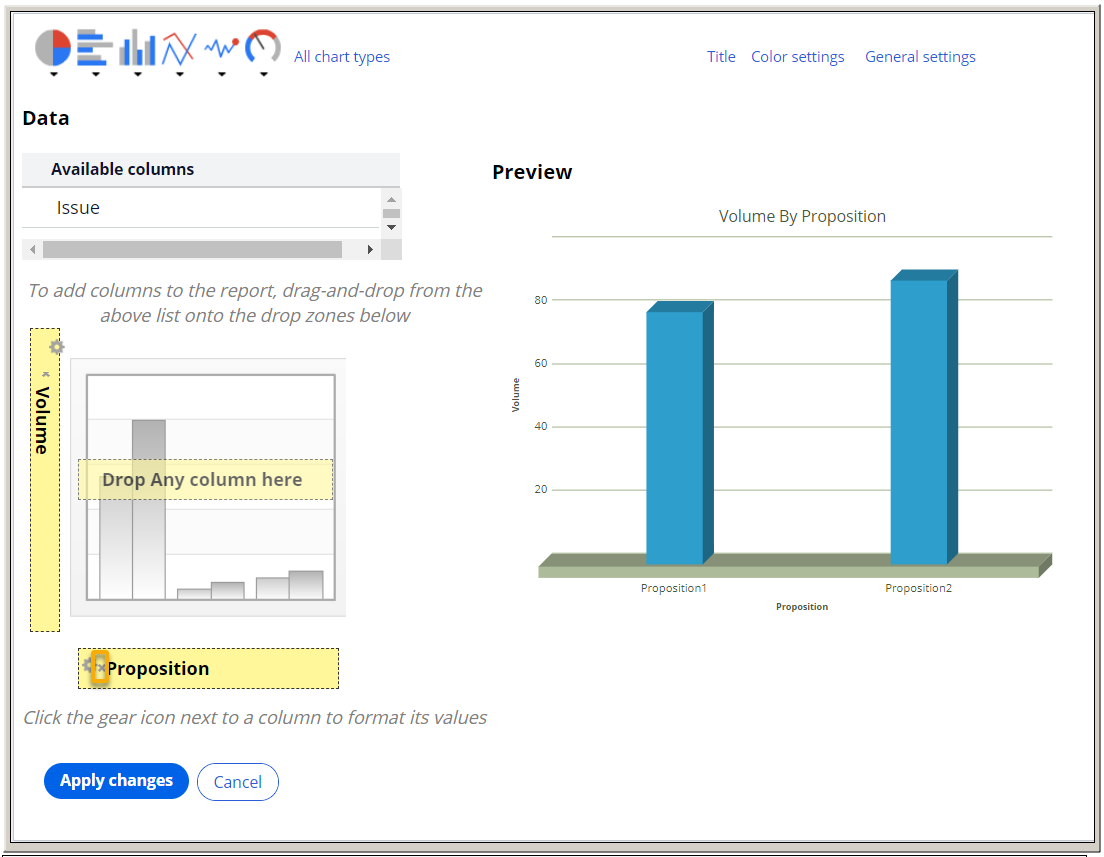
2. In the **Public categories** list, select **Interaction History**.
3. Click the **Proposition Distribution** report.
   The report shows the number of initiated actions as a bar chart. The same information is also available below in the form of a table. By default, the report is filtered to only include results from the past 30 days, as in the following figure:

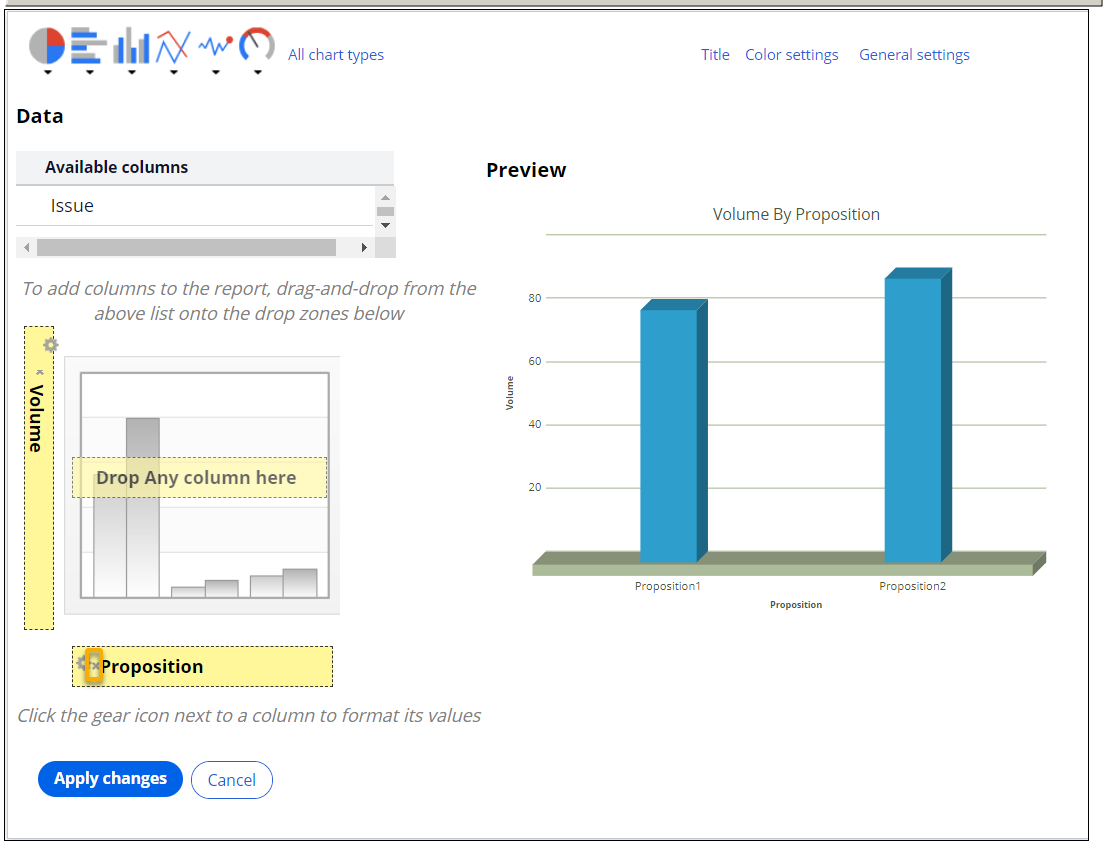
   Default time period for the Proposition Distribution report



4. Click **Edit report**.
5. To change the time frame for the report, for example, from monthly to weekly, in the **Filtered by section**, click **Period**, and then click **Select values**.
6. Edit the report configuration to group the actions by the issue to which the action belongs.
   a. Click **Edit chart**.
   b. Remove the default grouping by action ID by clicking **Delete**, as in the following figure:

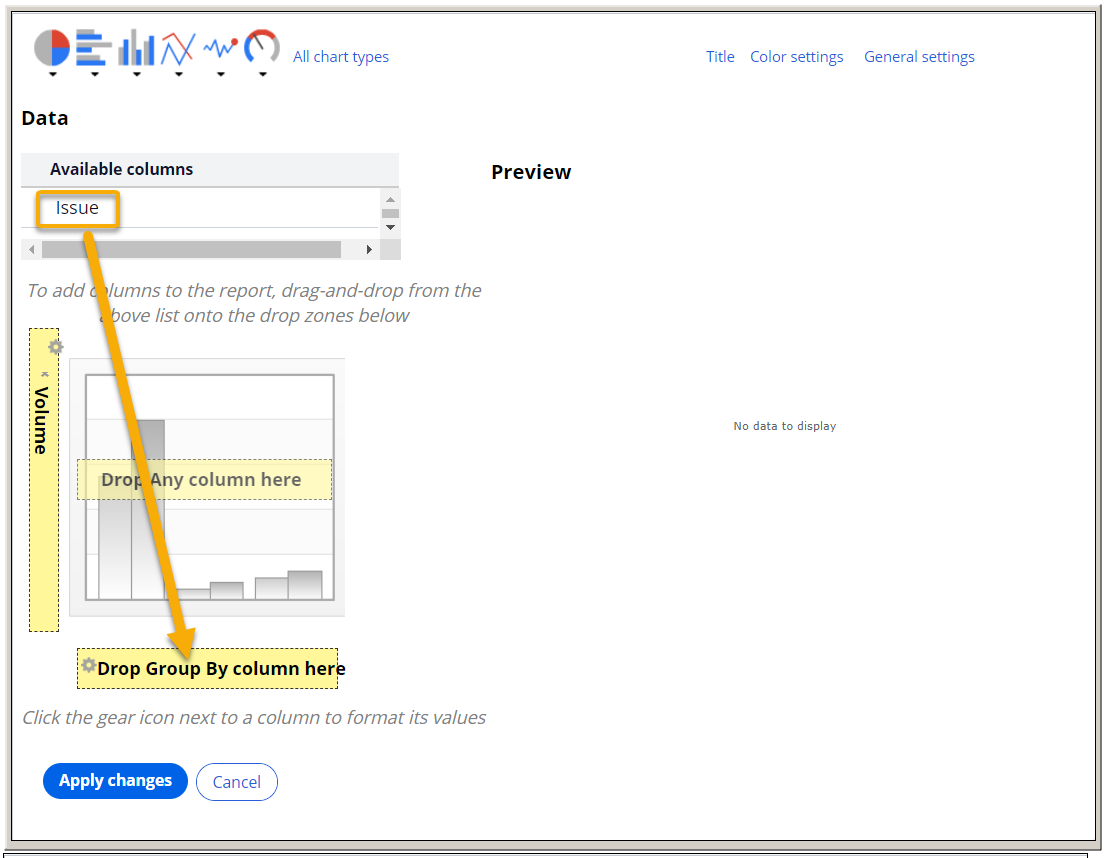
      Removing the default grouping

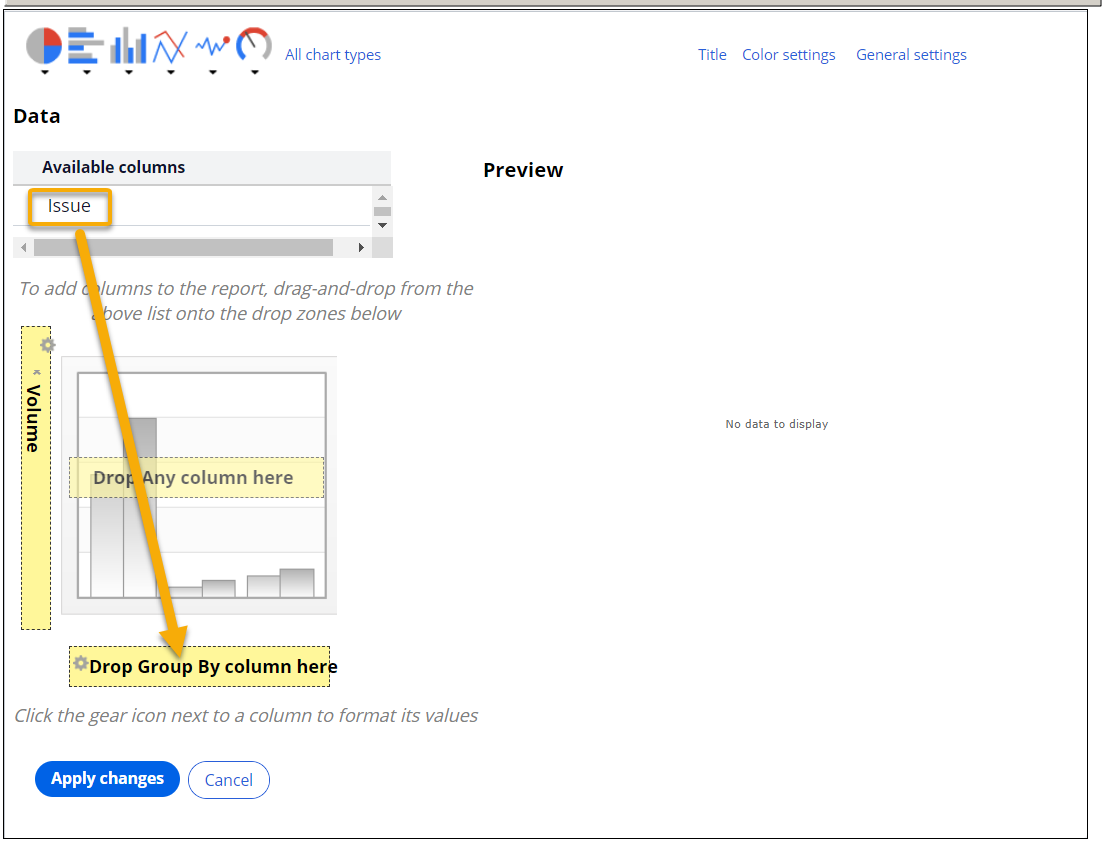




c. Drag and drop the **Issue** column from the **Available columns** table to the **Drop Group By column here** section, as in the following figure:

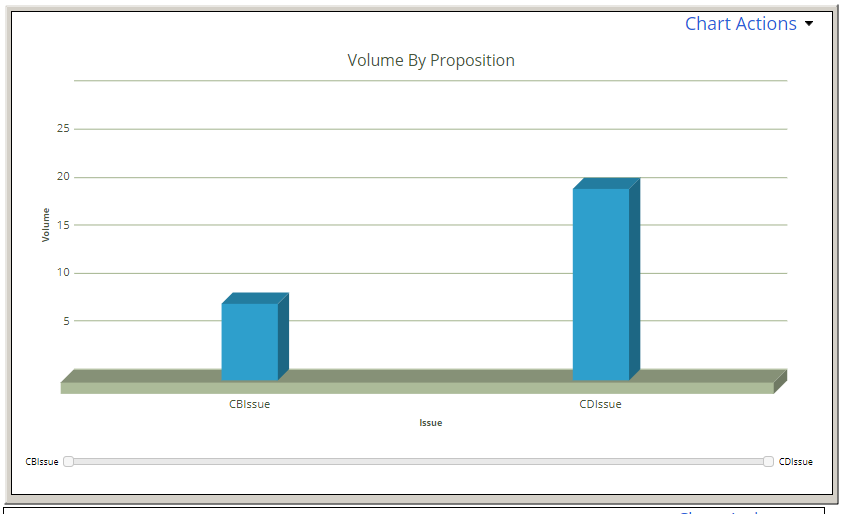
Grouping the report results by issue

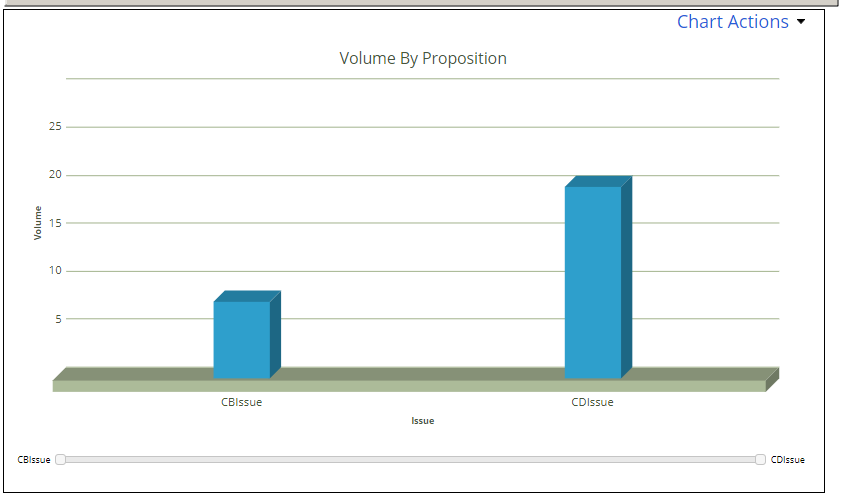




Result: The report now displays action volume grouped and aggregated by issue to which the actions belong, as in the following figure:

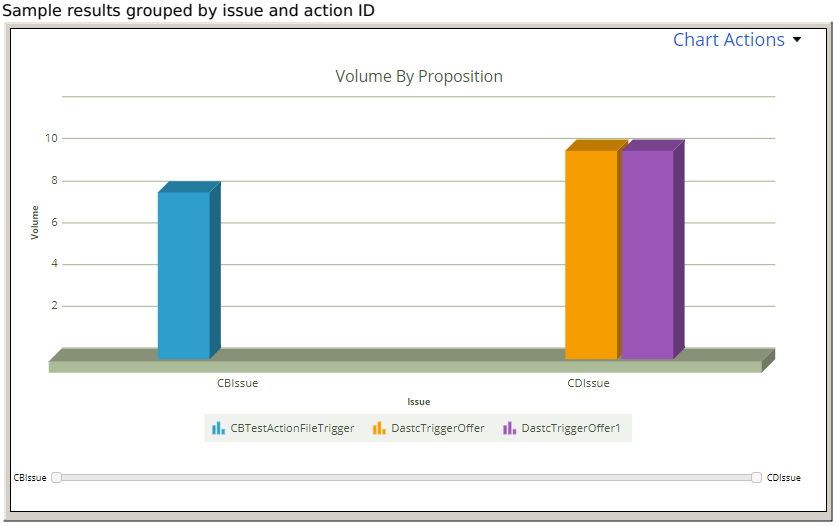
Sample report grouped by issue

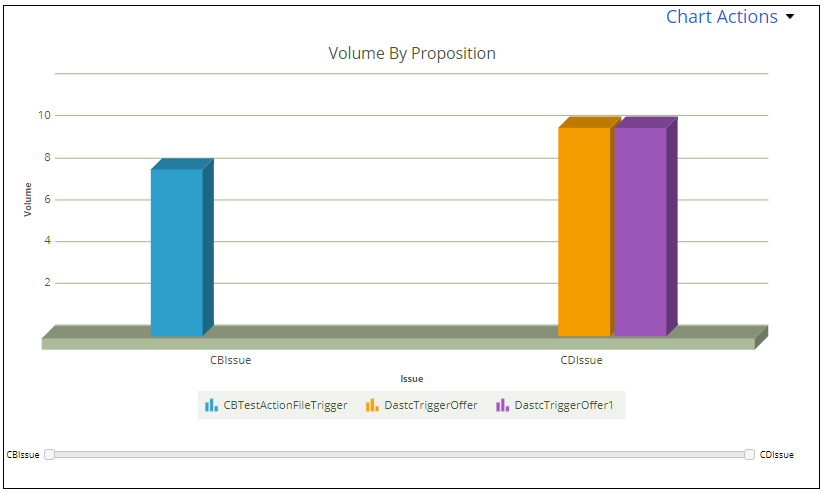




7. To see the results for individual actions within each issue, drag and drop the **Proposition** column from the **Available columns** table to the **Drop Any column here** section. **Result:** The report now displays the results for individual actions, as in the following figure:

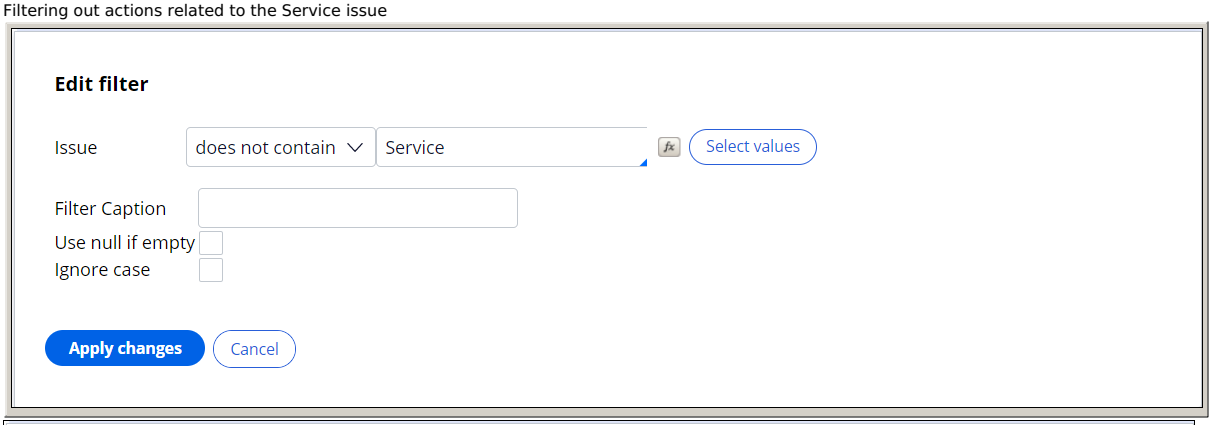
Sample results grouped by issue and action ID

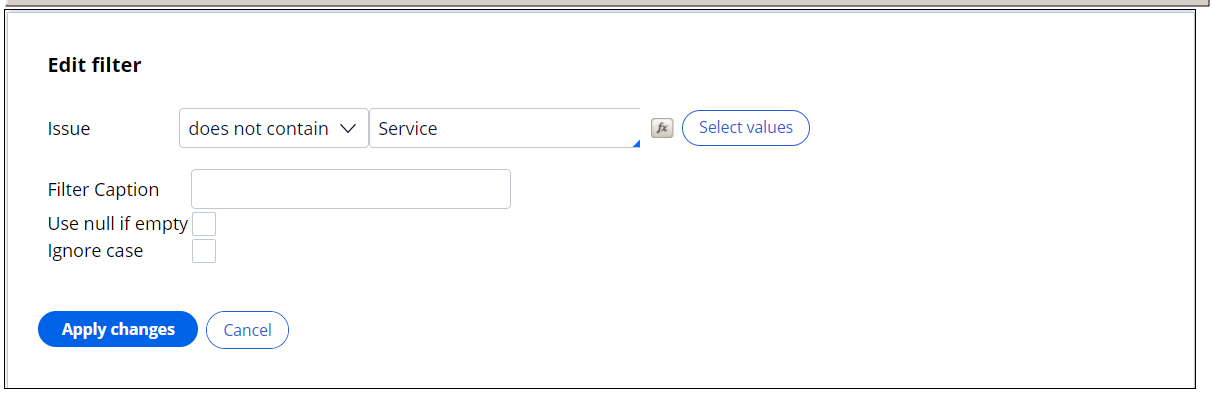




8. Configure the report filters to exclude irrelevant results. **For example:** If your Next-Best-Action business structure includes an issue which represents service messages and regulatory communications, actions related to this issue should be excluded from the results of this report, as they do not represent your marketing portfolio items.
   a. In the **Data Explorer** section of the report, find the item which corresponds to the type of data that you want to filter out. **For example:** To find the Issue item, click **More Action Dimensions Issue** .
   b. Drag and drop the item into the **Filtered by** section of the report.
   c. Edit the filter settings, as in the following figure:

      Filtering out actions related to the Service issue





   d. Click **Apply changes**.
9. Click **Preview**.
10. Click **Actions Save as** and enter a new name for the report. **For example:** Portfolio coverage by action and issue
11. Click **Done editing**. **Result:** The new report is now available in the Report Browser.
12. **Optional:** To automatically export the report results and send them out by email, schedule a recurring task.
    For more information see: Exporting reports

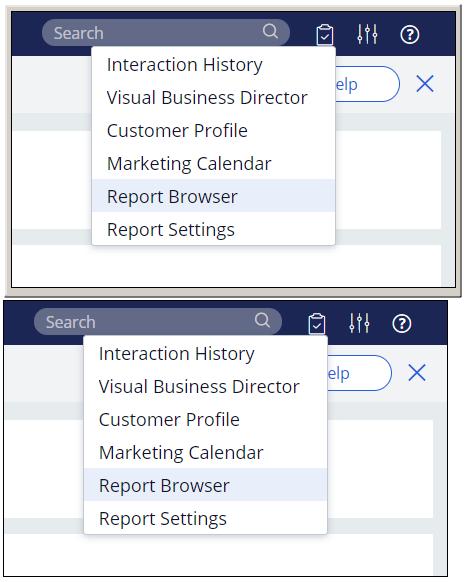
Configuring the Interaction History report to show responses from actions offered to customers

Configure the Interaction History report to get the right level of details from the interactions recorded in the Interaction History.

Pega Customer Decision Hub
The Interaction History report provides detailed insights into individual interactions recorded in the Interaction History. This tutorial describes how to configure this report so that it shows the responses that each action receives, along with other relevant data.

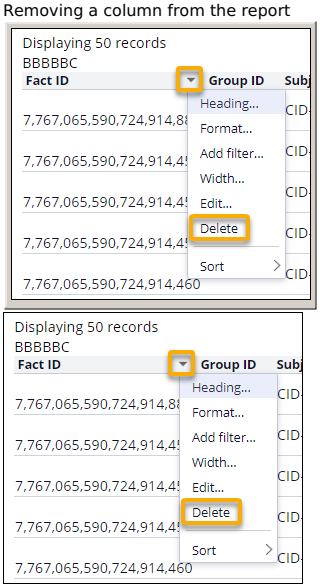
1. In the Pega Customer Decision Hub portal, click **Reports Report Browser** , as in the following figure:

   Accessing the Report Browser



2. In the **Public categories** list, select **Interaction History**.
3. Click the **Interaction History** report.
   The report shows the Interaction History records in the form of a table. By default, the report is filtered to only include results from the past 30 days.
4. Click **Edit report**.
5. To change the time frame for the report, for example, from monthly to weekly, in the **Filtered by section**, click **Period**, and then click **Select values**.
6. To remove the columns which are not relevant for the report, click the **Open menu** icon by the column that you want to remove, and then click **Delete**, as in the following figure:

   Removing a column from the report

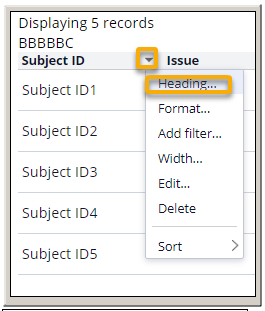


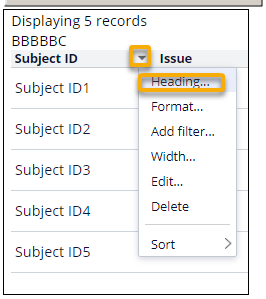
   Tip: Keep the following columns in the report:

   Subject ID
   : The ID of the customer for whom the interaction was recorded.
   Issue
   : The issue to which the action belongs.
   Group
   : The group to which the action belongs.
   Proposition name
   : The name of the action.
   Channel
   : The channel where the interaction occurred.
   Offered on
   : The date and time when the action was offered to the customer.
   Outcome
   : The outcome recorded for the interaction, that is, whether the action was accepted, rejected, or is still pending.
   Rank
   : The action rank. This value is determined by the configuration of the Next-Best-Action strategy.

7. To rename a column header, click the **Open menu** icon by the column that you want to remove, and then click **Heading**, as in the following figure:

   Renaming a column

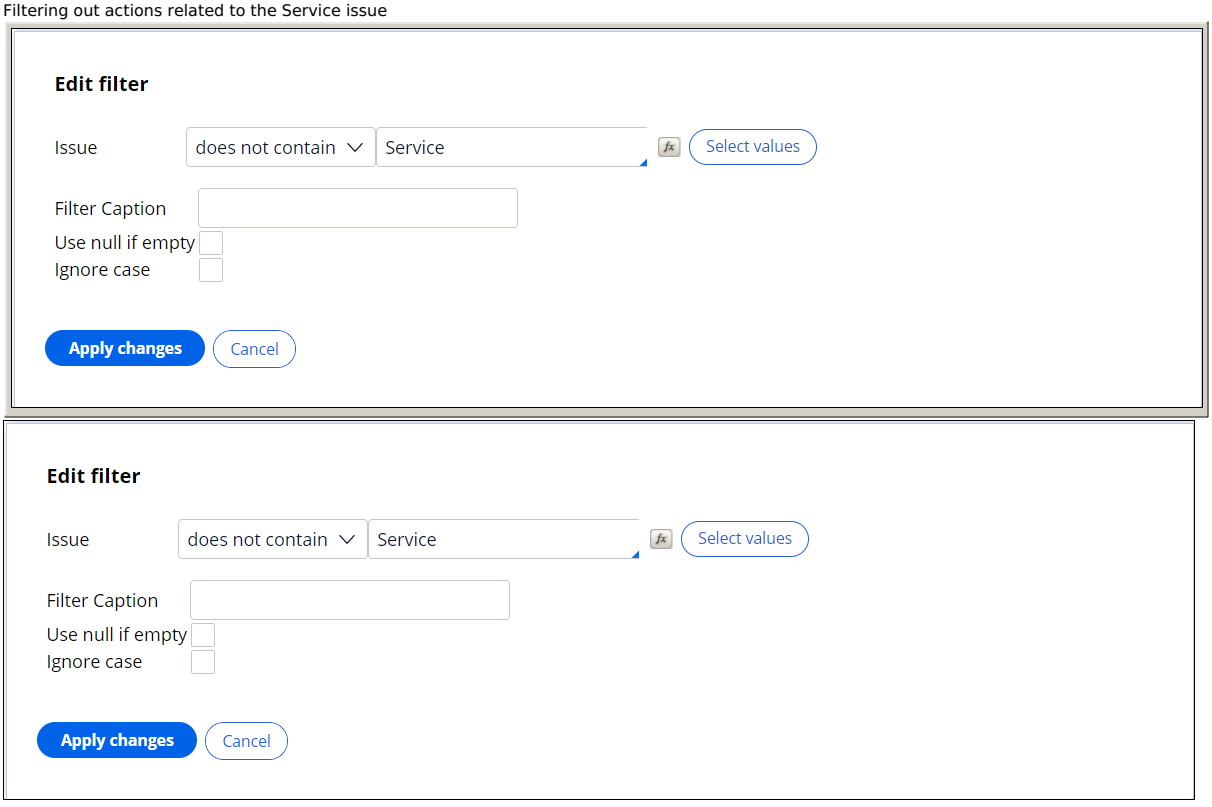




Tip: Rename **Subject ID** to **Customer ID**, and **Proposition** name to **Action name**.

8. Configure the report filters to exclude irrelevant results. **For example:** If your Next-Best-Action business structure includes an issue that represents service messages and regulatory communications, actions related to this issue should be excluded from the results of this report, as they do not represent your marketing portfolio items.
   a. In the **Data Explorer** section of the report, find the item which corresponds to the type of data that you want to filter out. **For example:** To find the Issue item, click **More Action Dimensions Issue** .
   b. Drag and drop the item into the **Filtered by** section of the report.
   c. Edit the filter settings, as in the following figure:

Filtering out actions related to the Service issue



   d. Click **Apply changes**.
9. Click **Preview**.
10. Click **Actions Save as** , and then enter a new name for the report. **For example:** Individual action performance
11. Click **Done editing**. **Result:** The new report is now available in the Report Browser.
12. **Optional:** To automatically export the report results and send them out by email, schedule a recurring task.
    For more information see: Exporting reports

Configuring the Interaction History report to rank accepted actions

Configure the Interaction History report to show only positive customer responses to see the success rate across your portfolio.

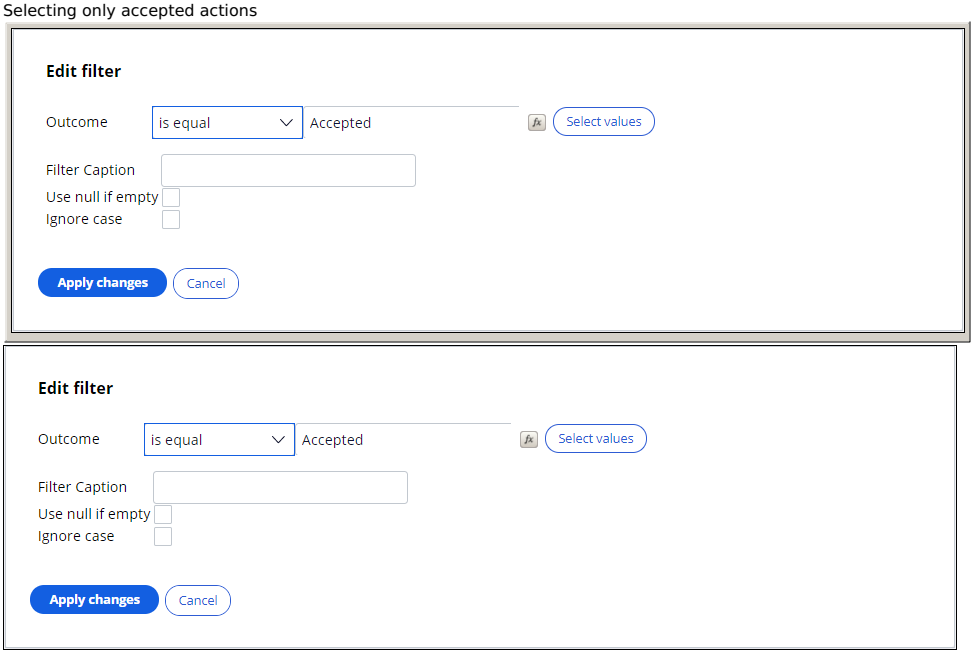
The Interaction History report provides detailed insights into individual interactions recorded in the Interaction History. This tutorial describes how to configure this report so that it shows only actions with a positive response from the customer, ranked according to configuration of the Next-Best-Action strategy.

Pega Customer Decision Hub
Before you begin: Create the individual action performance report. For more information, see Configuring the Interaction History report to show responses from actions offered to customers.

1. In the Pega Customer Decision Hub portal, edit the report which you created.
2. To order the report by the rank of each action, click the **Open menu** icon by the **Rank** column, and then click **Sort Highest to lowest** .

3. Configure the report filters to only include the data for actions which were accepted by customers.
   a. Click the **Open menu** icon by the **Outcome** column, and then click **Add filter**.
   b. Enter the name of the outcome which corresponds to an action being accepted by the customer, as in the following figure:

Selecting only accepted actions



   c. Click **Apply changes**.
4. Click **Preview**
5. Click **Actions Save as** and enter a new name for the report. **For example:** Accepted actions by rank
6. Click **Done editing**. **Result:** The new report is now available in the Report Browser.
7. **Optional:** To automatically export the report results and send them out by email, schedule a recurring task.
   For more information see: Exporting reports

Accessing Interaction History data for external reports

Create insight from data that spans beyond the Pega application scope by extracting Interaction History for use with an external Business Intelligence reporting tool.

Pega Customer Decision Hub

1. Work with your team to identify the Interaction History data that is required for reporting.
2. Develop a staging table to hold Interaction History data.
3. Extract the data from Pega Customer Decision Hub by using the File data set.
   For more information, see "Exporting customer data from Pega Customer Decision Hub" in the *Pega Customer Decision Hub Implementation Guide*.
4. Transform the data as necessary and load it into your Business Intelligence reporting tool.

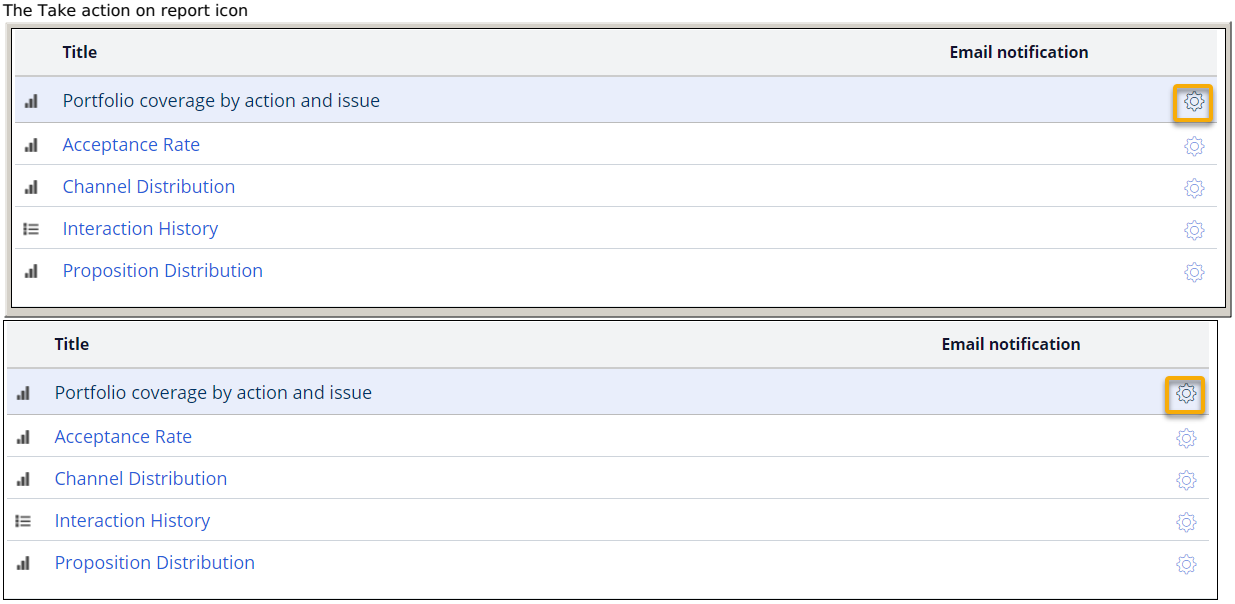
Exporting reports

To automatically export the report results and send them out by email, schedule a recurring task.

Pega Customer Decision Hub
Tip: A recurring export of report results is suitable for relatively low-volume reports that should be delivered to specific operators, for example, some KPI reports. For higher-volume reports, consider establishing an ETL process. For more information, see Accessing Interaction History data for external reports.

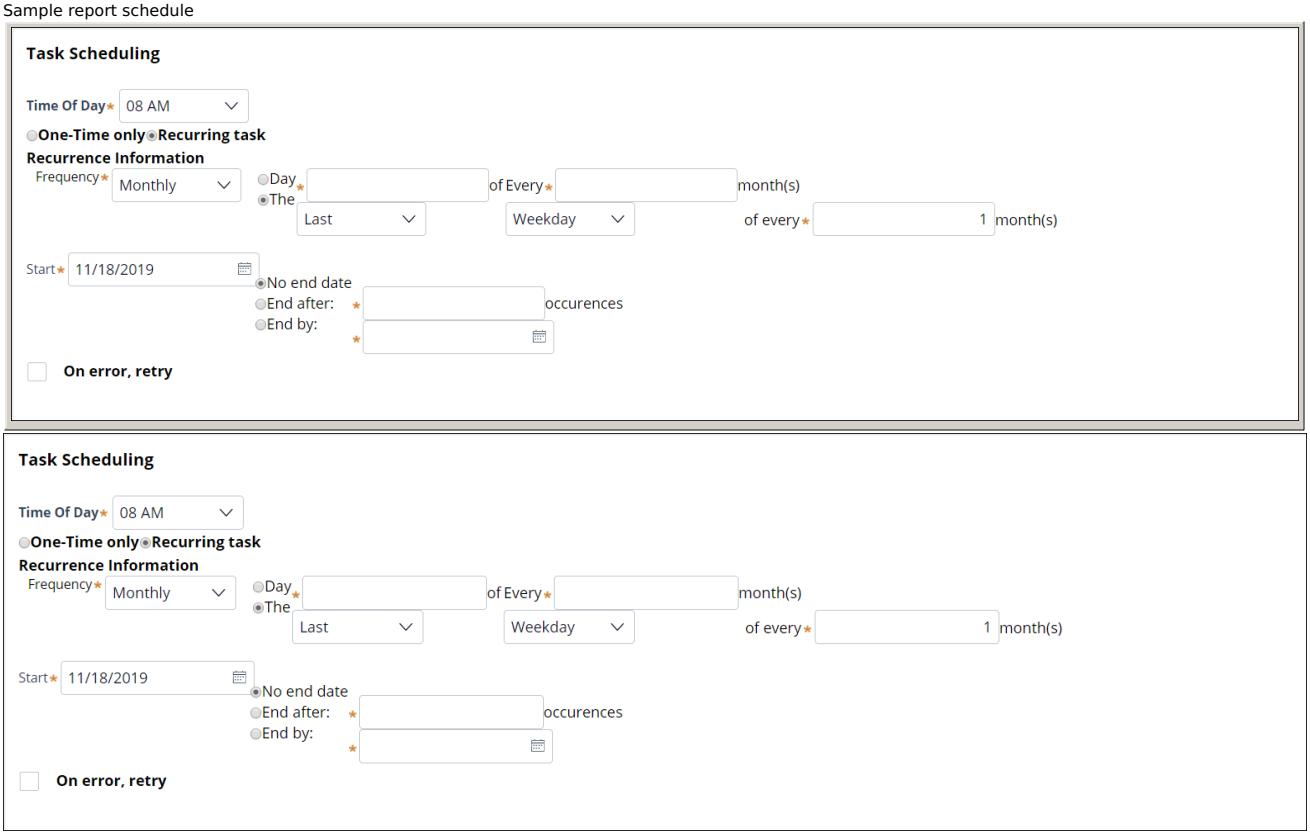
1. In the Pega Customer Decision Hub portal, click **Reports Report Browser** .
2. In the **Public categories** list, select **Interaction History**.
3. Click **Take action on report**, as in the following figure:

The Take action on report icon



4. Click **Schedule**.
5. In the **Task Scheduling** section, configure when the report should be sent out, as in the following figure:

Sample report schedule



Task Scheduling
Time Of Day* 08 AM
One-Time only Recurring task
Recurrence Information
Frequency* Monthly
Day* of Every* month(s)
The Last Weekday of every* 1 month(s)
Start* 11/18/2019
No end date
End after: * occurences
End by: *
On error, retry

6. In the **Task Output Processing** section, select the file format and recipients for the report.
   You can only send the report to users who have operator accounts in Pega Customer Decision Hub.
7. Click **Submit**.

Capturing negotiation outcomes from Pega Next-Best-Action Advisor

Pega Customer Decision Hub can capture information about retention or acquisition-related negotiations conducted by your customer service representatives (CSRs) with the help of Pega Next-Best-Action Advisor. Information about deals is stored in the database and can be accessed for reporting purposes, for example, to measure the agents' KPIs.

Pega Customer Decision Hub

Pega Next-Best-Action Advisor allows next-best-action recommendations to be displayed to customer service agents working in Pega Customer Service or a third-party customer service tool. Agents can use Pega Next-Best-Action Advisor to view the recommended offer, assess the customer's needs, and negotiate the pricing. The Negotiation capability of Pega Next-Best-Action Advisor gives the agents the necessary flexibility to define personalized pricing for the customer, while also taking into account the budget defined by the organization.

To support the agents in negotiating the most value for your organization, you can build KPI reports based on the outcomes of these negotiations captured by the CaptureNegotiationOutcome service.

Understanding how negotiation outcomes are captured by Pega Customer Decision Hub

The outcome capture process consists of the following stages:

1. The customer accepts, declines, or postpones the negotiated deal.
2. Pega Next-Best-Action Advisor sends POST requests to the *CaptureResponse* and *CaptureNegotiationOutcome* REST services available in Pega Customer Decision Hub.
3. Pega Customer Decision Hub records information about the interaction and the negotiation process in the following database tables:
   - The *CaptureResponse* service stores interaction details such as the ID of the actions presented to the customer, and the customer's response in the *pr_data_ih_fact* Interaction History database table.
   - The *CaptureNegotiationOutcome* service stores details about the negotiation process, such as the ID of the agent who conducted the negotiation, and how much of the allowed budget the agent used to retain or acquire the customer. This information is stored in the *pegadata.cdh_data_negotiation_outcome* database table. Additionally, the *pegadata.cdh_negotiation_ih_join* table associates the negotiation details with the interaction recorded by the *CaptureResponse* service.

Accessing negotiation outcome data for KPI reporting

You can define many varied reports based on the negotiation outcome data stored in the *pegadata.cdh_data_negotiation_outcome* database table. For example, you can measure the rate at which agents manage to retain customers by monitoring deal outcomes, or make sure that they are prioritizing the deals that bring the most value by comparing the amount of discount offered to the maximum negotiation budget.

You can export negotiation outcome data out of Pega Customer Decision Hub and into an external Business intelligence reporting tool by following the same process as for Interaction History reports. For more information, see Accessing Interaction History data for external reports.

- CaptureNegotiationOutcome REST service
- Negotiation outcomes data model

CaptureNegotiationOutcome REST service

CaptureNegotiationOutcome is a REST service that supports the POST HTTP method. Pega Customer Decision Hub uses this service to capture negotiation information from Pega Next-Best-Action Advisor and store it in the PegaDATA database, for example, for business intelligence reporting.

Pega Customer Decision Hub
The system calls the *CaptureNegotiationOutcome* service after a negotiation outcome in Pega Next-Best-Action Advisor, concurrently with the *CaptureResponse* service.

POST Service Request

For the POST request, the service expects a JSON Object with the following attributes:

Name	Required	Description
AgentID	✓	The identifier of the agent who conducted the negotiation.
AgentInteractionID	✓	A unique identifier associated with the negotiation interaction.
BudgetUsed	Optional	The budget that the agent used to acquire or retain the customer.
Commissions	Optional	The commission that the agent accrued.
ContextName	Optional	The context, or level of customer contact for multilevel decisioning. This property can represent, for example, an account, a household, or a policy membership.
CustomerID	✓	The customer identifier.
DiscountAmount	Optional	The amount of recurring discount offered.
InteractionID	✓	The interaction ID stored into the Interaction History Fact table for each cart item.
MaximumBudget	Optional	The maximum budget allowed by Pega Customer Decision Hub at the time when the negotiation interaction ends.
MinimumBudget	Optional	The minimum budget allowed by Pega Customer Decision Hub.
MonthlyRecurringCost	Optional	The recurring monthly cost of the offers for the customer.
MonthlyRecurringDiscount	Optional	The recurring monthly discount of the offers for the customer.
OneOffCost	Optional	The one-time cost of the offers for the customer.
OneOffDiscount	Optional	The one-time discount on the offers for the customer.
Outcome	✓	The outcome of the negotiation interaction.
TargetBudget	Optional	The target budget allowed by Pega Customer Decision Hub.

A sample invocation request is shown below:

"InteractionID":["5162592654195742888","5162592654195742890"],"MonthlyRecurringCost":331.99,"MonthlyRecurringDiscount":4.99,"CustomerID":"C1000012","AgentInteractionID":"S-441329","MinimumBudget":50.0,"TargetBudget":375.0,"DiscountAmount":4.99,"BudgetUsed":120,"Outcome":"Accepted","OneOffCost":225.0,"MaximumBudget":500.0,"AgentID":"csr01@uplusbank.com","OneOffDiscount":0.0,"Commisions":50,"ContextN

Negotiation outcomes data model

Pega Customer Decision Hub stores all properties related to negotiation outcomes in PegaDATA database tables. The recorded outcomes and associated Interaction History records can then be used to create business intelligence reports.

Pega Customer Decision Hub

Properties in the *pegadata.cdh_data_negotiation_outcome* table

The *pegadata.cdh_data_negotiation_outcome* table contains basic properties that describe the negotiation outcome.

Property	Data type and length	Description
agent_id	character varying (32)	The identifier of the agent who conducted the negotiation
agent_interaction_id	character varying (255)	A unique identifier associated with the negotiation interaction.**Note:** This property is used as the primary key to join the negotiation outcome with the related Interaction History record
budget_used	numeric (18,3)	The budget that the agent used to acquire or retain the customer
commisions	numeric (18,3)	The commission that the agent accrued
context_name	character varying (16)	The context, or level of customer contact for multilevel decisioning. This property can represent, for example, an account, a household, or a policy membership
customer_id	character varying (32)	The customer identifier
discount_amount	numeric (18,3)	The amount of recurring discount offered
maximum_budget	numeric (18,3)	The maximum budget allowed by Pega Customer Decision Hub at the time when the negotiation interaction ends
minimum_budget	numeric (18,3)	The minimum budget allowed by Pega Customer Decision Hub
monthly_recurring_cost	numeric (18,3)	The recurring monthly cost of the offers for the customer
monthly_recurring_discount	numeric (18,3)	The recurring monthly discount of the offers for the customer
one_off_cost	numeric (18,3)	The one-time cost of the offers for the customer
one_off_discount	numeric (18,3)	The one-time discount of the offers for the customer
outcome	character varying (16)	The outcome of the negotiation interaction
response_time	timestamp	Time when the outcome was recorded
target_budget	numeric (18,3)	The target budget allowed by Pega Customer Decision Hub

Properties in the *pegadata.cdh_negotiation_ih_join* table

The *pegadata.cdh_negotiation_ih_join* table contains the properties that join the *pegadata.cdh_data_negotiation_outcome* table to the *pr_data_ih_fact* Interaction History table.

Property	Data type and length	Description
agent_interaction_id	character varying (255)	A unique identifier associated with the negotiation interaction.**Note:** This property is used as the primary

Property	Data type and length	Description
		key for the *pegadata_cdh_data_negotiation_outcome* table.
pxinteractionid	numeric (19,0)	Unique 64-bit identifier for all records that belong to the same interaction. **Note:** This property is used as the primary key for the *pr_data_ih_fact* table.

Note: For more information about the *pr_data_ih_fact* database table and other Interaction History tables, see Pega Customer Decision Hub Interaction History data model.

Managing business changes

Use revision management to make everyday changes to, for example, the description or expiry date of a product, or even small changes to the risk score calculation.

Revision management enables business users to respond quickly to changes in the external and internal factors that influence their business. Responses might include introducing new offers, imposing eligibility criteria, or modifying existing business strategies.
Pega Customer Decision Hub

- Understanding business change management, revision management, and pipeline automation
- Using fast-track change requests in Pega Customer Decision Hub
- Managing business-as-usual changes with Pega 1:1 Operations Manager

Understanding business change management, revision management, and pipeline automation

With business change management, business users can respond to changing requirements by modifying and deploying your application's rules in a controlled manner.

Pega Customer Decision Hub

Business change management and revision management

Pega Platform provides business users with the means to implement and test modifications to their applications outside the enterprise release cycles. Revision management enables business users to respond quickly to changes in the external and internal factors that influence their business. Responses might include introducing new actions or propositions, imposing eligibility criteria, or modifying existing business strategies.

Use the revision management process to:

- Give business users the ability to make, test, and implement changes to business rules.
- Define the rules that are available to business users by creating an application overlay and managing revisions in the production environment.

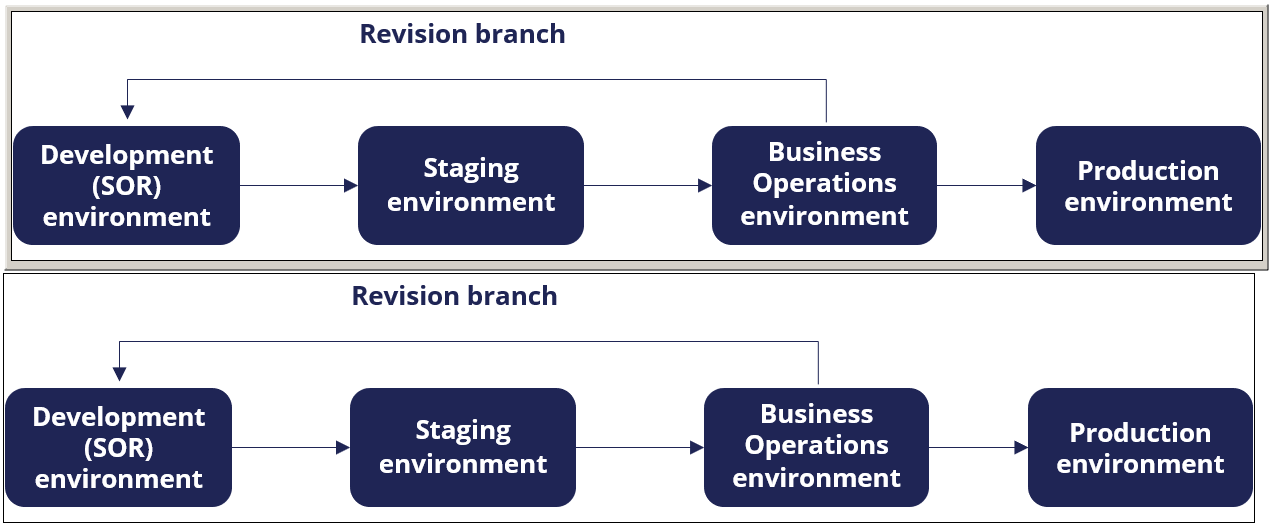
Depending on the type of rule that you want to modify, business changes can be launched from the Pega Customer Decision Hub portal, or from a Pega 1:1 Operations Manager application. For more information, see Understanding the relationship between Pega 1:1 Operations Manager and revision management.

Business change management and pipeline automation

Although revision management packages can be manually created and deployed to the target environment, you can enhance your business change management process by using the automated deployment pipelines provided by Pega's Deployment Manager 5.1 or newer. Use Deployment Manager to increase the efficiency of business-as-usual application changes and automatize the deployment of revision packages.

Instead of manually deploying revision packages, you can run CI/CD pipelines to merge a revision package to the development system of record (SOR) and subsequently deploy that package through to your production environment by using a deployment pipeline.

Changes made in your business operations environment are merged to the system of records and deployed to each environment, as shown in the following figure:



- Understanding the relationship between Pega 1:1 Operations Manager and revision management
- Application overlays and the business change management process
- Launching a revision change request

Understanding the relationship between Pega 1:1 Operations Manager and revision management

If you use Pega Customer Decision Hub and your organization has adopted Next-Best-Action Designer, use Pega 1:1 Operations Manager to manage business-as-usual changes to Action and Treatment rules.

Pega Customer Decision Hub

Pega 1:1 Operations Manager is a new module of Pega Customer Decision Hub. Use Pega 1:1 Operations Manager in conjunction with revision management to create and deploy new actions to production, as well as request updates to existing actions.

Pega 1:1 Operations Manager extends the standard revision management capabilities of Pega Customer Decision Hub by providing a simple process where a business-as-usual change request is triggered by a non-technical team member with the use of a transparent user interface. In this way, Pega 1:1 Operations Manager makes it possible to quickly record and process standard, high-volume change requests related to actions.

Pega 1:1 Operations Manager uses the same flows and activities as standard revision management. Because of that, it seamlessly integrates with revision management and complements normal revision management operations. For example, if a change request is withdrawn or rejected in Pega 1:1 Operations Manager, the status also changes in Revision Manager. If a revision is withdrawn or rejected in Revision Manager, the status change is reflected in Pega 1:1 Operations Manager.

For more information, see the *Pega 1:1 Operations Manager User Guide* on the Pega Customer Decision Hub product page.

Tip: All rules available in Pega 1:1 Operations Manager can also be managed by using the standard revision management process. However, using Pega 1:1 Operations Manager simplifies and streamlines business change management by empowering non-technical users, such as marketers, to submit change requests directly in a dedicated portal. Revision managers no longer need to be involved in submitting and processing simple, business-as-usual change requests.

Application overlays and the business change management process

An application overlay is an application that is built on top of a decision management enterprise application. An application overlay defines the scope in which business users can change the application (for example, by managing propositions, modifying business rules, or running simulations) to adjust the application to constantly changing business conditions and requirements.

Pega Customer Decision Hub

System architects define the application overlay components, such as the revision ruleset or revision records, during the implementation process of Pega Customer Decision Hub. For more information, see the *Pega Customer Decision Hub Implementation Guide* on the Pega Customer Decision Hub product page.

Launching a revision change request

As a revision manager, launch a change request by first creating a new revision, and then adding an associated change request within that revision.

Pega Customer Decision Hub
Note: For operations change requests launched by business users through Pega 1:1 Operations Manager, the corresponding revision is created automatically. As a revision manager, you do not need to manually create it.

1. Log in to the Pega Customer Decision Hub portal as an operator with the *RevisionManager* access role.
2. In the left navigation panel, click **Revision Management**.
3. On the Revision Management landing page, click **New revision**.
4. On the **Set up new revision** screen, select **Create new revision option is selected**.
5. Enter the name of the revision, for example, Set the Apple iPhone price to 200.
6. Define the revision objective, for example, Modify product pricing info.
7. Define the priority of the revision by modifying the **Urgency** setting.
   The lower the urgency value, the more priority is assigned to the revision.
8. Assign the deadline for completing the revision by configuring the **Due by** setting.
9. Click **Continue**.
10. Select the rules that you want to include in the revision, or specify the type of rule that you want to create.
11. In the **Change request** section, click **Edit**.
12. In the **Edit Change Request Details** window, in the **Assign to section**, select **Operator**.
13. From the drop-down menu, select the operator that you want to assign to the change request.
14. Provide the name of the change request, for example, Update the Apple iPhone price to 200.
15. **Optional:** To associate this change request with an objective that is different than the revision objective, modify the **Associate an objective field**.
16. **Optional:** In the **Description** field, provide additional details, for example, Modify the Price property.
17. **Optional:** Define the change request priority in the **Urgency** field.
18. **Optional:** Define the completion deadline for this change request in the **Due by** field.
    The default value is seven days.
19. Click **Submit**.
20. In the **Refine selection** section, include the propositions whose configuration you want to update.
21. Click **Submit**.

Result: The change request advances to the Open stage. A strategy designer who has the necessary access rights to work on the tasks required to complete the request may now start working to build the necessary changes. For more information, see Completing change requests.

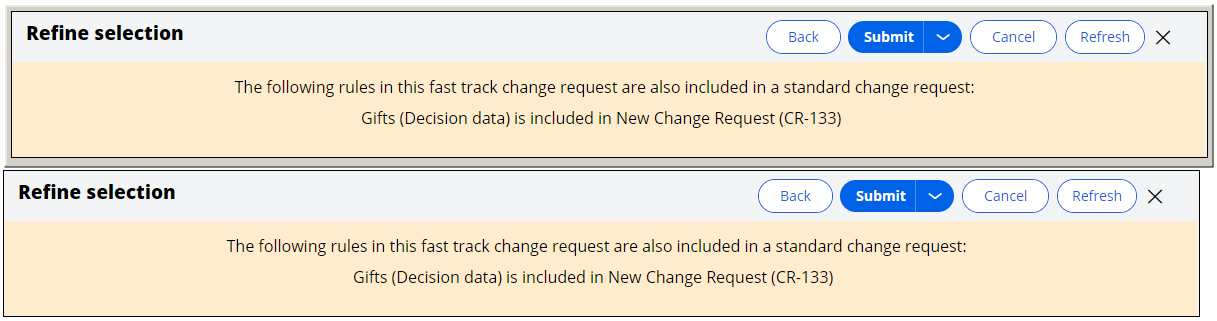
- Completing change requests
- Resolving revisions
- Launching fast-track change requests
- Deploying revisions with Deployment Manager
- Optional: Managing revisions manually
- Supported rule types for revision management
- Stages of the Revision case type
- Stages of the Change Request case type

Completing change requests

As a strategy designer, complete the change request by performing the required changes and submitting your work for approval.

Pega Customer Decision Hub
This document describes working with a change request submitted by a revision manager in the Pega Customer Decision Hub portal. For information about completing change requests submitted by business user in Pega 1:1 Operations Manager, see the *Pega 1:1 Operations Manager User Guide* on the Pega Customer Decision Hub product page.

1. Log in to the Pega Customer Decision Hub portal as an operator with the *RevisionManager* access role.
2. On the **Home** screen, in the **Work under Revision Management** section, open the change request.
3. In the **Scope of changes** section, open the rule that you want to update.
   Note: Some rule types open in a new portal. For example, Prediction rules open in Prediction Studio. Make sure that your operator account has the necessary access group to open the rule type.
4. Update the rule as requested.
5. Click **Submit**.
6. **Optional:** To increase the priority of the business changes, convert the change request into a fast-track request by clicking **Change to fast track**.
   Caution: If the changes in a fast-track change request conflict with any standard change request, the following error message is displayed:



   Resolve potential conflicts by manually propagating the rule changes to the standard change request or by withdrawing the conflicting rules from the standard change request. Otherwise, the changes that you introduced in the standard change request overwrite any changes that you made in advance.For more information about fast track change requests, see Launching fast-track change requests.

Result: The updated rule is submitted for review. The revision manager assesses whether the changes that the strategy designer made are satisfactory and approves the revision for deployment.

Resolving revisions

As a revision manager, assess whether the rule updates are satisfactory. If you approve of the changes, you can submit the revision for deployment. Before you can submit a revision, you must approve all change requests that comprise that revision.

Pega Customer Decision Hub

1. Log in to the Pega Customer Decision Hub portal as an operator with the *RevisionManager* access role.
2. On the **Home** screen, in the **Work under Revision Management** section, open a change request that the strategy designer completed.
3. In the **Scope of changes** section, inspect the rule changes.
4. To accept the changes, click **Approve**.
5. Click **Submit**.
6. Repeat steps 2-5 for each change request associated with the revision that you want to submit for deployment.
7. After you resolve all change requests that comprise a revision, ensure that you understand the impact of the changes by selecting the **Simulate revision** action. Simulating the revision tests the changes using predefined simulation settings. Make sure that you understand the impact of the changes before deploying the revision.
8. Submit that revision for deployment to other environments.
   - The recommended way to deploy your revisions is by using Pega Deployment Manager 4.8 or newer. For more information, see Deploying revisions with Deployment Manager.
   - Alternatively, you can manually package the revision for deployment by clicking **Submit Download revision** . After that, you can manually import the revision into the target environment. For more information, see Optional: Managing revisions manually.

Launching fast-track change requests

Address high-priority business needs efficiently through fast-track change requests. This type of change request provides a mechanism to quickly deploy urgent changes, without disrupting revisions that are in progress.

Pega Customer Decision Hub
Before you begin: As a system architect, enable fast-track change requests in your application by assigning business users with the following access groups:

<overlay_name>FastTrackRevisionManager
Initiates and resolves fast-track change requests.
<overlay_name>FastTrackStrategyDesigner
Amends or creates business rules as part of a fast-track change request, in the context of a production application and in isolation from standard revision management process.

For more information, see Creating application overlays.

To complete a fast-track change request, strategy designers can address high priority changes within the context of the enterprise application, isolated from the regular revision process. Upon approval, such change requests are immediately ready for deployment as part of a new, automatically created revision.

As a revision manager, you can create a fast-track change request in the following ways:

- Create a dedicated fast-track change request by selecting the **Fast track** option during change request creation. For more information, see Launching a revision change request.
- Convert a regular change request that has been approved. For more information, see Completing change requests.

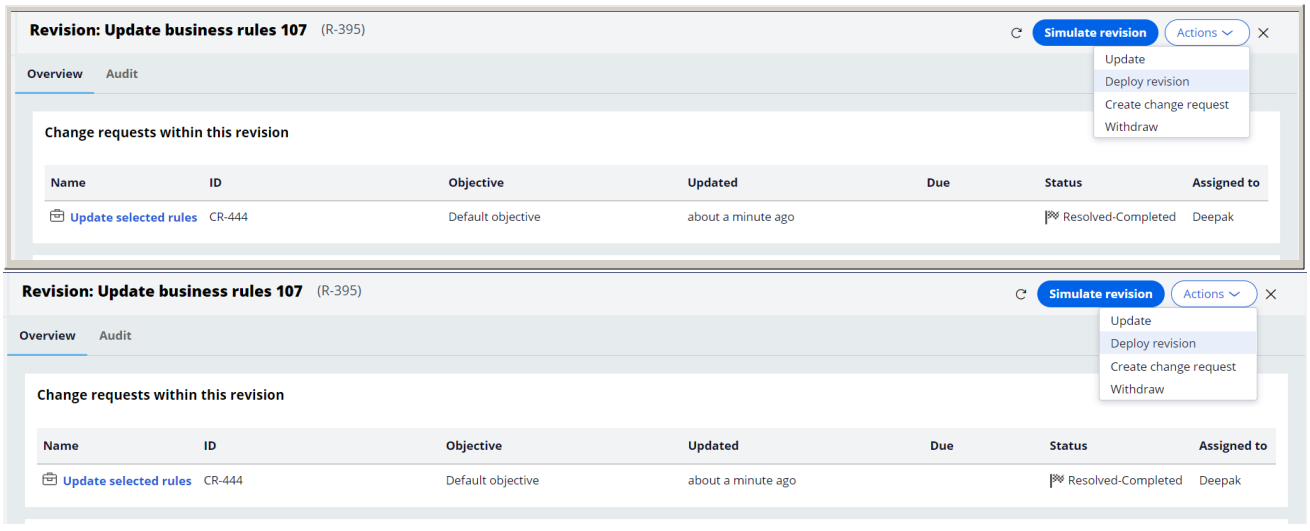
Note: Initiate fast-track change requests with caution. For changes that do not require immediate deployment, create standard change requests instead.

Deploying revisions with Deployment Manager

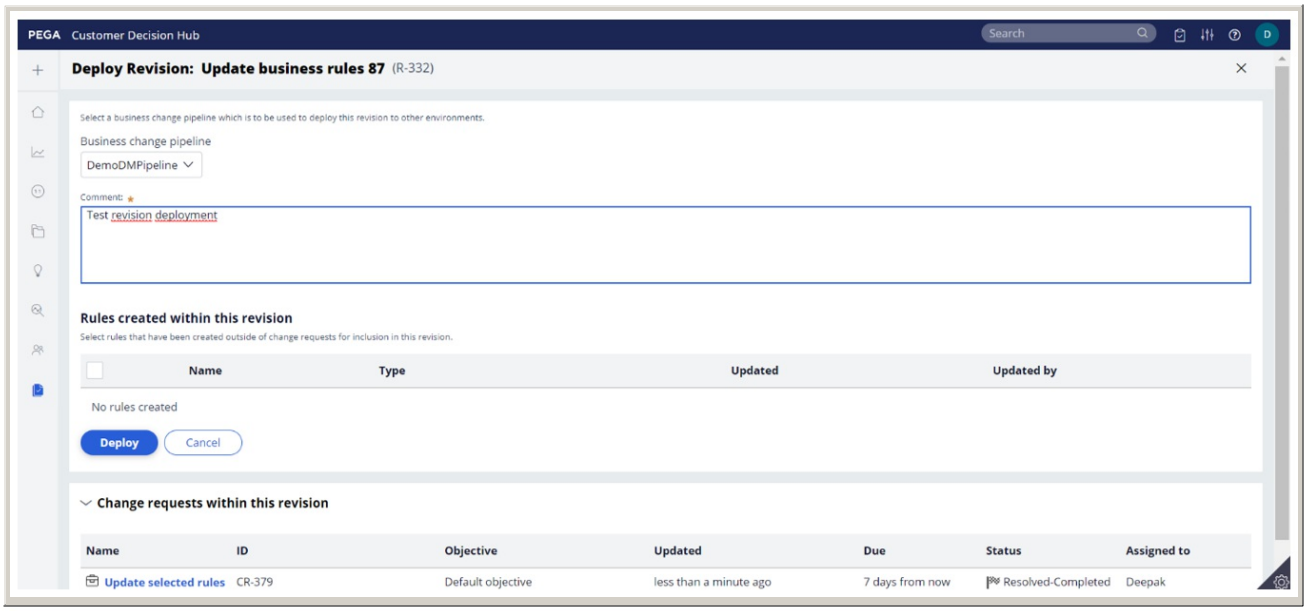
As a best practice, you should use Deployment Manager 5.1 or newer to automate your business change management process, run CI/CD pipelines to merge a revision package to the development system of record (SOR), and subsequently deploy that package through to your production environment by using a deployment pipeline.

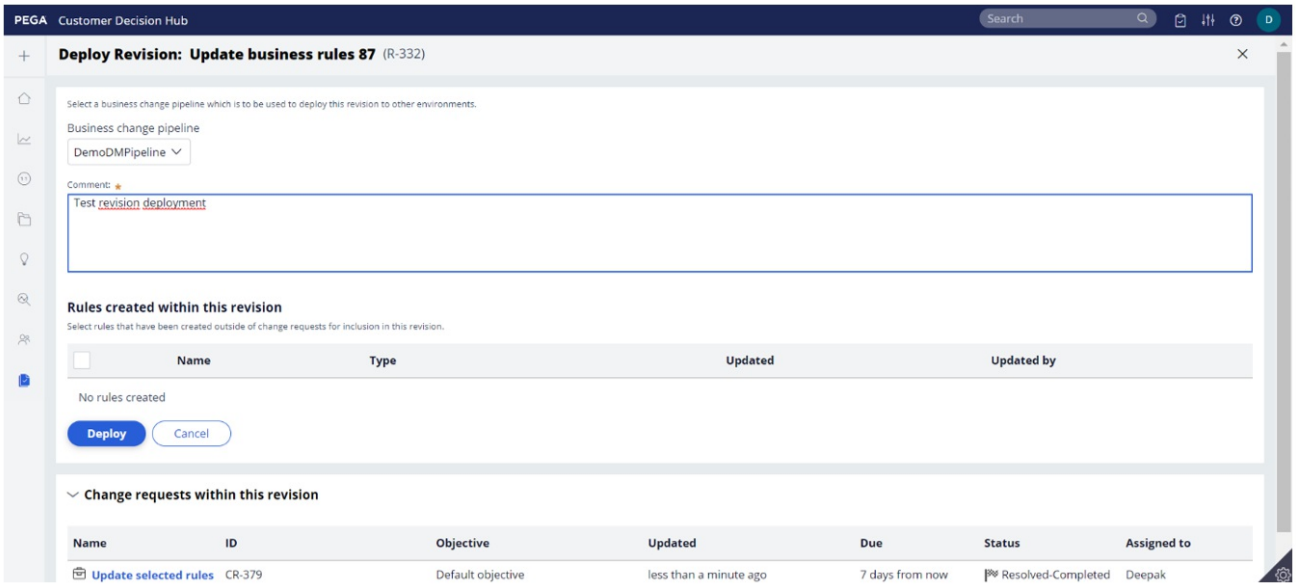
Pega Customer Decision Hub
Before you begin: Configure Deployment Manager. For more information, see Using Deployment Manager for model-driven DevOps.**Important:** Your Deployment Manager version must be 5.1 or newer. Previous versions of Deployment Manager do not support working with revision management packages. For information about upgrading Deployment Manager, see Installing or upgrading Deployment Manager 5. After a revision manager submits a change request in the business operations environment and a strategy designer implements the required changes, the new revision must be deployed to production. To automatize the deployment process, use Deployment Manager pipelines. You can merge a revision to a deployment pipeline directly from the Pega Customer Decision Hub portal.

1. Log in to the Pega Customer Decision Hub portal as an operator with the *RevisionManager* access role.
2. In the left navigation panel, click **Revision Management**.
3. On the Revision Management landing page, click **Revisions**.
4. Click the revision that you want to deploy and review its contents.
5. Click **Deploy revision**.

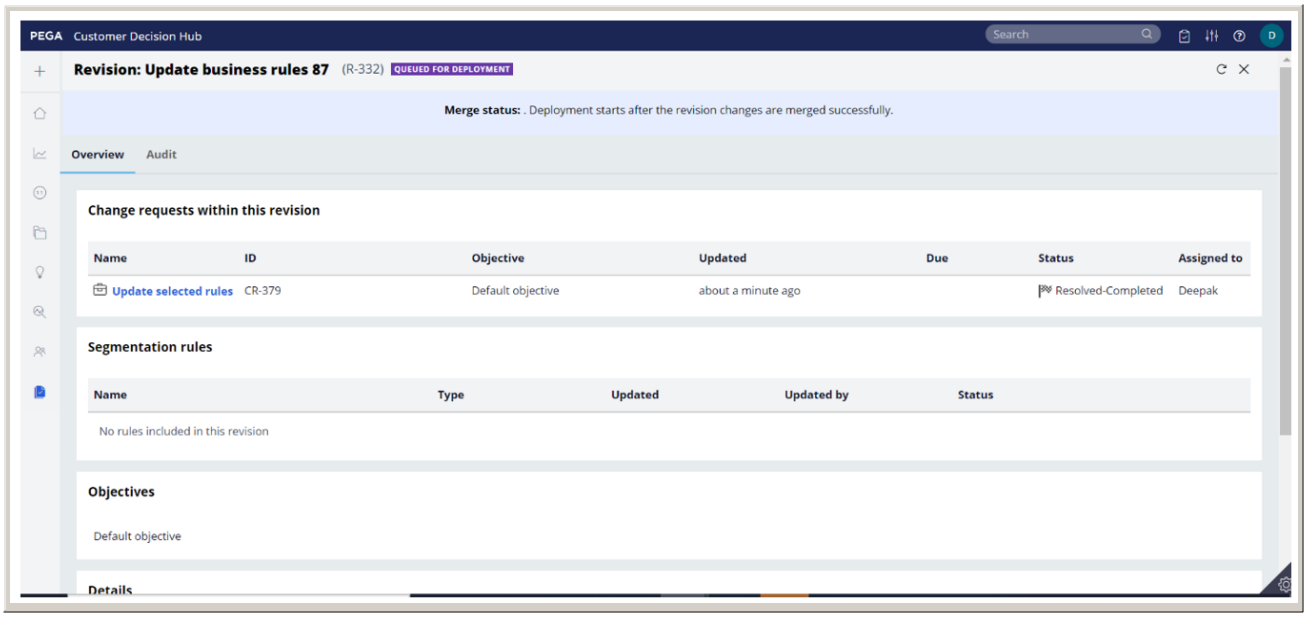


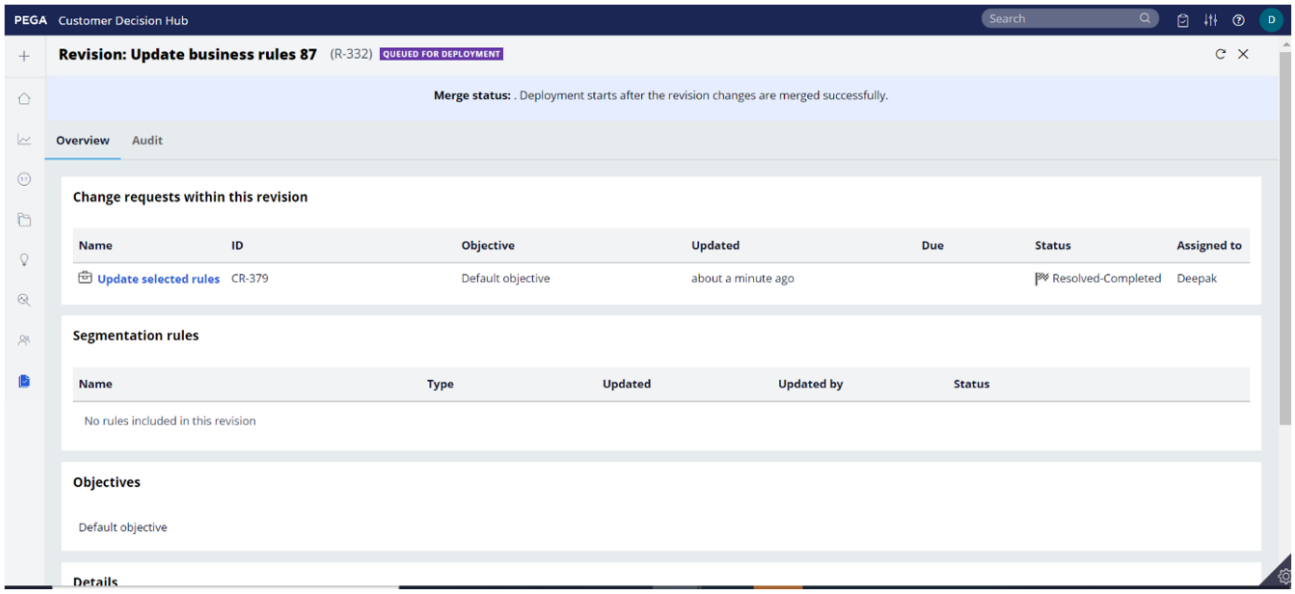
6. In the **Pipelines** section, select a Deployment Manager pipeline to merge the package to the development system of record, as shown in the following figure:



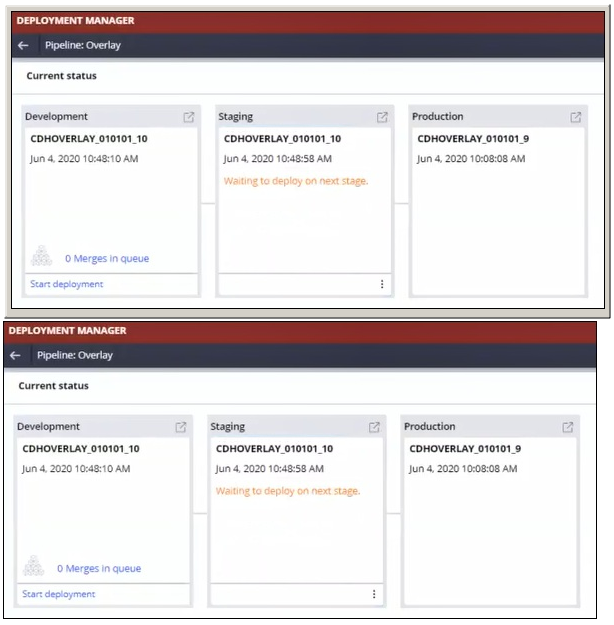


7. As a best practice, in the **Comment** field, enter a short description of the changes included in the revision, for example, Introduced a new business issue.
8. Click **Deploy**. **Result:** Deployment Manager queues the revision to merge it into a pipeline. After a successful merge, the revision deploys automatically to all environments in the pipeline. You can monitor the status of the deployment process in the Pega Customer Decision Hub portal, as shown in the following figure:

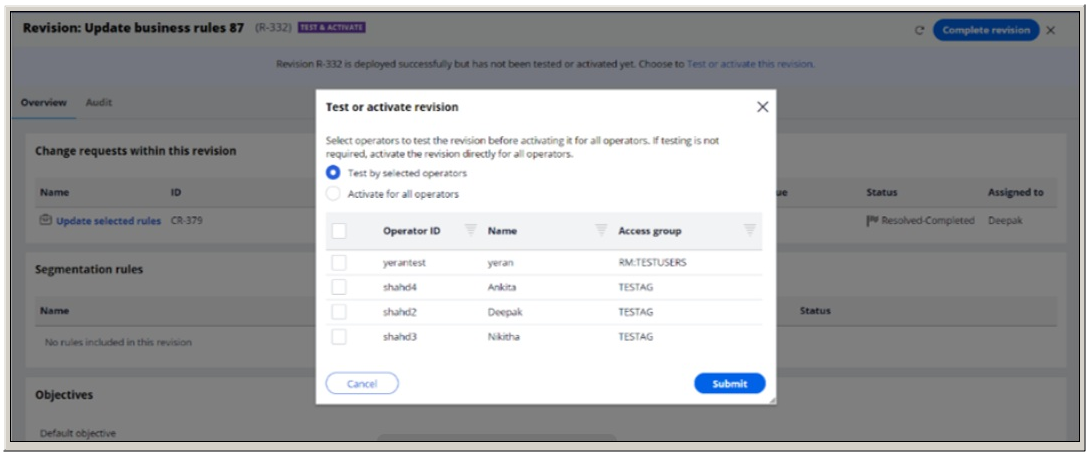


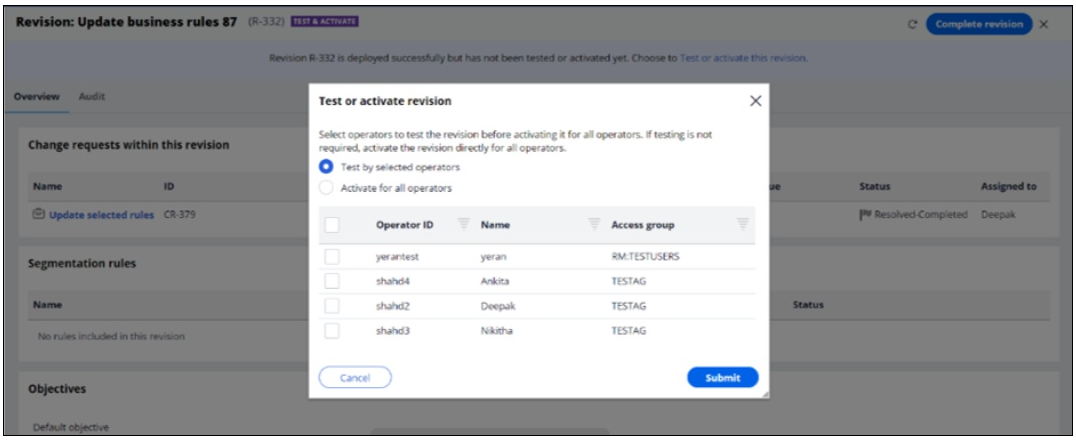


9. **Optional:** Depending on the configuration of your deployment pipeline, you may need to manually approve the revision for deployment to your production environment. If required, click the **Deployment Manager portal** link and approve the revision in Deployment Manager, as in the following figure:



10. **Optional:** If the revision deployment fails at any stage, click **Cancel** to complete the revision with status Completed-Failed.
11. When the revision reaches the Test & Activate stage, click **Test and activate this revision** to select the operators to test the revision before you activate it, as in the following figure:





Tip: If you do not want to test the revision, select **Activate for all operators** instead. Selecting this option activates the revision directly.

12. Click **Complete revision** to resolve the revision and allow the deployment of new revisions.

- Understanding business change management, revision management, and pipeline automation

Optional: Managing revisions manually

As a best practice, you should always use Deployment Manager 5.1 to automate your business change management process. However, if necessary, you can also manually import, activate, discard, or roll back revisions in the production environment.

Pega Customer Decision Hub
Each operation that you complete causes changes in the system, such as changes to application version numbers or to access groups.

- Importing revisions
- Activating revisions
- Discarding revisions
- Rolling back revisions

- Understanding business change management, revision management, and pipeline automation

Importing revisions

A revision manager creates the revision package in the business sandbox, in the context of an application overlay. You can import that package to Pega Platform to propagate rule changes to the production environment or first test the changes with a selected group of application users.

Pega Customer Decision Hub

Business users can activate revisions through direct deployment of revisions. However, as a system architect, you can preserve or discard direct deployment revisions while importing a revision. For more information, see the Pega Community article *Direct deployment of revisions in decision management*.

1. In Dev Studio, click **Configure Decisioning Infrastructure Revision Management Revisions** .
2. Click **Import revision**.
3. In the **Import Revision Package** wizard, click **Import**.
4. Perform the following steps in the **Import Revision Package** wizard:
   a. Load the revision package.

      Upload the revision package that you want to activate in the production environment.

      1. Click **Choose File**.
      2. Select the file from your local directory.
      3. Click **Next** to advance to the next step.

   b. Select the revision package.

      Select a revision package from all packages included in the file that you imported. You can view information about each package to avoid an incorrect selection.

      1. Expand the **Select package** drop-down menu.
      2. Select a package from the list of available items.
      3. Click **Next** to advance to the next step.

   c. Review the contents.

      View all rules that were modified as a result of the revision management process. You can also view the rule update information and any comments provided while submitting modifications as part of change requests.

      - Click **View Comments** to display any annotations or remarks associated with a rule change.
      - Click **Next** to advance to the next step.

   d. Deploy the revision package.

      Test the changes or deploy the revision without testing.

      Perform one of the following actions:

      - Test the revision:

        1. Select **Deploy for at least one test user**.
        2. Select the test operators. Only the operators whose default access group corresponds to the application where the revision is imported can test revisions. If no test operators are available, you are the test operator.
        3. Click **Test**. The revision is deployed for test operators. Its status changes to **Testing**.

      - Activate the revision for all users without testing:

        1. Select **Deploy and activate for all users**.
        2. Click **Activate**. The revision is deployed and its status changes to **Active**.

- Understanding business change management, revision management, and pipeline automation
- Activating revisions
- Discarding revisions
- Rolling back revisions

Activating revisions

Activate a revision in test to propagate the changes included in the revision to all operators. Activate only those revisions that have been approved by testers. You can have only one active revision in the system at a time.

Pega Customer Decision Hub

1. In Dev Studio, click **Configure Decisioning Infrastructure Revision Management Revisions** .
2. Click **Activate** next to a revision with the **Test** status.

Result:

The status of the revision changes to **Active**. The activation of a revision results in the following system changes:

- The application version is incremented by one, for example, *PegaBank 01.01.02*.
- All overlay applications point to the new version.
- The access group for which the revision was activated changes to reflect the new application version. To view the changes, application users must log out and then log in again.
- A *Rollout* access group is created, for example *PegaBank:AdministratorsRollout*, which points to the new application with the revision ruleset.

- Understanding business change management, revision management, and pipeline automation
- Importing revisions
- Discarding revisions
- Rolling back revisions

Discarding revisions

When test users finish testing a revision and find that it is not working as expected, you can discard the revision from the production system.

Pega Customer Decision Hub

1. In Dev Studio, click **Configure Decisioning Infrastructure Revision Management Revisions** .
2. Click **Discard** next to a revision with the **Test** status.

Result:

The revision is removed. Discarding a revision results in the following system changes:

- The current application version is deleted.
- The access group of application operators is updated to point to the application version that immediately preceded the version number of the discarded revision. For example, if you discard application version *01.01.02*, the operator's access group will point to version *01.01.01*.
- All overlay applications, including the direct deployment overlay, point to the previous application version.
- The *Rollout* access group is removed and changes to the previous access group version. For example, *PegaBank:AdministratorsRollout* changes back to *PegaBank:Administrators*.

- Understanding business change management, revision management, and pipeline automation
- Importing revisions
- Activating revisions
- Rolling back revisions

Rolling back revisions

You can roll back revisions that are already in production. You might do this when you find serious issues with the new application version that were not discovered during testing.

Pega Customer Decision Hub

When a revision is rolled back, the system withdraws this revision from the production environment and restores the previous active revision.

1. In Dev Studio, click **Configure Decisioning Infrastructure Revision Management Revisions** .
2. Click **Roll-back** next to the active revision that you want to withdraw. The revision status changes to **Rolled back**.

Result:

Revision rollback causes the following system behavior:

- A new application version is created. The new application is the same as the last active application version prior to the revision import that created the application version that is now rolled back. For example, if you roll back application version *01.01.02*, the new application version number is *01.01.03* and the contents of that application version are identical to version *01.01.01*.
- The new application version is active.
- All access groups for which the revision was activated point to the new application version.
- All overlay applications, including the direct deployment overlay, point to the new application version.

- Understanding business change management, revision management, and pipeline automation
- Importing revisions
- Activating revisions
- Discarding revisions

Supported rule types for revision management

The revision management functionality provides business users with means to request, make, audit, and test changes to enterprise applications. The following table presents the rules that are available in revision management.

Pega Customer Decision Hub

Depending on the rule type, change requests can be launched through Pega Customer Decision Hub, or through a Pega 1:1 Operations Manager application. All rules available in Pega 1:1 Operations Manager can also be managed by using the standard revision management process. However, using Pega 1:1 Operations Manager simplifies and streamlines business change management by empowering non-technical users, such as marketers, to submit change requests directly in a dedicated portal.

Rule type	Available in Pega 1:1 Operations Manager?
• Action • Treatment	Yes
• Adaptive Model • Aggregation Dataset • Analysis Project • Contact Policy • Data Flow • Data Set • Data Transform • Decision Data • Decision Table	

Rule type	Available in Pega 1:1 Operations Manager?
• Decision Tree • Email Treatment • Event Strategy	
• Geofence • Map Value • NBA Definition • NBA Hierarchy • Output Template • Passbook Treatment • Predictive Model • Property • Proposition Filter • Real-time Container • Real-time Event • Report Definition • Sample • Scorecard Strategy • Segment • SMS Treatment • Strategy Builder • Supporting Decision • Test Case • Test Case Result • Text Analyzer • Volume Constraint • When	No

- Understanding the relationship between Pega 1:1 Operations Manager and revision management

Stages of the Revision case type

The primary purpose of the *Revision* case type is to initiate the process of changing business rules in your application. This case type covers all aspects of the revision life cycle. A revision can have one or more change requests associated with it. You can modify the stages, steps, processes, or assignments that are part of the *Revision* case type to make it simpler or more complex, depending on the business needs.

Pega Customer Decision Hub

The primary path in the *Revision* case type is represented by the following stages:

Stage	Step	Status	Purpose
Create	Define Revision	Final	Provide revision details.
	Create Revision	Final	Create a revision through the designated activity.
	Add Change Request	Final	Create one or more change requests through the designated activity.
Open	View Revision	Available	View the revision. Optionally, update revision details, add change requests, withdraw or submit revision.
Submit	Simulate revision	Available	Monitor the progress of the revision simulation and review the performance trend.**Note:** To use the **Simulate revision** stage in your revision process, you must configure the necessary settings under the *pySetRevisionSimulationOptions* data transform. For more information, see Simulating your revision changes.
	Package Revision	Available	Create a RAP file that contains the modified rules.
Direct Deployment	View DD Revision	Available	Activate or withdraw a revision, independently from application administrators. Available if direct deployment of revisions is enabled.
Activate	Activate Revision	Final	Make a revision available for testing or in the production environment.
	Initiate Rollback	Available	Withdraw a revision from the test or production environment.

The *Revision* case type can also have the following alternate stages:

Stage	Step	Status	Purpose
Withdraw	Withdraw Revision	Final	Withdraw a revision if in Open or Direct Deployment stage.
Rollback	Roll Back Revision	Final	Withdraw an active revision through a designated activity.

- Simulating your revision changes
- Stages of the Change Request case type

Simulating your revision changes

As part of the revision life cycle, you can verify the performance impact of your changes within the revision package. To enable this capability, configure the *pySetRevisionSimulationSettings* data transform.

Pega Customer Decision Hub
After you configure the data transform, every time you complete a revision and are at the Simulate step, a funnel simulation is initiated with the same input audience and strategy. Once completed, it shows the indicative performance of the simulation which you can use to track performance trends from revision to revision.**Note:** All the following steps must be done from the context of the enterprise application for which the overlay is created, and not the overlay application itself. Overlay applications usually do not have an open ruleset version, so it is not possible to save the data transform into your implementation ruleset and extend it when in the overlay application context.

1. Log in to Dev Studio as an operator with access to the Revision Management overlay application.
2. In the left navigation pane, click **Records Data Model Data Transform** .
3. Search for and open the *pySetRevisionSimulationSettings* data transform in the *RevisionManagement-Work* class.
   Note: The default audience for the data transform is set to use the customer sampling data set from the **Data migration** landing page, and the default strategy is set to use the Next-Best-Action Designer top-level strategy.
4. Click **Save as**, and then save the data transform to an enterprise application ruleset that is visible to the Revision Management overlay application, for example, the *Artifacts* ruleset.
5. Extend the new data transform to set the source and strategy details for the revision simulation.

Stages of the Change Request case type

By default, the *Change Request* case type is the subcase that is created in the first stage of the *Revision* case type life cycle.

Pega Customer Decision Hub

The *Change Request* case type defines the way that change requests are created, submitted, and resolved. You can modify the stages, steps, processes, or assignments that are part of the *Change Request* case type to make it simpler or more complex, depending on the business needs.

The primary path in the *Change Request* case type is represented by the following stages:

Stage	Step	Status	Purpose
Create	Define Changes	Final	Provide change request details.
	Create Change Request	Final	Create a change request through the designated activity.
Open	View Change Request	Available	Review the change request. Optionally, update change request details or withdraw it.
Submit for testing	Submit for testing	Pending-Testing	Simulate and test the changes of this change request, before submitting and completing it. Open a Next-Best-Action Designer configuration rule, action rule, or proposition filter rule, and run audience simulations.
Submit	Submit Changes	Final	Submit the change request for review and approval.
	Approval Required?	Available	Decide whether approval is required.
Approve	Approve Changes	Available	View the list of rules for approval.
Reassign	Reassign CR	Final	View the rules in the change request to assign back to the originator.

The situations in which you cancel, withdraw, or reject change requests are represented by the following alternate stages:

Stage	Step	Status	Purpose
Withdraw	Withdraw Changes	Final	Withdraw a change request in-progress.
Reject	Reject Changes	Final	Reject the change request after it has been reviewed.

- Stages of the Revision case type

Using fast-track change requests in Pega Customer Decision Hub

Pega Customer Decision Hub uses the fast-track change requests feature introduced in Pega Platform. Use fast-track change requests to address high-priority or emergency rule changes without the need to disturb a revision that is in progress.

Pega Customer Decision Hub

As a revision manager, you can start fast-track change requests directly on the Revision Management landing page. In addition, you can convert an approved standard change request to a fast-track change request. For more information, see Resolving fast-track change requests.

Managing business-as-usual changes with Pega 1:1 Operations Manager

Pega 1:1 Operations Manager is a new module of Pega Customer Decision Hub. Use Pega 1:1 Operations Manager in conjunction with Revision Management to push business-as-usual changes into production.

Pega Customer Decision Hub

For more information, see the *Pega 1:1 Operations Manager User Guide* on the Pega Customer Decision Hub product page. The application is not intended to be deployed to production.

Configuring the email and SMS accounts used by Pega Customer Decision Hub

Configure email and SMS accounts so that Pega Customer Decision Hub can send emails, as well as receive and send SMS messages.

Pega Customer Decision Hub

- Configuring an outbound email account
- Configuring an outbound SMS account
- Configuring an inbound SMS account
- Configuring SMS delivery receipts

Configuring an outbound email account

Configure the predefined email accounts which Pega Customer Decision Hub uses to send out emails, including messages related to actions or campaigns. Optionally, you can also set up additional email accounts for action-related emails.

Pega Customer Decision Hub
Note: Prepackaged connectors for email service providers (ESPs) Mailgun, SendGrid, and Mailjet are available on Pega Marketplace. Install the connectors to help reduce deployment time by automating the configuration of an email account.

1. In App Studio, click **Settings Channels Email** .
2. Click the **Edit** icon by a predefined email account to configure it.
   By default, Pega Customer Decision Hub includes the following outbound email accounts:
   - **Default** - This account is used to send out most types of emails, including password reset emails, Email and Passbook Treatments in actions, and other internal and external notifications. It is not used for Campaign-related emails.
   - **PegaMKT-Work** - This account is used to send Campaign-related emails, including Campaign Suspended and Campaign Approved notifications.

   You must provide the required connection data for both accounts to ensure that emails to customers and internal users are properly delivered.
3. **Optional:** If you want Email, Passbook and Multi Treatments in actions to be sent from a non-default email address, click **Add new** to add a new email account.
   Note: After configuring the new email account, select it as the delivery account for the Send Email, Send Passbook, and Send Multi shapes in the actions where you want to use the non-default email address for correspondence.
4. **Optional:** In the **Account Availability** section, specify delivery time frames for email related to actions.
   Select one of the following options:
   - **Always Available** - The account is always available to deliver messages.
   - **Specify Range** - The account only delivers messages during the specified time range. Specify the following value for the time range:
     - **From** - Start value for the time range, for example, 9.00 AM
     - **To** - End value for the time range, for example, 7.00 PM
     - **Time Zone** - Time zone for the start and end time values. By default, Pega Customer Decision Hub uses the time zone of the current user.

   Note: Delivery time frames do not apply to emails which are not related to actions, for example, work object-related notification.

Configuring an outbound SMS account

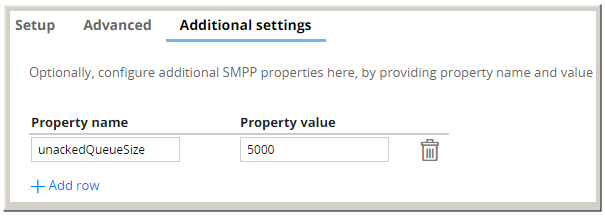
Configure the predefined outbound SMS accounts which Pega Customer Decision Hub uses to send out SMS messages, including messages related to actions. Optionally, you

can also set up additional SMS accounts for action-related messages.

Pega Customer Decision Hub

1. In App Studio, click **Settings Channels SMS Outbound** .
2. Click the **Edit** icon by the **Default** account to configure it.
3. In the **Setup** tab, fill out the required connection details:
   - **Host Address** - The address of the SMS center host, for example, my.smsc.com.
   - **Host Port** - The port number of the SMS center host, for example, 1041.
   - **User ID** and **Password** - Authentication credentials for the SMS account.
   - **Test Number** - The number which is used to receive test messages for this account.
   - **Sender's Number** - The number which is displayed as the sending number for messages sent from this account.
4. **Optional:** In the **Maximum Connections** field, enter the maximum number of connections available at the same time.
5. **Optional:** Select **Track delivery receipt** to enable delivery receits. For more information, see Configuring SMS delivery receipts.
6. **Optional:** Set **Delivery mode** to **Asynchronous** to send messages in bulk using the asynchronous option.
   Configure the following options:
   - **Unacknowledged queue size** - The maximum queue size for messages that were delivered to a consumer but not yet acknowledged.
   - **Wait on rejection (seconds)** - The interval to wait for the message to be rejected.
   - **SMS messages per second** - The number of messages to be sent out per second.
   - **Acknowledgement timeout (milliseconds)** - The interval before a message is considered unacknowledged.
7. **Optional:** In the **Account Availability** section, specify delivery time frames for SMS messages related to actions.
   Select one of the following options:
   - **Always Available** - The account is always available to deliver messages.
   - **Specify Range** - The account only delivers messages during the specified time range. Specify the following value for the time range:
     - **From** - Start value for the time range, for example, 9.00 AM
     - **To** - End value for the time range, for example, 7.00 PM
     - **Time Zone** - Time zone for the start and end time values. By default, Pega Customer Decision Hub uses the time zone of the current user.

   Note: Delivery time frames do not apply to messages which are not related to actions, for example, work object-related notification.
8. **Optional:** In the **Advanced** tab, configure advanced account settings, such as response timeout, character encoding, and reconnect options.
9. **Optional:** In the **Additional settings** tab, specify the name and value for any additional SMPP configuration settings which are not available in the **Setup** and **Advanced** tabs.



Setup
Advanced
Additional settings
Optionally, configure additional SMPP properties here, by providing property name and value
Property name
Property value
unackedQueueSize
5000
Add row

10. Click **OK** and then click **Test** to test your settings.
    The test message is sent to the **Test Number** specified in step 3.
11. **Optional:** If you want SMS and Multi Treatments in actions to be sent from a non-default SMS account, click **Add connection** to add a new SMS account.
    Note: After configuring the new SMS account, select it as the delivery account the Send SMS and Send Multi shapes in the actions where you want to use the non-default SMS account for messaging.

Configuring an inbound SMS account

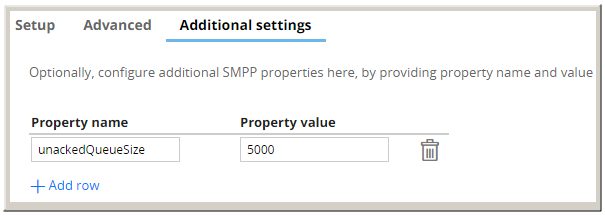
Configure an inbound SMS account which Pega Customer Decision Hub uses to listen for Customer responses to actions, and then specify how to process inbound SMS messages that are received by this inbound SMS account.

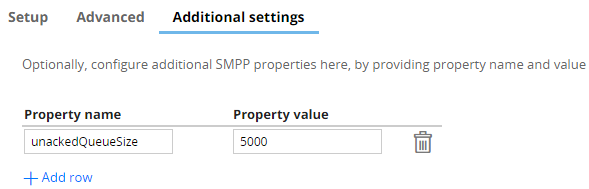
Pega Customer Decision Hub

When the inbound SMS account receives a message, Pega Customer Decision Hub checks the sender's phone number against the Customer table to find the Customer who sent the message. If no matching Customer is found, the message is treated as an error and moved to the *UnhandledInboundSMS* workbasket for further processing.

You can configure the inbound SMS account to parse the SMS text to determine the response, and then associate each response with a different action. In a simpler scenario, you can associate all responses to a particular account with a single action.

1. In App Studio, click **Settings Channels SMS Inbound** .
2. Click **Add connection**.
3. In the **Connection Setup** tab, fill out the required connection details:
   - **Account name** - A unique name for the inbound SMS account, for example, InboundSMSMain. The name must contain only letters and numbers.
   - **Host address** - The address of the SMS center host, for example, my.smsc.com.
   - **Host port** - The port number of the SMS center host, for example, 1041.
   - **Inbox address** - The address of the SMS inbox on which Pega Customer Decision Hub should listen for responses.
   - **IP address** - The IP address of the SMS center host.
   - **User ID** and **Password** - Authentication credentials for the SMS account.
4. **Optional:** In the **Additional settings** tab, specify the name and value for any additional SMPP configuration settings which are not available in the **Connection Setup** tab.





5. In the **Response behaviors** tab, specify how the application should process inbound SMS messages.
   a. **Optional:** Specify what type of messages this inbound SMS account will receive.
   The following options are available:
      - **Messages and delivery receipts** - The SMS account can receive incoming messages, as well as delivery receipts for SMS messages which Pega Customer Decision Hub sends to Customers.
      - **Delivery receipts only** - The SMS account can receive only delivery receipts.

   For more information, see Configuring SMS delivery receipts.
   b. Select how Pega Customer Decision Hub should match received messages to responses:
      - **One response** - Select this option to assign the same action and response value for all SMS messages received on this account.
      - **Decision table** - Select this option to select one or more decision tables which associate the received message with a positive or negative action response. Each result from the decision table can be associated with the same action or with different individual actions.
   c. Configure the necessary settings for matching received messages to responses.
      - If you selected the **One response** option, in the **Responses for inbound messages** section, select an action and a response. Depending on the selected response, all incoming messages are treated as either acceptance or rejection for the selected action.
      - If you selected the **Decision table** option, in the **Manage decision tables** section, configure how the incoming messages should be matched to actions, and then select the decision table for parsing the response. For more information, see Using decision tables to manage SMS action responses.
   d. Specify how Pega Customer Decision Hub should process responses to actions which are not active for a given Customer:
      - **Push it to a workbasket** – Select this option to prevent Customers from responding to actions which are not active for them. Responses to inactive actions are treated as an error and moved to the *UnhandledInboundSMS* workbasket for further processing. You can access this workbasket in the Case Manager portal.
      - **Start the action for the customer** – Select this option to initiate a new action for the Customer if they respond to an action which is not active for them.
6. Click **OK**.
7. Toggle the control in the **Running** column to start listening for incoming messages.
   The **Connection status** column indicates the current status of the account.

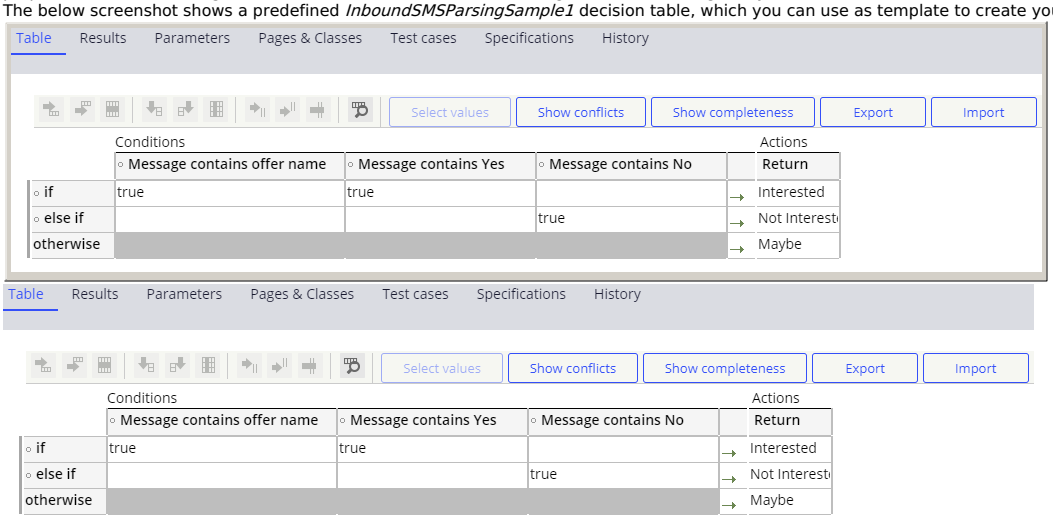
- Using decision tables to manage SMS action responses

Using decision tables to manage SMS action responses

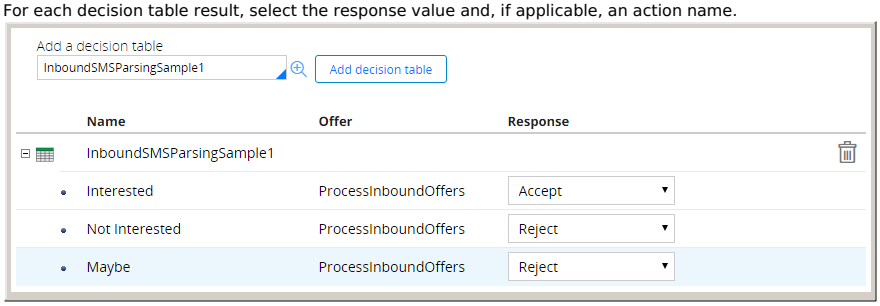
If you selected the **Decision table** option while configuring response behavior for your inbound SMS account, you must select one or more decision tables which associate inbound SMS messages with a positive or negative action response. Each result from the decision table can be associated with the same action or with different individual actions.

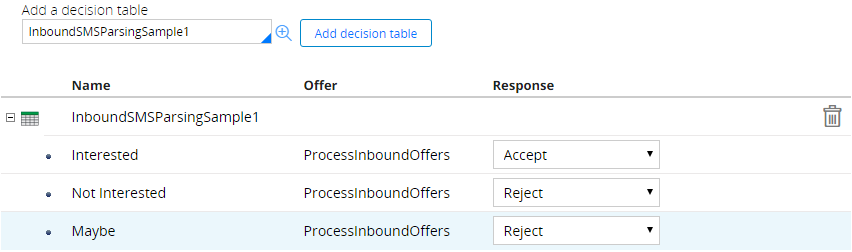
Pega Customer Decision Hub

1. In Dev Studio, configure a decision table in the *PegaMKT-Work-Offer* class. Below are some considerations for accessing message and Customer data in decision tables:
   - Customer data is available through the *.Customer* page property under the action class.
   - Inbound message data is available through the *.InboundResponseData.SMS* page property under the action class. The *.InboundResponseData.SMS* page includes properties for the incoming number (*.SourceNumber*) and the message content (*.MessageBody*).
   - The below screenshot shows a predefined *InboundSMSParsingSample1* decision table, which you can use as template to create your own decision tables.



   Note: If you want to use more than one decision table, do not specify a return value for the **otherwise** row in the decision tables. Leaving the **otherwise** row blank allows Pega Customer Decision Hub to move on to evaluating the next decision table in the list.
2. In the **Manage decision tables** section of the **Response behaviors** tab, select one of the following options:
   - **One action** – Select this option if all messages received on this account should be associated with a single action.
   - **Individual actions** – Select this option if this account receives responses to multiple actions. In this case, specify each action name alongside the response value in the decision table grid.
3. **Optional:** If you selected the **One action** option, use the **Action** field to specify the action with which all incoming messages should be associated.
4. Select a decision table and click **Add decision table**.
5. For each decision table result, select the response value and, if applicable, an action name.





6. **Optional:** Select and add another decision table.
   When an inbound message arrives, Pega Customer Decision Hub evaluates the decision tables from top to bottom until it finds a matching decision result.

Configuring SMS delivery receipts

You can enable delivery receipts for the SMS messages that Pega Customer Decision Hub sends out.

Pega Customer Decision Hub
Note: If you enable SMS delivery receipts, the receipts will be processed and included in a Customer's Interaction History.

1. In App Studio, click **Settings Channels SMS Outbound** .
2. Edit or create an outbound SMS account for which you want to enable delivery receipts.
3. On the Setup tab of the SMS account, select **Track delivery receipt**.
4. Click **Settings Channels SMS Inbound** .
5. Edit or create an inbound SMS account which corresponds to the outbound SMS account where you enabled delivery receipts.
   The inbound account must have the same connection setup data as the outbound account.
6. On the Response behaviors tab, specify what type of messages this inbound SMS account will receive.
   - **Messages and delivery receipts** - The SMS account can receive incoming messages, as well as delivery receipts for SMS messages which Pega Customer Decision Hub sends to Customers.
   - **Delivery receipts only** - The SMS account can receive only delivery receipts.

Troubleshooting Pega Customer Decision Hub

If you encounter issues while working with Pega Customer Decision Hub, refer to this guide for troubleshooting information.

Pega Customer Decision Hub

- Troubleshooting actions and flows
- Troubleshooting campaigns
- Troubleshooting the Context Dictionary
- Troubleshooting Customer Profile Designer
- Troubleshooting Next-Best-Action Designer
- Troubleshooting output templates
- Troubleshooting Paid Media Manager
- Troubleshooting real-time events
- Troubleshooting treatments
- Troubleshooting customer journeys
- Monitoring system health in Pega Customer Decision Hub
- Identifying and reporting issues
- Understanding Pega Customer Decision Hub errors and warnings
- Understanding agent errors and warnings
- Understanding channel errors and warnings
- Understanding configuration errors and warnings
- Understanding runtime errors and warnings

Troubleshooting actions and flows

Use the following troubleshooting tips to help solve issues related to action processing.

Pega Customer Decision Hub

Issue: Accept and reject counts for actions do not update, or actions expire even though a customer has responded

If action accept and reject counts in the Pega Customer Decision Hub portal appear to be incorrect, for example, they do not change even though a customer has accepted an action, verify that the *MKTResponseHandler* data flow is running.

1. Log in to Dev Studio as an operator whose access group points at the *PegaMarketing_FW* application version 8.**Tip:** If required, you can create the required access group by copying the *MKTTemplate:MarketAdmins* template group.
2. In Dev Studio, click **Configure Decisioning Decisions Data flows Real-time processing** .
3. Check the status of the *MKTResponseHandler* data flow.
4. If the flow is not running, in the **Actions** column, click **Manage Restart** .

Troubleshooting campaigns

Use the following troubleshooting tips to help solve issues related to campaign runs.

Pega Customer Decision Hub

Issue: Campaigns that use the same customer segment may fail

Campaigns use customer segments for audience targeting. After a campaign run, segments are refreshed, that is, the membership of a segment is updated based on the outcomes of the campaign strategy. If a segment is used in two or more campaigns that run at the same time, the campaigns may fail because the segment cannot be refreshed.

To solve this issue, schedule the campaign runs so that they do not overlap, or do the following steps to enable transactional segment refresh, which ensures correct segment population in the case of concurrent refreshes:

1. In the Pega Customer Decision Hub portal, click **Audiences Segments** .
2. Edit the segment that is used in parallel campaigns.
3. In the **Options and Schedule** tab of the segment, in the **Data Option**s section, select **Enable Transactional Refresh**.

Issue: After a volume constraint is applied, the campaign does not initiate actions

Campaigns use volume constraints to limits the number of actions that are delivered to customers, and thus prevent action overexposure. When a campaign runs, it first executes the strategy to determine the potential set of actions that should be processed. Next, if a volume constraint has been associated with the campaign, any enabled

action and channel constraints are applied to the action set to determine the final list of actions which needs to be processed.

If after applying the volume constraint the campaign no longer initiates any actions, verify that the strategy that determines the set of actions is configured correctly. For example, ensure that the strategy does not set the *pxPriority* property to a negative value, as *pxPriority* is used as the base weight for an action during constraint run time and does not support negative values.

Issue: In multilevel decisioning with action bundles, the parent action is initiated only once

In environments with multiple context entities, if you configure a campaign that runs at the top level (for example, Account) and sends out action bundles to child context entities (for example, Subscribers), the campaign will initiate a single bundle parent action for bundles with the same bundle name, instead of a parent action for each child context entity. Bundle member actions will be sent to all child context entities.

This is the expected behavior. If you want to use multiple bundles, give each bundle a unique name. For example, you can append the customer ID to the bundle name to generate unique names such as FlexBundleCID-1.

Issue: In environments with multiple context entities and real-time events, campaign configured at a non-primary context level does not trigger actions

In environments with multiple context entities, if you configure a campaign that runs at a non-primary context level (for example, Device) and that uses real-time event triggers, the campaign will run, but the real-time events will not initiate actions.

To solve this issue, do not use event triggers for non-primary customer contexts.

Troubleshooting the Context Dictionary

Use the following troubleshooting tips to help solve issues related to Context Dictionary configuration.

Pega Customer Decision Hub

Issue: "No unlocked rulesets" error appears when saving the Context Dictionary

When saving the Context Dictionary configuration, you may receive an error that says No unlocked rulesets/versions found that are valid for this record. Unlock at least one ruleset/version that can contain records of this type. If unlocking a ruleset does not resolve the error, check that the contact policy rule *CustomerContactPolicy* is available only under the primary customer class of your Pega Customer Decision Hub implementation application:

1. In the Dev Studio search field, search for the contact policy rule *CustomerContactPolicy* that is under the primary customer class of your Pega Customer Decision Hub implementation application.
2. If the *CustomerContactPolicy* rule is available in any other ruleset apart from the one into which you want to save the Context Dictionary configuration, delete the rule manually.
3. In App Studio, save the Context Dictionary configuration.

Troubleshooting Customer Profile Designer

Use the following troubleshooting tips to help solve issues related to Customer Profile Designer.

Pega Customer Decision Hub

Issue: The number of allowed summaries and data aggregates is not sufficient

By default, you can define 3 summaries with 120 data aggregates per summary. If that is not sufficient, you can modify the following dynamic system settings in the *PegaMKT-Engine* ruleset:

- *cpd/summaries_dataset/max_aggregates_number*
- *cpd/designer/max_summary_associations_number*

Caution: Adding additional aggregates and summaries has an impact on the performance of Pega Customer Decision Hub. If you change the shipped values, you must run your own performance tests to determine the impact of this change on your Service Level Agreements.

Issue: The summary is not created when you create a Summary rule

If the summary is not created when you create Summary rules on the **Profile Data Sources** landing page, check that the backing class can be created.

Class names in Pega Customer Decision Hub can only have a maximum of 56 characters, and the Summary rule creation creates a corresponding class that is a pattern-inherited child class of the selected data source class. Because of that, if your data source class is already 45 characters long, the summary name can be no longer than 10 characters, as the - character in the class name takes 1 character. If the summary name exceeds the number of available characters, the backing class for the summary is not created.

Issue: Aggregation does not work correctly

Some internal keywords that cannot be used as aggregate properties. If the aggregation does not work as expected, verify that *EventType* is not used as an aggregate property.

Issue: The integration service is not picked up

If your service is not getting picked up, verify that your service package has an appropriate Service access group defined.

Issue: The service is running but data flows are not executed

If the service is running but you data flows are not executed on your system, check that nodes are allocated to the decisioning infrastructure, especially for the Decision Data Store, Adaptive Decision Manager, and Batch data flows. If you opt to use a Stream service, ensure there is also at least one node dedicated to Stream.

Issue: Locked rulesets error

All operations available in the Customer Profile Designer, such as summary creations, adding data sources, and associating summaries or data sources with a customer context, require an unlocked ruleset version of the *<application name>-rules-<application version>* rule. If there is no unlocked ruleset version for the rule, the Customer Profile Designer shows an error. The error can also occur when the unlocked ruleset version is not the latest available version (that is, if the current ruleset version is locked, but an older unlocked version is available).

To solve this issue, do the following steps:

1. Log in to your Pega Customer Decision Hub application as an operator with access to Dev Studio.
2. In Dev Studio, click the name of your application, and then click **Definition**.
3. Click the **Configure** icon by the *<application name>-rules-<application version>* ruleset.
4. Create a new version of the ruleset.

Tip: Always ensure that the unlocked ruleset version of *<application name>-rules-<application version>* is the newest version available.

Issue: After deleting a summary or removing a data source, the deleted item is still visible

If a summary or data source that you deleted is still displayed in Customer Profile Designer, and then the user finds that the same summary or data source still exists on CPD, it may be because the deleted item was originally created in a *<application name>-rules-<application version>* ruleset version that has since been locked.

For example, if you create a summary in the ruleset version *uplusbank-rules-02*, lock the ruleset and create the ruleset version *uplusbank-rules-03*, and then update the summary, the summary is automatically copied to the unlocked ruleset *uplusbank-rules-03*. However, an older version of the summary still exists in the locked ruleset *uplusbank-rules-02*, and is visible in Customer Profile Designer.

If you must delete a summary created in an older version of the ruleset, unlock the ruleset where the summary or data source was originally created:

1. Log in to your Pega Customer Decision Hub application as an operator with access to Dev Studio.
2. In Dev Studio, click the name of your application, and then click **Definition**.
3. Unlock the version of the *<application name>-rules-<application version>* ruleset where you originally created the summary or data source.
4. Delete the summary or data source in the unlocked ruleset.

Issue: Cannot find a datasource on the Profile Data Sources landing page

If you cannot find a specific data source in the list of available data sources on the **Profile Data Sources** landing page, ensure that the data source is available as part of the current application. Only data sources that are available within the application are visible on the **Profile Data Sources** page. **Note:** Interaction Stream and Interaction History data sources are automatically filtered out, and are not available for selection on the **Profile Data Sources** page.

Issue: Trying to save the customer profile results in an error

After upgrading your Pega Customer Decision Hub application from a version earlier than 8.6, you may encounter an issue while trying to save the customer profile for the first time. The issue occurs if you have previously manually changed the *contextflow* data flow before the upgrade, for example, by adding or removing any flow shapes.

To solve this issue, revert the manual changes to the data flow:

1. Log in to your Pega Customer Decision Hub application as an operator with access to Dev Studio.
2. In Dev Studio, open the class of the context where you edited the *contextflow* data flow.
3. Click **Data Model Data Flow** .
4. Open the *contextflow* data flow and check the current configuration.
5. In Pega Customer Decision Hub, click **Data Profile Data Sources** .
6. Add the data sources that were manually added or removed before the upgrade.
7. Click **Data Profile Data Sources** .
8. Associate any data sources that were manually added before the upgrade with a customer context.
9. Remove the association of any data sources that were manually removed before the upgrade.
10. Click **Save**.

Issue: Cannot find the data source or summary when trying to add new associations

If you cannot find a specific summary or data source in the list of available associated data in the **Customer Profile Designer**, check if it has not yet been associated with a customer context. After a summary or data source has been associated with a context entity, you cannot associate it with any other customer contexts.

Issue: Cannot modify an aggregate

If you create a summary aggregate in a specific version of the *<application name>-rules-<application version>* ruleset, and then try to edit it after locking the ruleset version where it was originally created, saving the edited aggregate may result in an error.**Note:** The error will occur if changing the aggregate results in a new property type, for example, if the aggregate property had the type `Double`, and was then changed to `Integer`.

To solve this issue, save the edited aggregate under a new unique name.

Issue: After modifying a summary, the aggregation service restarts

For summaries of stream data sources, if you edit, add, or remove a summary aggregate, or if you change the **Group by** keys for the summary, the aggregation process restarts.

This is the expected behavior. Aggregation for all the defined aggregates in the edited summary restarts from the last event that the aggregation service processed before you modified the summary.

Troubleshooting Next-Best-Action Designer

Use the following troubleshooting tips to help solve issues related to Next-Best-Action Designer configuration.

Pega Customer Decision Hub

Issue: Upgrade to Next-Best-Action Designer version 2 fails

If you have previously configured next best action using Next-Best-Action Designer version 1, upgrading to Next-Best-Action Designer version 2 is normally a seamless process. However, in some cases the process can fail due to the presence of older *NBA Config* and *NBA Model* rules in your system. If this happens, and you are not able to delete these rules directly from the Next-Best-Action Designer landing page, perform the following steps to find the rules in your system and manually delete them:

1. In Dev Studio, edit the class *Rule-PegaMKT-NBA-Model*.
2. List all the instances to find the rule that you want to delete.
3. Write an activity to delete the rule instance based on the *pzInsKey* property.
4. Edit the class *Rule-PegaMKT-NBA-Config*.
5. List all the instances to find the rule that you want to delete.
6. Open the rule and click **Delete**.

Issue: "Exception in stage: Trigger_NBA_TopLevel" error appears when running an outbound schedule or container

After an upgrade to Next-Best-Action Designer version 2, if you try to run either an outbound schedule or a real-time container, the run may fail with the error message Exception in stage: Trigger_NBA_TopLevel. To resolve the error, re-save the Next-Best-Action Designer taxonomy:

1. In App Studio, click **Next-Best-Action Designer Taxonomy** .
2. Click **Edit**.
3. Click **Save**.
4. Run the outbound schedule or real-time container again.

Issue: After editing and saving a Supporting Decision rule, some custom Strategy Result properties generate empty values

When you re-save a Support Decision rule, the associated real-time container data flow is recreated with the **Output selected properties** option enabled on the Strategy shape. If you do not select the custom Strategy Result properties, those property values generate empty values in the response.

To solve this issue, edit the real-time container data flow and enable the **Output all properties** on the Strategy shape.

Troubleshooting output templates

Use the following troubleshooting tips to help solve issues related to output template processing.

Pega Customer Decision Hub

Issue: After updating to version 8.6, files generated from output templates are no longer recognized by downstream processing logic

File output templates export data out of Pega Customer Decision Hub in the form of files which can be imported into another tool. Before version 8.6, the file created from a template used the name specified in the template configuration, without a unique identifier.

Starting in Pega Platform 8.6.3, each file name has a unique identifier, automatically generated based on the data flow node, thread ID, and timestamp. Because of that, if your process to consume output files expects files with a specific name, it may not be able to process the resulting files correctly.

If you have configured the process before updating to Pega Customer Decision Hub version 8.6, but the exported files are no longer recognized by downstream processing logic after the update, ensure that the downstream tool is configured to recognize the files by a pattern rather than the full name. For example, when referring to files exported to the repository, use the * character to indicate a pattern instead of using the full file name. For example, use `Export*.csv` to refer to the files.

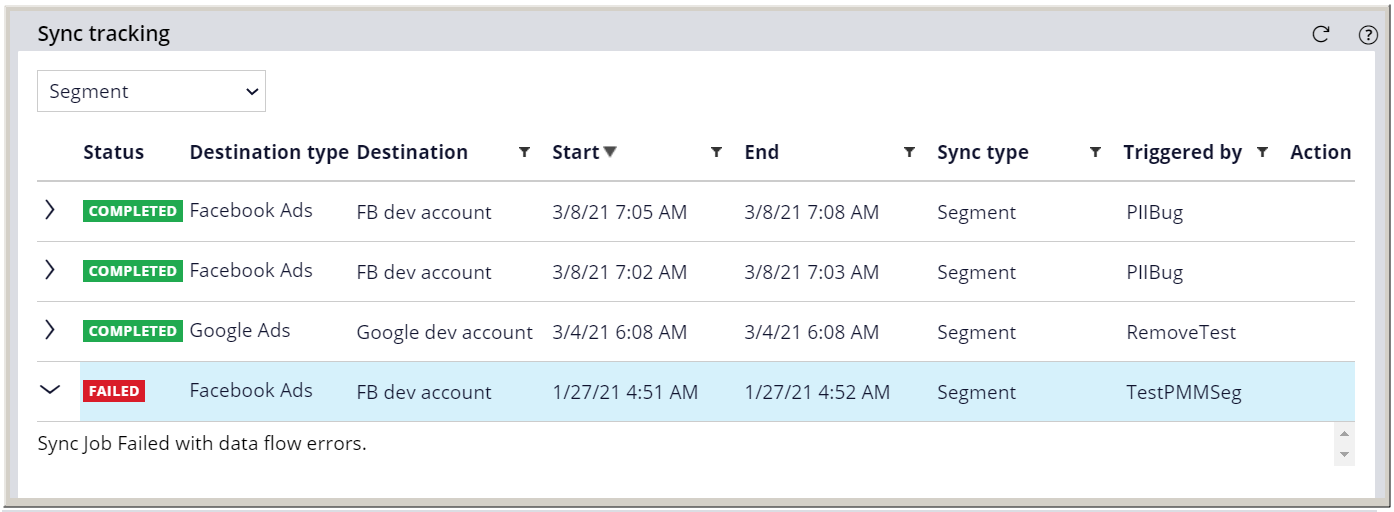
Troubleshooting Paid Media Manager

Use the following troubleshooting tips to help solve issues related to Paid Media Manager configuration.

Pega Customer Decision Hub

Issue: Paid synchronization job fails with the message "Sync Job Failed with data flow errors"

If a synchronization job fails due to a data flow error, the Sync Tracking tab of the **Paid Media** landing page displays the message `Sync Job Failed with data flow errors`, as in the following figure:



Sync tracking
Segment
Status
Destination type
Destination
Start
End
Sync type
Triggered by
Action
COMPLETED
Facebook Ads
FB dev account
3/8/21 7:05 AM
3/8/21 7:08 AM
Segment
PIIBug
COMPLETED
Facebook Ads
FB dev account
3/8/21 7:02 AM
3/8/21 7:03 AM
Segment
PIIBug
COMPLETED
Google Ads
Google dev account
3/4/21 6:08 AM
3/4/21 6:08 AM
Segment
RemoveTest
FAILED
Facebook Ads
FB dev account
1/27/21 4:51 AM
1/27/21 4:52 AM
Segment
TestPMMSeg
Sync Job Failed with data flow errors.

To solve this issue, do the following steps:

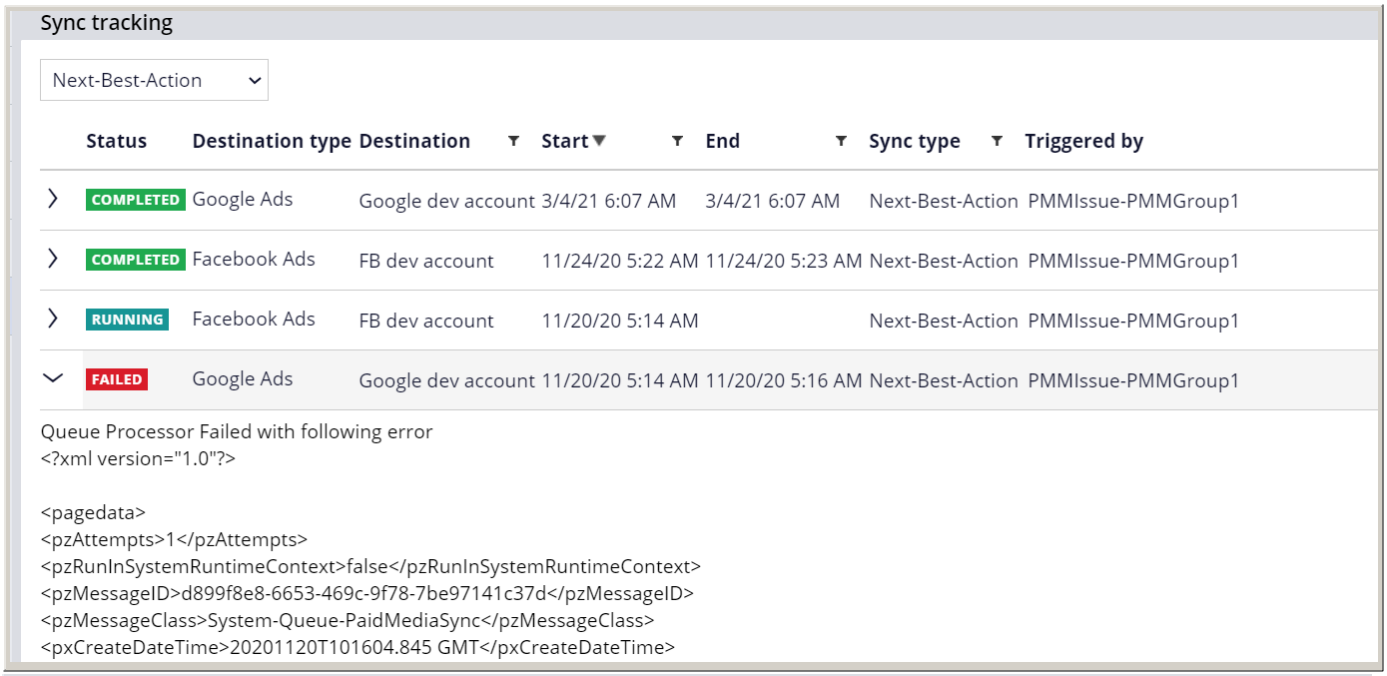
1. In Dev Studio, click **Configure Decisioning Data flow Batch** .
2. Look for the failed data flow and diagnose the error.
3. After diagnosing and resolving the error cause, rerun the paid run, for example, by re-executing the segment run or outbound run, or by waiting for their next scheduled run.

Issue: Paid synchronization job fails with the message "Queue processor failed with following error"

If a synchronization job fails due to a queue processor error, the Sync Tracking tab of the **Paid Media** landing page displays the message `Queue processor failed with following error`.

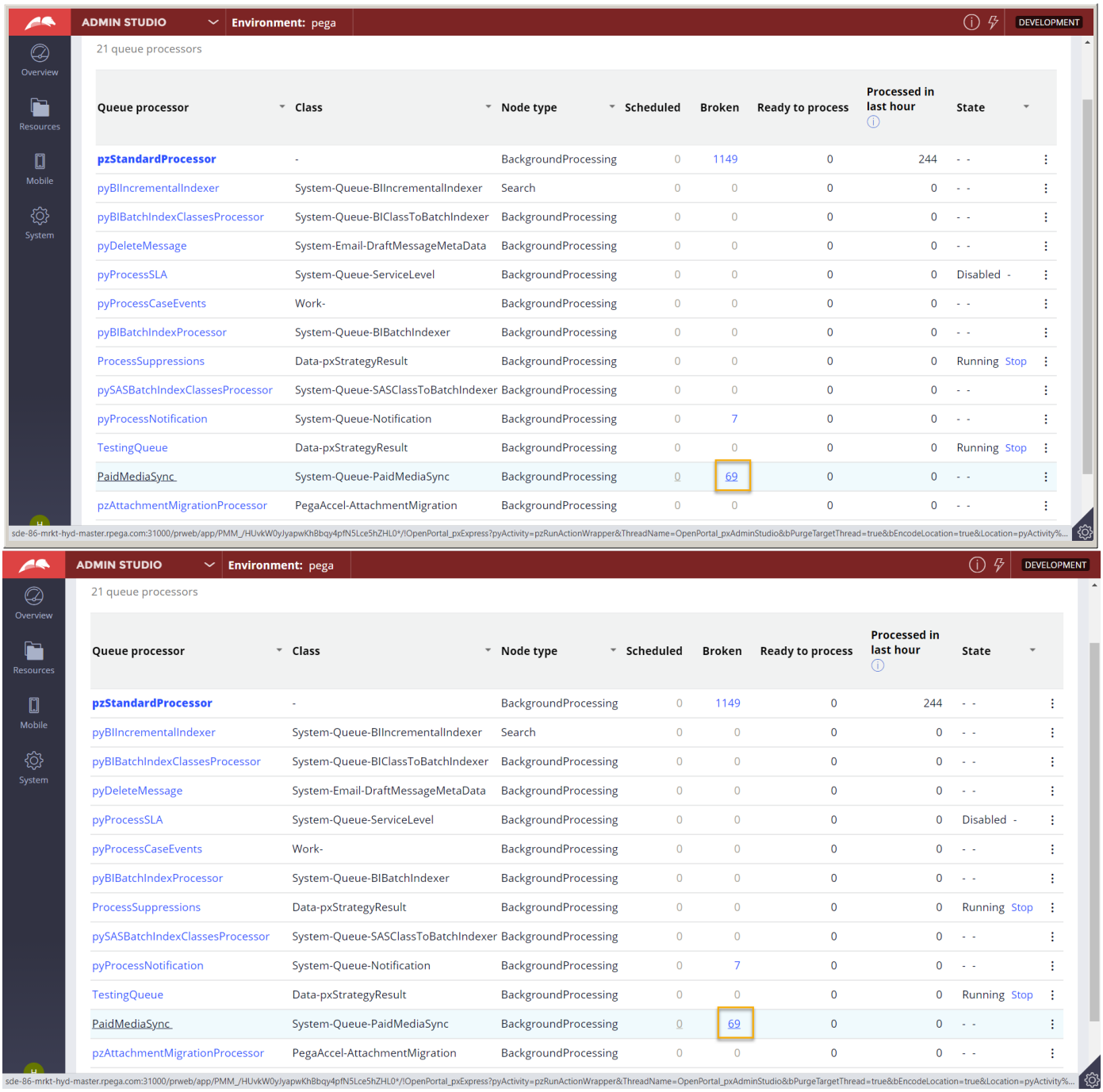
To solve this issue, do the following steps:

1. In App Studio, click **Settings Channels Paid Media** .
2. Click the arrow by the failed synchronization job to view the error details, as in the following figure:



Sync tracking
Next-Best-Action
Status
Destination type
Destination
Start
End
Sync type
Triggered by
COMPLETED
Google Ads
Google dev account
3/4/21 6:07 AM
3/4/21 6:07 AM
Next-Best-Action
PMMIssue-PMMGroup1
COMPLETED
Facebook Ads
FB dev account
11/24/20 5:22 AM
11/24/20 5:23 AM
Next-Best-Action
PMMIssue-PMMGroup1
RUNNING
Facebook Ads
FB dev account
11/20/20 5:14 AM
Next-Best-Action
PMMIssue-PMMGroup1
FAILED
Google Ads
Google dev account
11/20/20 5:14 AM
11/20/20 5:16 AM
Next-Best-Action
PMMIssue-PMMGroup1
Queue Processor Failed with following error
<?xml version="1.0"?>
<pagedata>
<pzAttempts>1</pzAttempts>
<pzRunInSystemRuntimeContext>false</pzRunInSystemRuntimeContext>
<pzMessageID>d899f8e8-6653-469c-9f78-7be97141c37d</pzMessageID>
<pzMessageClass>System-Queue-PaidMediaSync</pzMessageClass>
<pxCreateDateTime>20201120T101604.845 GMT</pxCreateDateTime>

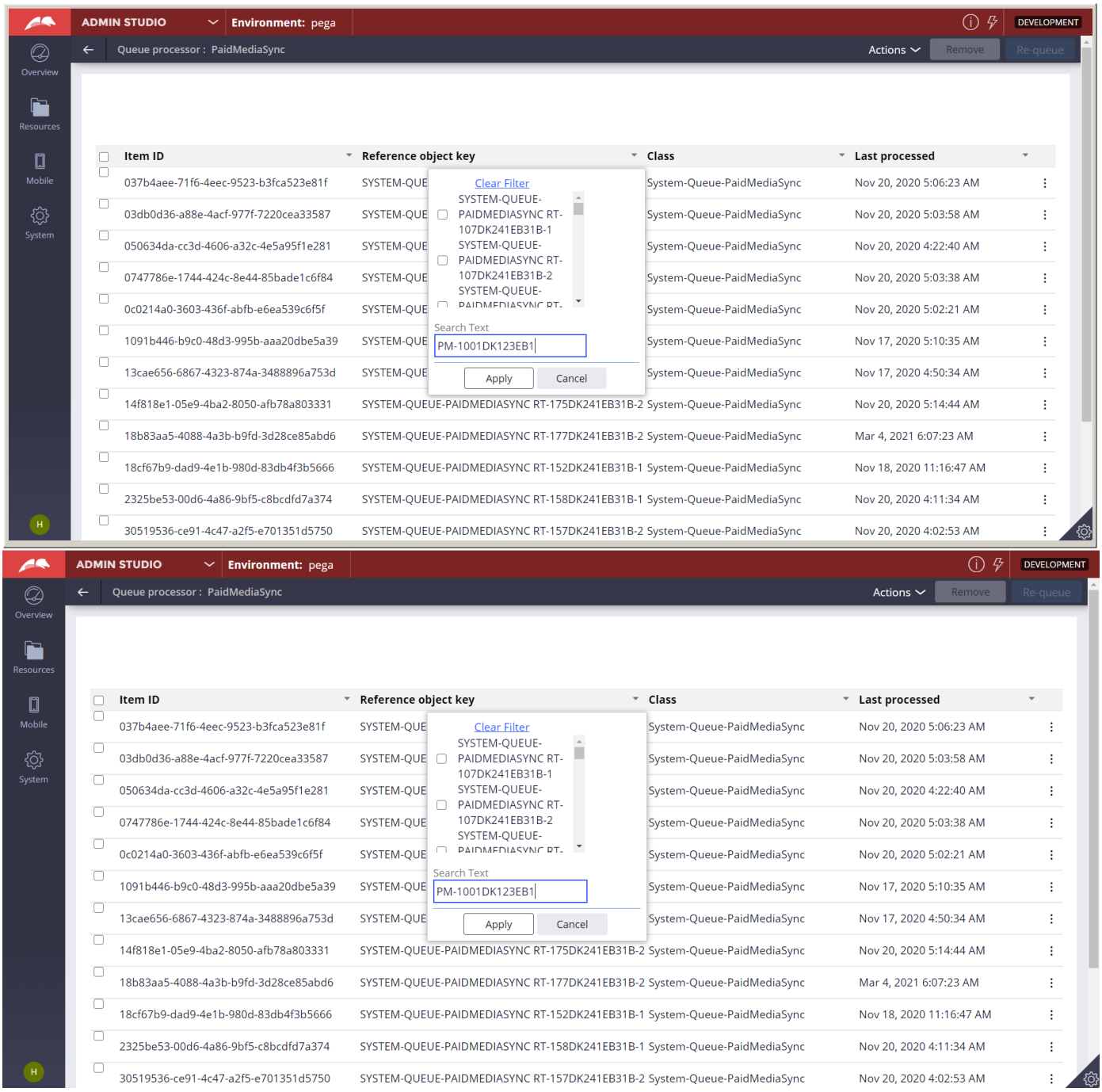
3. At the bottom of the XML section, find and note down the *JobID* and *DestinationKey*, for example, PM-1001 and DK123EB.
4. Switch to Admin Studio.
5. Click **Resources Queue processors** .
6. In the list of queue processors, find PaidMediaSync, and click the number in the **Broken** column, as in the following figure:



ADMIN STUDIO
Environment: pega
DEVELOPMENT
Overview
Resources
Mobile
System
21 queue processors
Queue processor
Class
Node type
Scheduled
Broken
Ready to process
Processed in last hour
State
pzStandardProcessor - BackgroundProcessing 0 1149 0 244 - -
pyBIIncrementalIndexer System-Queue-BIIncrementalIndexer Search 0 0 0 0 - -
pyBIBatchIndexClassesProcessor System-Queue-BIClassToBatchIndexer BackgroundProcessing 0 0 0 0 - -
pyDeleteMessage System-Email-DraftMessageMetaData BackgroundProcessing 0 0 0 0 - -
pyProcessSLA System-Queue-ServiceLevel BackgroundProcessing 0 0 0 0 Disabled -
pyProcessCaseEvents Work- BackgroundProcessing 0 0 0 0 - -
pyBIBatchIndexProcessor System-Queue-BIBatchIndexer BackgroundProcessing 0 0 0 0 - -
ProcessSuppressions Data-pxStrategyResult BackgroundProcessing 0 0 0 0 Running Stop
pySASBatchIndexClassesProcessor System-Queue-SASClassToBatchIndexer BackgroundProcessing 0 0 0 0 - -
pyProcessNotification System-Queue-Notification BackgroundProcessing 0 7 0 0 - -
TestingQueue Data-pxStrategyResult BackgroundProcessing 0 0 0 0 Running Stop
PaidMediaSync System-Queue-PaidMediaSync BackgroundProcessing 0 69 0 0 - -
pzAttachmentMigrationProcessor PegaAccel-AttachmentMigration BackgroundProcessing 0 0 0 0 - -
sde-86-mrkt-hyd-master.rpega.com:31000/prweb/app/PMM_/HUvkW0yJyapwKhBbqy4pfN5Lce5hZHL0*/!OpenPortal_pxExpress?pyActivity=pzRunActionWrapper&ThreadName=OpenPortal_pxAdminStudio&bPurgeTargetThread=true&bEncodeLocation=true&Location=pyActivity%...

7. Filter the **Reference object key** column by entering the concatenated *JobID* and *DestinationKey* that you noted in step 3.

   For example, if the *JobID* is PM-1001 and the *DestinationKey* is DK123EB1, enter PM-1001DK123EB1, as in the following figure:



8. Click the **More** icon by the failed synchronization item to view the full error details.**Tip:**

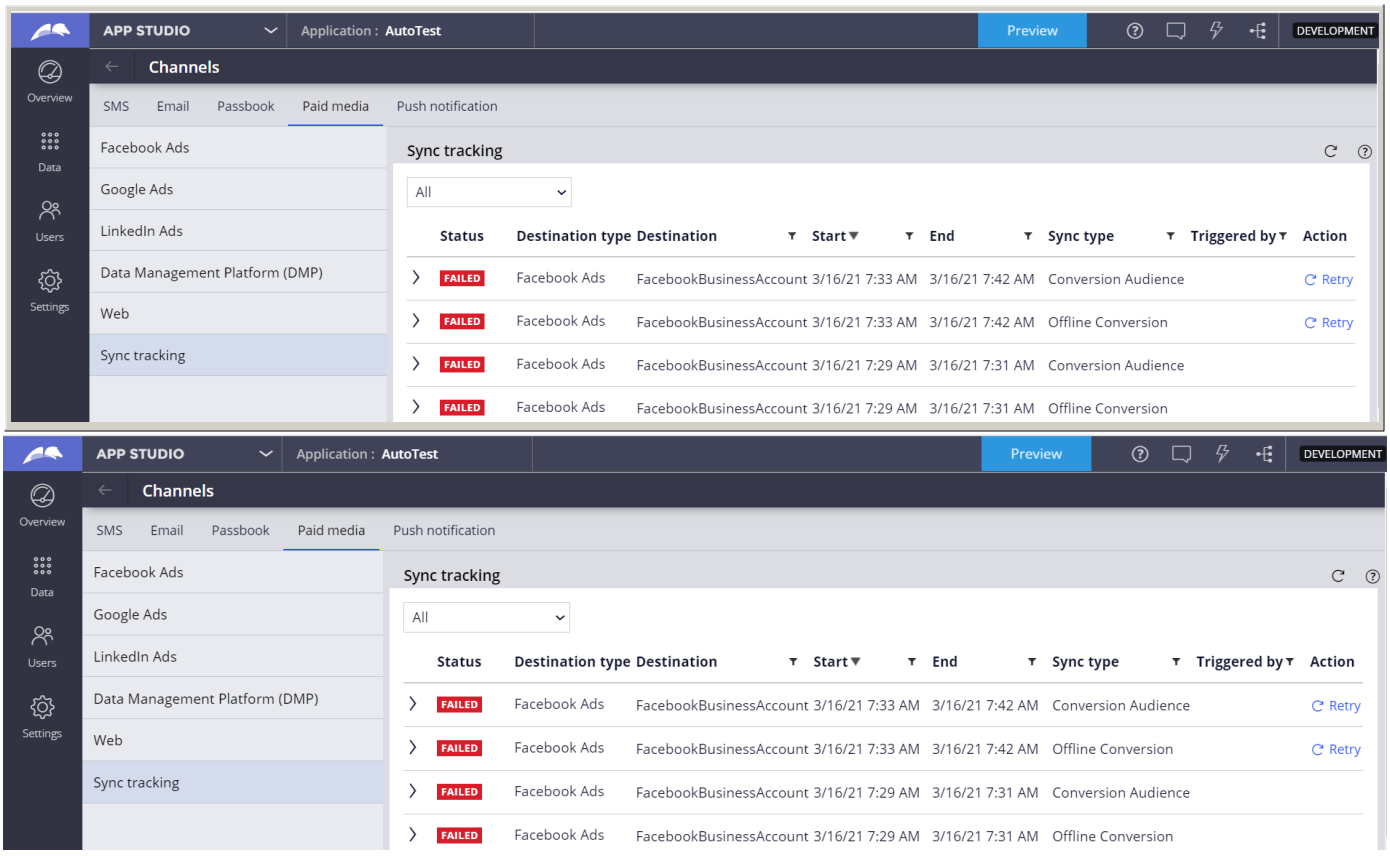
   When analyzing the error details, look for the Caused by section at the bottom of the error log. This section should contain a message that describes the cause of the error. For example, the following error message indicates that a paid synchronization run has failed due to an expired access token for a specific Google Ads paid destination:

   Caused by: com.pega.mkt.paidmedia.sync.AudienceSyncException@42ebe857[paidMediaErrors=[com.pega.mkt.paidmedia.error.PaidMediaError@14147b5e[paidRunId=RT-176,dataFlowRunId=N/A,code=GG-AE-004,message=The access token in the request header is either invalid or has expired., severity=CRITICAL,occurredOn=Fri Nov 20 05:16:04 EST 2020,platform=Google,destination=XXX-YYY-ZZZZ,syncType=REALTIME]]]

   At the bottom of the error log you can see the following additional information to help you discover which specific paid media run is associated with the error:

   JobID
   Unique identifier of a specific paid media run.
   DestinationKey
   Unique identifier of a paid destination
   SyncType
   Type of paid run, for example, Segment, Outbound, or Real-time.

9. After diagnosing and resolving the error cause, in App Studio, click **Settings Channels Paid Media** , and then click **Retry** by the failed synchronization item, as in the following figure:



Troubleshooting real-time events

Use the following troubleshooting tips to help solve issues related to real-time events processing.

Pega Customer Decision Hub

Issue: Data flow triggered by real-time events fails with the error "Could not find pxInteractionHistoryDataSet Data-set"

Using a real-time event to trigger single-case processing data flows (that is, data flows where the source and destination are configured as abstract) may result in the error Could not find pxInteractionHistoryDataSet Data-set. To solve the issue, re-validate the *DFWriteEventTriggerIH* data flow under the *Data-Decision-Request* class.

1. In Dev Studio, click **Configure System Release Upgrade Validate** .
2. Click **Revalidate and save**.
3. In the **Type** list, select **Rule-Decision-DDF**.
4. In the **RuleSet Name** list, select **PegaMKT-Engine**.
5. In the **RuleSet Version** list, select the rule set that contains the *DFWriteEventTriggerIH* data flow.
6. Click **List**.
7. Clear all check boxes apart from the check box for the *DFWriteEventTriggerIH* data flow.
8. Click **Run**.

Troubleshooting treatments

Use the following troubleshooting tips to help solve issues related to treatments.

Pega Customer Decision Hub

Issue: Missing styling for table or grid structures sourced from an embedded section in an email treatment

If the email treatment contains a section with a repeating grid layout, the section is rendered without styling when the email is received. To solve this issue, include the required styling for the section in the **Inline style** field on the section.

1. Click the grid layout and in the header click the **View properties** icon.
2. In the **Properties** window, click the **Presentation** tab.
3. To define inline style for the layout, enter CSS code in the **Inline style (not for production use)** field.**Note:** Entering inline style results in a guardrail warning. You can justify the warning by noting that, for an email sent externally, the email does not have access to any runtime CSS from Pega Platform. Because of that, any styling must be contained within the email HTML.

Troubleshooting customer journeys

The purpose of this section is to outline common customer journey troubleshooting scenarios.

Pega Customer Decision Hub

You might encounter the following issues:

- Issue: Actions are missing journey data
- Issue: Journey actions not available in the Next-Best-Action strategy framework output
- Issue: Customer journey results may not be appearing as expected in the Journey Visualizer
- Issue: Data sets that need to be reset
- Issue: The same stage is being output for each interaction
- Issue: The interaction did not get the stage that was anticipated
- Issue: Customer journey stage criteria are missing
- Issue: Next-Best-Action template support

- Issue: Actions are missing journey data
- Issue: Journey actions not available in the Next-Best-Action strategy framework output
- Issue: Customer journey results may not be appearing as expected in the Journey Visualizer
- Issue: Data sets that need to be reset
- Issue: The same stage is being output for each interaction
- Issue: The interaction did not get the stage that was anticipated
- Issue: Customer journey stage criteria are missing
- Issue: Next-Best-Action template support

Issue: Actions are missing journey data

Use the following troubleshooting tips to help solve issues with actions that are missing journey data.

You might encounter the following issues:

- Cause: The action is not active
- Cause: The action does not have an association with a journey and stage
- Cause: JourneyActions Decision Data Rule is misconfigured
- Cause: Decision Data Rules are missing form fields
- Cause: Treatment configuration
- Cause: Contact Policy suppression

Pega Customer Decision Hub

Cause: The action is not active

An action must be active in order to be included in the Next-Best-Action customer journey processing.

Solution: Check if the action is active

1. Open the action for which the journey processing is intended.
2. Navigate to the **Details** tab of the rule.
3. Click **Check out** .
4. Set the **Availability** field to **Always** or **Within a defined time period**,and then set a date and time range that is applicable for processing.
5. Click **Save**, and then **Check in**.

Cause: The action does not have an association with a journey and stage

To be processed as part of a customer journey, at least one journey and stage needs to be associated with the action.

Solution: Check if the actions are associated with the journey and stage

1. Open the action for which the journey processing is intended.
2. Navigate to the **Engagement policy** tab of the rule.
3. Click **Check out** .
4. Add the intended journey and stage combination by clicking the **Add customer journey stage**, and then select values from the drop-down list. You can add more than one journey and stage combination to the action.
5. Click **Save**, and then click **Check in**.

Cause: *JourneyActions* Decision Data Rule is misconfigured

The associations between an action and a journey-stage are backed by the *JourneyActions* decision data rule. If this rule does not exist in the default destination ruleset, or it has not been updated with the desired journey-stage association, the action will not be included in the customer journey processing.

Solution: Check the *JourneyActions* DDR configuration

1. Open the **JourneyActions** decision data rule in the default destination ruleset.
2. Navigate to the **Data** tab of the rule.
3. Search through the list of associations to find the **Name** and **Description** of the expected action. If it is not there, the association was not made correctly. See the second scenario to add the association by using the action rule.
4. If the action **Name** link is there, open the **Clipboard** by using the link in the footer.
5. Expand the **RH_1** page on the left side of the **Clipboard**.
6. Expand the sub-page **pxPage** and then the **pxResults** sub-page .
7. Find the **pxResults** page that matches the association by *index* and click on it.
8. In the panel on the right side, the value for *pyName* should match the action name.
9. The values for *pyJourney* and *pyStage* should match those associated with the action rule, as well as the journey configuration on the **Engagement policy** tab of Next-Best-Action Designer.
10. If the values are all correct, there may be a different reason why there is no data for the action in the journey. If any of these values are not correct, then the intended association was not made correctly. See the second scenario to add the association by using the action rule.

Cause: Decision Data Rules are missing form fields

The Decision Data Rules (DDR) for actions in the journey stage configuration may not have *pyJourney* and *pyStage* defined as part of their form fields. To process interactions for actions associated with a specified journey and stage, the DDR for the group that contains that action must include *pyJourney* and *pyStage* as form fields.

Solution: Check if the DDRs have *pyJourney* and *pyStage* form fields configured

1. Open the DDR that shares the name of the group in which the relevant journey-associated action is included. **For example:** If an action is in the Sales group, search for and open the *Sales* DDR.
2. Navigate to the **Form** tab of the rule.
3. Click **Check out**.
4. Add form fields for the *pyJourney* and *pyStage* properties.
5. Click **Save**, and then click **Check in**.
6. Repeat this process for any groups that contain at least one action associated with a journey. **Result:** If all other criteria passed, the next applicable action-based interaction should appear as expected.

Cause: Treatment configuration

The action may not have been processed over the expected channel. The metrics displayed in the Journey Visualizer overlay are driven by a specific set of outcomes, based on the name of the channel.

The following table lists the metrics displayed based on the shipped channels:

Metric	Channels
Impression count	All channels
Click Through rate (Outbound)	Email, SMS

Click Through rate (Inbound)	CallCenter, Microsite, Web
Conversion rate (Outbound)	Email, SMS
Conversion rate (Inbound)	CallCenter, Microsite, Web
Volume	All other channels
Clicked count	All other channels

Solution: Use an action

1. Open the action for which journey processing is intended.
2. Navigate to the **Treatments** tab.
3. Click **Check out**.
4. If the intended channel does not have a treatment defined in the list, add one or more treatments using the **Add channel** button.
   If none exist, you can create a treatment from the selector by clicking **Create**.
5. Click **Apply** on the selector when all the treatments have been selected for that channel.
6. If the intended channel has at least one treatment defined in the list, ensure that the channel is enabled for the action by turning on the **Channel** switch.
7. Click **Save**, and then click **Check in**.

Solution: Use Next-Best-Action Designer

1. In **Next-Best-Action Designer** , click **Channels**.
2. Locate the desired channel from the list of available channels. If the **Channel** switch is toggled to the left, the channel is not enabled. In this case, click **Edit**, and then turn on the **Channel** switch.
3. Click the **Save** button.

Cause: Contact Policy suppression

Contact policies may be suppressing actions. Contact policies provide a mechanism for controlling and restricting how many times a customer is contacted in accordance with corporate or regulatory guidelines. If the action is being suppressed, it will not be included in customer journey processing. For more information regarding Contact policies, see Setting constraints, contact policy limits, and controls.

Issue: Journey actions not available in the Next-Best-Action strategy framework output

Use the following troubleshooting tips to help solve issues with actions that are not available in the Next-Best-Action strategy framework output.

You might encounter the following issues:

- Cause: DDR entries are not correctly stored
- Cause: Inactive actions are being filtered out
- Cause: Engagement policies conflict with user data
- Cause: Constraints are restricting output
- Cause: Entry criteria for a customer journey stage are too restrictive
- Check that the entry criteria for subsequent journey stages are not too permissive
- Check that the action has treatments associated with it
- Check whether outbound model maturity is enabled
- Confirm whether the journey actions are being correctly output by Next-Best-Action Designer

Pega Customer Decision Hub

Cause: DDR entries are not correctly stored

Ensure that the necessary DDR entries have been correctly stored.

Solution: Check the *JourneyActions* and *JourneyStages* Decision Data Rules

1. Open the *JourneyStages* Decision Data Rule in the default destination ruleset.
2. Find an entry for the journey stage with the format [*JourneyName*]/[*StageName*].
3. Locate the entry for the stage that you want to check on the clipboard, by navigating to the corresponding entry in the RH_1.pxPage.pxResults() list.
   Make a note of the values of *ModelName*, *pyJourney*, and *pyStage* – you will use these to compare with the entries in *JourneyActions*.**Note:** *ModelName* should always be NBA.
4. Check the action which has been associated in the *JourneyActions* DDR, by searching for the correct entry in the RH_1.pxPage.pxResults() list, and find the values of *ModelName*, *pyJourney*, *pyStage*, *pyName*, *pyIssue*, and *pyGroup*.
5. The first three values must exactly match the entry in *JourneyStages* for the association to be considered valid.

Cause: Inactive actions are being filtered out

Inactive actions will always be filtered out by the strategy framework.

Solution: Check that the action is active in the system

1. Open the action for which the journey processing is intended.
2. Click the **Details** tab of the rule.
3. Click **Check out** .
4. Check the value of *Availability* to confirm whether the action is active.
5. Click **Save**, and then click **Check in**.

Cause: Engagement policies conflict with user data

Remember that engagement policies are applied from the most broad to most specific context. Confirm that the defined engagement policies are not in conflict with your defined user data.

Solution: Check that engagement policies and contact policies are not filtering the action out

Policies are applied in this order:

1. The All groups engagement policies in Next-Best-Action Designer.
2. The Group-level engagement policies in Next-Best-Action Designer.
3. The Action-level engagement policies on the action rule form.

Cause: Constraints are restricting output

Solution: Check that constraints are not restricting output on the given channel for the user

1. In the navigation pane of Pega Customer Decision Hub, click **Next-Best-Action Designer** .
2. Click **Constraints**.

3. Analyze the **Customer contact limits** section to confirm that the customer has not already reached their communication limit on the expected channel.

Cause: Entry criteria for a customer journey stage are too restrictive

Solution: Check whether the entry criteria for the customer journey stage are not too restrictive

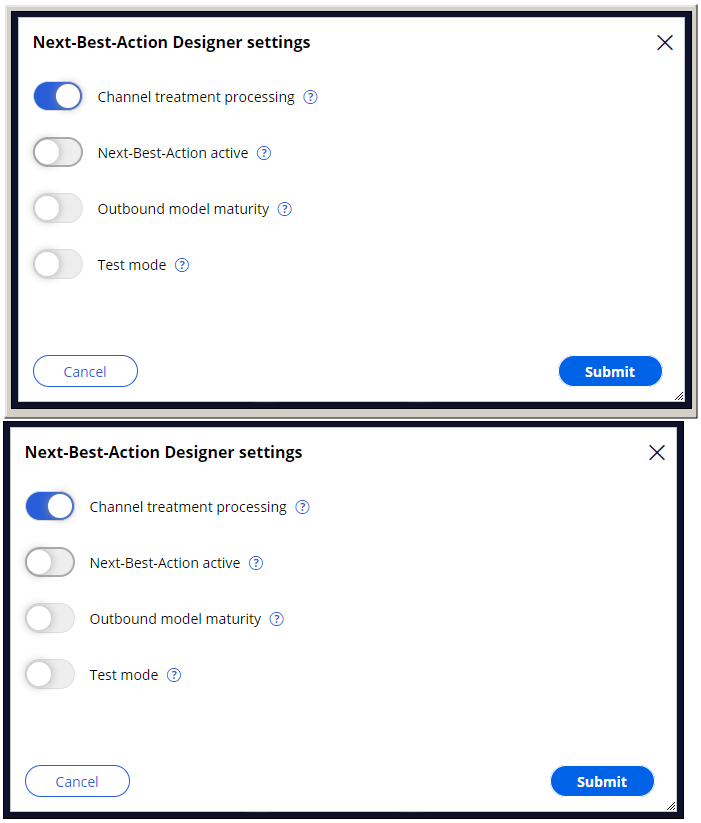
1. In the navigation pane of Pega Customer Decision Hub, click **Next-Best-Action Designer** .
2. Click **Engagement Policy**.
3. Select the customer journey that you want to check from the **Business structure** section.
4. On the specific stage, click the **More** icon.
5. Choose **View entry criteria**.
6. Analyze the criteria and make sure that they are not too restrictive.

Check that the entry criteria for subsequent journey stages are not too permissive

Similarly to the previous scenario, review whether the entry criteria for later stages will allow the user data to be considered for that stage. The strategy framework, within a specific journey, will try to place users in the latest stage for which they are eligible. That means, they may begin the journey midway through your defined stages if the entry criteria are too permissive for later stages. A common misconfiguration is to have very permissive entry criteria on the final stage of the journey, which will mean that users automatically end up at the end of the journey first.

Check that the action has treatments associated with it

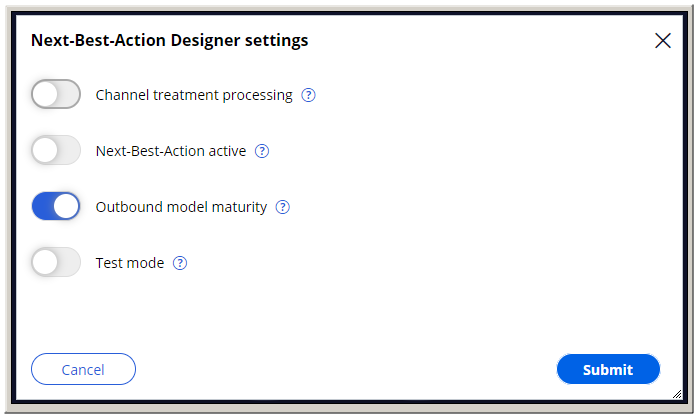
If outbound treatment processing is enabled, the action needs to have treatments associated with it to be considered for output. The value for outbound treatment processing can be checked by clicking the **Properties** icon in the header of Next-Best-Action Designer. See the following figure for reference:

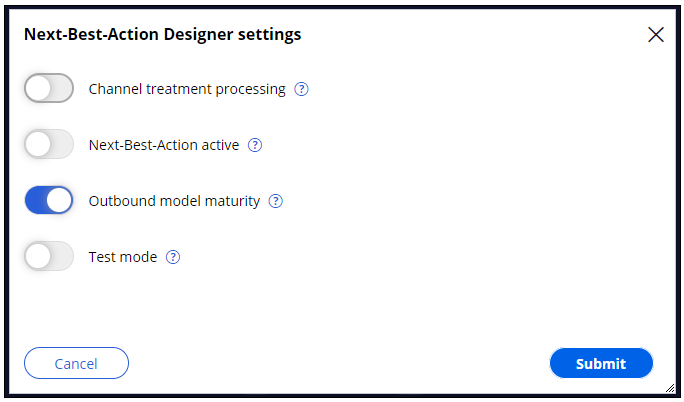


Check whether outbound model maturity is enabled

If the action or environment is relatively new, check whether outbound model maturity has been enabled. This can prevent new actions from being output by the strategy framework for the majority of customers, until the model has matured. For more information about model maturity, see Model learning for new actions.

The value of outbound model maturity can be checked by clicking the **Properties** icon in the header of Next-Best-Action Designer. As in the following figure:





Confirm whether the journey actions are being correctly output by Next-Best-Action Designer

For container service invocations, in order to proceed with the interactions that are being recorded, the journey action that you expect must be returned as the first ranked result (that is, the top result) for the subject ID for which the action results are being requested. Other actions may be weighted or ranked higher than your journey actions. Review your journey action weighting and propensities to see if they need to be manually adjusted.

For outbound scheduled runs, the journey actions should be included in the batch output results.

In both cases, the results should include the appropriate data for *pyChannel*, *pyJourney* and *pyStage*. If there is a mismatch between these values and what is stored in the *JourneyStages*, *JourneyActions*, or *ChannelSettings* DDRs, the journey's interaction data will not be correctly captured.

Issue: Customer journey results may not be appearing as expected in the Journey Visualizer

Use the following troubleshooting tips to help solve issues with customer journey results not appearing as expected in the Journey Visualizer.

You might encounter the following issues:

- Cause: A Real-time Data Grid node is not behaving properly
- Cause: Check the Journey Visualizer timeframe
- Cause: The action's Channel is not active
- Cause: Check Form Fields on Important Decision Data Rules
- Cause: An interaction does not meet the entry criteria for the customer journey stage
- Cause: Check Engagement Policy and action eligibility criteria
- Cause: Is the journey-associated action the top-ranked output?

For all of the troubleshooting scenarios, it is important to understand that none of these fixes will be able to reflect any data processed while the given problem was present. Next-Best-Action customer journeys process interactions as they are received, and as such, do not have the ability to retroactively apply changes to interactions made while the problem was present. Once all problems are resolved, all data going forward should be accurately recorded.

Pega Customer Decision Hub

Cause: A Real-time Data Grid node is not behaving properly

It's possible that a Real-time Data Grid node is not behaving properly. This service is crucial to capturing Next-Best-Action customer journey data. If the nodes are not configured properly, no journey data will be recorded.

Solution: Check the Real-time Data Grid node status

1. In the header of Dev Studio, in the **Configure** menu, click **Decisioning Infrastructure Services Real-time Data Grid** .
2. Ensure that all nodes are reporting NORMAL status.
3. If a node status is not NORMAL, click the **Execute** menu in that node's row, and then select the **Start** option to make the node operational again.

Cause: Check the Journey Visualizer timeframe

In the Journey Visualizer, there is a series of controls that allow you to configure the timeframe for the journey data that you want to view. The options are Current Quarter, Last Quarter, and This Year. By default, Current Quarter is selected.

Ensure that the selected timeframe is applicable for the date and time captured in the responses that should be shown.

Cause: The action's Channel is not active

Solution: Ensure that the action's Channel is active

1. Open **Next-Best-Action Designer**
2. Click **Channels**.
3. Locate the channel that you want to check from the list of available channels. If the **Channel** switch is toggled to the left, the channel is not enabled. In this case, click **Edit**, and then turn on the **Channel** switch.
4. Click **Save**.

Cause: Check Form Fields on Important Decision Data Rules

There are a number of Decision Data Rules that are used to support customer journey functionality. When you have problems with the customer journey results, the specific rules are:

- Group-level DDRs (one DDR for each Group defined in the hierarchy, named according to the Group's name and in the SR class path)
- *NBA_Journey*, located in the *<SR Class>-NBA_Journey-NBA_Journey* class

Each of these rules should have *pyJourney* and *pyStage* on the Form tab as form fields. If the property is not defined on the tab, journey data will not be saved properly to the necessary data sets, and the data will not be reflected in the Journey Visualizer.

Solution: Check form fields on important Decision Data Rules

If these properties are not present on the Form tab, they will need to be added manually:

1. Click **Check out** for the DDR, if checkout is required for the environment.
2. At the bottom of the property list on the **Form** tab, click the **Add field** button, and add fields for pyJourney and pyStage.
3. Click **Save**, and then click **Check in**.
   If this DDR has been moved from a previous environment (or will be moved to another environment), it is very important that the *pyEditElement* circumstanced section

that is associated with the DDR is moved with it. It is also worth checking that the associated *pyEditElement* is configured as expected.

Cause: An interaction does not meet the entry criteria for the customer journey stage

Each customer journey stage allows for configuration of detailed filtering. If an interaction does not meet the entry criteria for a stage, it will not be considered for that stage in the journey. If an interaction doesn't qualify for at least one stage in the journey, then the interaction will not be reflected in the Journey Visualizer at all.

Solution: Check the existing entry criteria

1. Open **Next-Best-Action Designer** .
2. Click **Engagement Policy**.
3. In the **Business structure** section, choose the customer journey that you want to check.
4. Click the **More** icon on the specific stage that you want to check.
5. Select **View entry criteria** (in read-only mode), or **Edit stage entry criteria** (in editable mode).
6. Ensure that the expected interaction meets at least one stage's criteria, and either update the criteria or revise the interaction to reflect the desired behavior.
   It is possible for more than one journey stage to be applicable for any given interaction. In this scenario, the backing logic will generally attempt to place the interaction in the furthest right applicable stage. This includes the Complete (last) stage. If the interaction is not showing up where it is expected to be applicable, ensure there are no other stages that are also applicable.

Cause: Check Engagement Policy and action eligibility criteria

It is possible to configure eligibility, applicability, and suitability conditions at the Action, Group, and All Groups levels, to control how an action is prioritized. Unlike with the journey criteria, which only determine whether an interaction is applicable to the journey, this potentially means that certain actions will never be presented to the customer if the conditions are not met. If there is an action for which an interaction is expected, it is important to check that the relevant conditions for that action are not preventing it from being presented.

The action conditions can be checked directly on the action rule form. It is also possible to see any inherited conditions from the Group-level criteria.

The Group-level conditions can be viewed on the Engagement policy tab of the Next-Best-Action Designer. Be sure to check the All actions conditions, as well as the conditions for the context in which the action is defined.

Lastly, the All Groups conditions can be checked on the All groups node on the Engagement policy tab of Next-Best-Action Designer. Check conditions defined for the context in which the action is defined.

If the interaction with the action in question meets all of the criteria in these locations, it should be considered for presentation to the customer.

Cause: Is the journey-associated action the top-ranked output?

It is possible for multiple actions to be returned in some circumstances. It is then important to determine whether the customer journey-associated action in question is the top-ranked action output.

Cause: Container service needs to return action as the highest ranked action

When the Container service is called, it may return more than one applicable action as part of the response. Determine whether the action in question is being returned with a *pxRank* value of 1.

Cause: Verify Interaction History data

A lot of understanding of the situation can be gained by checking the Interaction History table to determine what data has been written for the interaction in question.

This can be found in the *pr_data_ih_fact* table in the *pegadata* schema of the database. By using *pzjourneyid*, it is possible to determine what customer journey information is associated with the interaction because it is joining on to the *pzid* of the *pr_data_ih_dim_journey* table. This will provide the *pyJourney* and *pyStage* values that were recorded. Similar action can be taken with the *pzchannelid* column in the *fact* table, to determine the *pyChannel* value stored in the *pr_data_ih_dim_channel* table.

Cause: Verify journey data on interaction

Whether the interaction is being verified in a container REST response or in the Interaction History, it is important to confirm that *pyJourney* and *pyStage* match the journey and stage IDs for the expected customer journey. If they do not, their data will not be reflected in that journey and stage.

For more information, see Cause: JourneyActions Decision Data Rule is misconfigured.

Cause: Ensure that actions have treatments

If an action does not have treatments associated, it will not be output for channels that do not have treatments. Add at least one treatment to each channel for which the action should be considered, then save and Check in the action. Then, re-process the interaction.

For more information, see Cause: Treatment configuration.

Cause: Ensure the correct Channels are active

On the Channels tab of Next-Best-Action Designer, ensure that the channels for which the action has treatments (or the channels that should be active in general) are toggled on. Any channels that are off will not be considered for Next-Best-Action output.

For more information, see Cause: The action's Channel is not active.

Cause: Ensure journey and stage IDs match data in DDRs and Proposition Filters

Behind the scenes, each customer journey and stage is given an identifier that is used to tie all of the configurations together and support runtime processing. If for any reason, these values have gotten out of sync, proper functionality will be disrupted.

In the *<Model Name (usually NBA)>_Journey DDR*, there should be a row for each stage of each customer journey associated with the model. Each row should have the convention *<Model Name><JourneyID><StageID>*. That is, if there is a customer journey called "Platinum Cards" with a "First Awareness" stage and a "Complete" stage, there should be a row that is likely called *NBAPlatinumCardsFirstAwareness* and a row called *NBAPlatinumCardsComplete*.

These rows should match the proposition values in the *JourneyEligibility* proposition filter, where conditions are stored under *<Model Name>_JourneyIssue* and *<Model Name>_JourneyGroup*. If there is not a row in the proposition filter for each proposition row in the DDR, data has fallen out of sync and the customer journey may need to be re-saved.

Cause: A note about Model Maturity

Model maturity is a feature that impacts the back-end processing of interactions that may result in newer actions being suppressed until the model has had a number of interactions processed. This can be a factor in why a particular action is not promoted, even if all conditions and criteria are met. It is possible to toggle this feature off for development and testing purposes, but this is definitely not recommended in any customer environments, as this behavior is intended in real-world scenarios.

For more information, see Check whether outbound model maturity is enabled.

Cause: A note about Contact Policies

Next-Best-Action Contact Policy will impact how many times a particular entity can receive an action in a given time frame. The applicable Contact Policy can be checked by

selecting the Constraints tab of Next-Best-Action Designer, and viewing the **Customer contact limits** section.

The backing Contact Policy rule can also be accessed by clicking **Edit** on the Constraints tab. An **Edit customer contact policy** link will appear beneath the **Customer contact limits** section. This link does not appear in read-only mode.

For more information, see Cause: Contact Policy suppression.

Issue: Data sets that need to be reset

Use the following troubleshooting tips to help solve issues with data sets that need to be reset.

Pega Customer Decision Hub

Cause: There is a data left in customer journey data set

To reset journey data, three data sets must be truncated. This is an all-or-nothing process, because leaving data in any data set will result in corrupted data for the customer journey feature.**Important:** There is no possibility to remove specific journey data from the data sets. Removing any journey data requires removing ALL journey data.

The three data sets that you must reset are:

LastInteraction
: Stores the information for the stages that a customer has interacted with, but does not store customer information. This data set is used to populate the nodes in the Journey Visualizer.

SequenceOfInteractions
: Stores the connections between the stages of the journeys that the customers have moved through, but does not store customer information. This data set is used to populate the links between nodes in the Journey Visualizer.

JourneyStages
: Stores a record for each journey stage with which a customer recorded an interaction.

Solution: Truncate the data sets

Reset a data set by performing the following steps:

1. Click **Actions** in the toolbar and select **Run**.
2. In the **Operation** drop-down menu, select Truncate, and then click **Run**.

Issue: The same stage is being output for each interaction

Use the following troubleshooting tips to help solve issues with the same stage being output for each interaction.

Pega Customer Decision Hub

Cause: Customers are remaining in the same place in a journey

There are occasions when interactions for the same customer will output the same journey stage time after time, causing the customer to remain in the same place in a journey.

Journey stage assignment is not necessarily associated with how many interactions a customer has in a journey, but with the conditions that are defined for the journey.

Each stage can have criteria defined for it, and if the recipient does not meet the criteria for a stage, they will never be placed there. Instead, recipients are only assigned to a stage for which they meet all of the associated criteria.

For more information, see Specifying entry criteria for customer journey stages.

Issue: The interaction did not get the stage that was anticipated

Use the following troubleshooting tips to help solve issues with the interaction not getting the stage that was anticipated.

Pega Customer Decision Hub

Cause: The customer is be eligible for multiple journey stages

When testing or observing customer journey behavior, an interaction that is expected to end up associated with one stage, sometimes ends up associated with a different stage. This is expected behavior, though it is important to understand why.

For any given interaction, it is possible that a customer can be eligible for multiple stages. However, the interaction must be assigned a single journey stage. To reconcile this, the journey portions of the Next-Best-Action Strategy Framework will assign the stage that is farthest along in the journey, or the stage that is furthest to the right. For example, if an interaction is eligible for both stages 1 and 2, the interaction will be assigned to stage 2, as it is further along in the journey.

Additionally, this determination accounts for the “Complete” (last) stage. The last node in the Journey Visualizer can sometimes be confusing, since it is represented slightly differently from the other nodes. You may think that it does not have any data, when actually the data is represented in the last node. This most commonly occurs when the last stage is the recipient’s first interaction, as there will be no line connecting the last node to any other.

If there are no criteria defined in the last stage, this will always be the stage assigned for the interaction, since the default behavior is to return true. To avoid this scenario, be sure to define criteria in the last stage so that it properly restricts eligibility for journey completion.

For more information, see Specifying entry criteria for customer journey stages and Associating actions with customer journey stages.

Issue: Customer journey stage criteria are missing

Use the following troubleshooting tips to help solve issues with missing customer journey stage criteria.

Pega Customer Decision Hub

Cause: The *JourneyEligibility* proposition filter is misconfigured

While safeguards are in place to prevent this from happening, it is possible that changes to journey stage criteria appear to have been saved correctly (with no error message), but are not present when you reload Next-Best-Action Designer and look at the criteria.

The first step should be to confirm that the *JourneyEligibility* proposition filter rule has been properly saved into the user’s artifacts ruleset. If the only copy exists in *PegaMKT-Engine*, then the save-as has failed.

The next step is to make sure that the *JourneyEligibility* proposition filter is in a ruleset that has an unlocked version available. This should be the case, as the journey would not be editable if the artifacts ruleset did not have an unlocked version available, but it is possible that the rules have somehow become out of sync.

Lastly, it is possible that the configured conditions have caused some sort of error that was not properly caught by the existing error handling at design time.

For all of these cases, saving the journey on the Engagement policies tab of Next-Best-Action Designer should perform the necessary rule updates that are required to retain eligibility changes. To debug this issue further, trace the *SaveConfig* activity in the *Rule-PegaMKT-NBA-Config-Journey* class. The *SaveConfig* activity calls the *SaveNBAJourneyData* activity, which in turn calls the *UpdateJourneyPropFilter* activity. Tracing this path will likely reveal the cause of the failure.

Issue: Next-Best-Action template support

Use the following troubleshooting tips to help solve issues with Next-Best-Action template support.

Pega Customer Decision Hub

Cause: Next-Best-Action templates do not support customer journeys

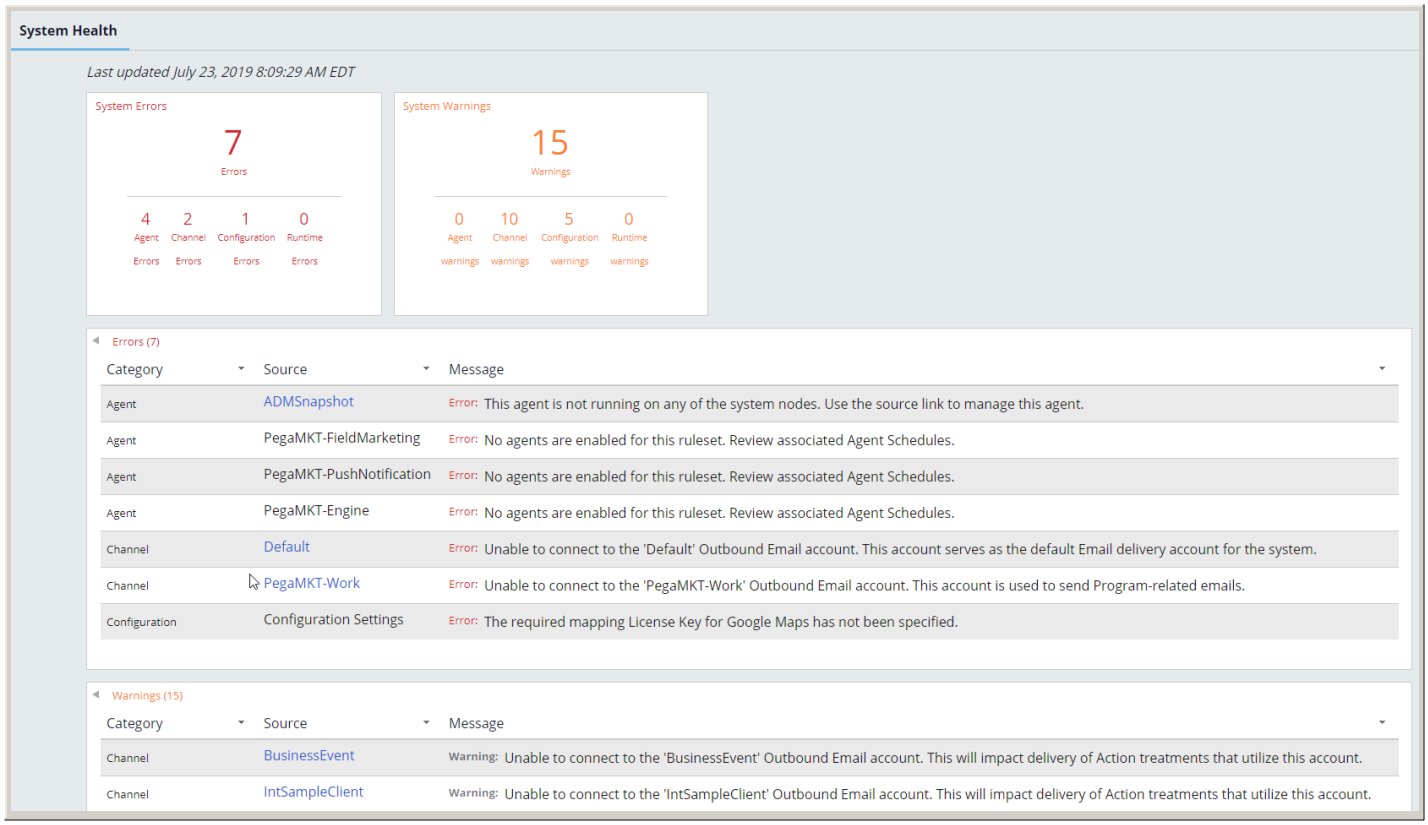
As of Pega Customer Decision Hub version 8.6, Next-Best-Action templates do not support customer journeys that are configured as part of the model definition. The exported Next-Best-Action templates will not include any customer journey information.

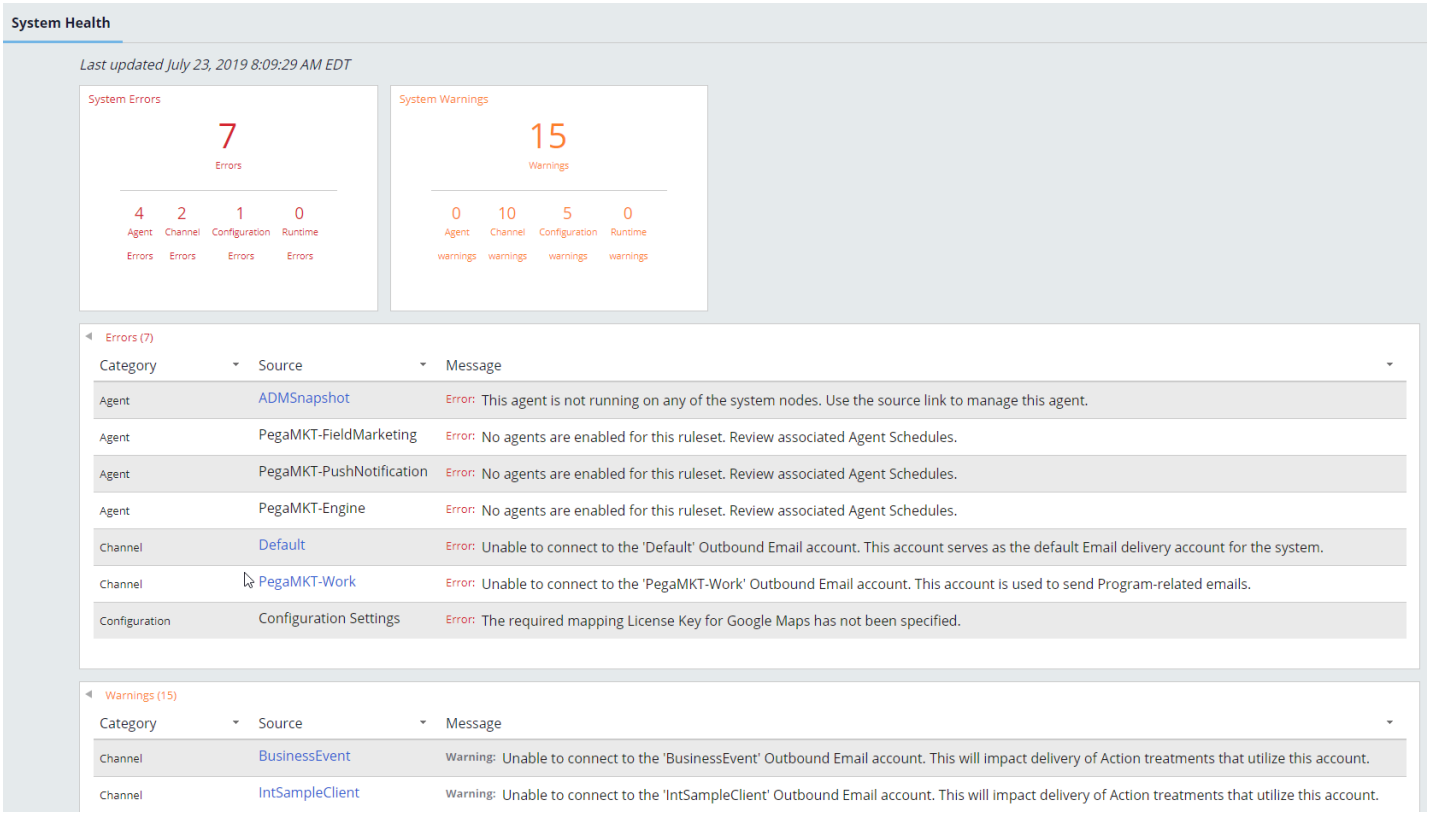
Monitoring system health in Pega Customer Decision Hub

Pega Customer Decision Hub provides a diagnostic tool for your application called System Health. System Health allows administrators to quickly review errors and warnings associated with their application. The System Health landing page is visible only to administrators and is accessible in the Pega Customer Decision Hub portal via: **Configuration Settings System Health** .

Pega Customer Decision Hub

Each time the System Health landing page is launched (or refreshed), the health check is initiated. Once the check is complete, the results of the check are displayed and resemble the following:





Note: Due to the nature of some of the checks performed (e.g. validating Email and SMS account connectivity), the System Health check may take some time to complete. During this duration, a loading screen is displayed. To reduce this load time, administrators should ensure all connection-related errors and warnings are resolved.

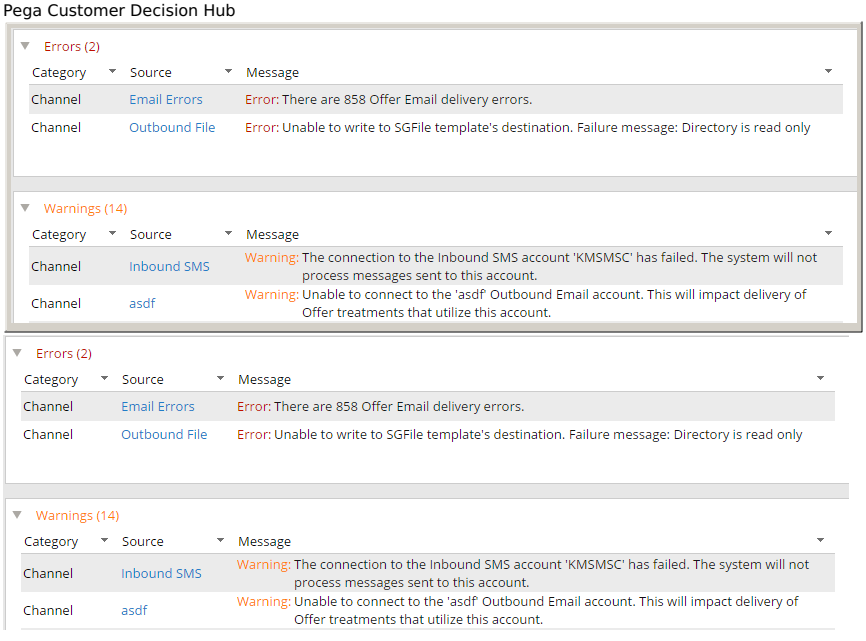
The System Health page displays the following information:

- Last updated date and time
- Errors and Warnings cards
- Errors and Warnings grids

Identifying and reporting issues

The Errors and Warnings grids provide detailed information on the health check failure (i.e. error or warning).

Pega Customer Decision Hub



The following columns are available in both grids:

Column Name	Description
Category	Category of the error or warning (Agent, Channel, Configuration, or Runtime)
Source	Actionable area of the system associated with the error or warning. This could be a landing page, a data instance, a rule form, or a work object. Clicking this field will open the referenced object or page. In most cases, the administrator should be able to rectify the failure by making appropriate changes in the opened object or page.
Message	Descriptive text that explains the error or warning. For longer messages, use the **More** link to view the complete message and the **Less** link to revert to the abridged view.

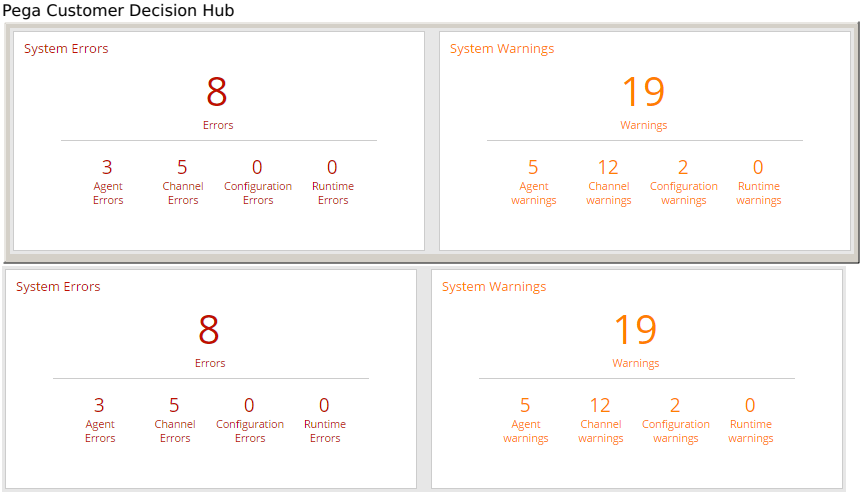
Column Name	Description

The header of each grid displays the total number of items in the grid. Users can also click the header to toggle the grid view between expanded and collapsed. Additionally, each column in the two grids supports sorting and filtering.

Understanding Pega Customer Decision Hub errors and warnings

The Errors and Warnings cards are displayed at the top of the System Health landing page and provide the administrator with a quick overview of the status of their application.

Pega Customer Decision Hub



System Errors
8
Errors
3
Agent Errors
5
Channel Errors
0
Configuration Errors
0
Runtime Errors
System Warnings
19
Warnings
5
Agent warnings
12
Channel warnings
2
Configuration warnings
0
Runtime warnings

Each card displays the total number and a breakdown of this total into the following categories:

- Agent
- Channel
- Configuration
- Runtime

Clicking anywhere on a card ensures that the corresponding (Errors/Warnings) grid below is visible. This may result in the other grid being collapsed.

Understanding agent errors and warnings

The following table explains the scenarios that contribute to the Agent Errors and Agent Warnings lists.

Pega Customer Decision Hub

Failure Scenario	Use Source link to...
The agent is not running on any system nodes.	Launch the Agent Detail page for this agent. Administrators can manage (stop and start) the agent from this page.
The agent has failed and stopped on at least one system node.	Launch the Agent Detail page for this agent. Administrators can manage (stop and start) the agent from this page.
The agent has stopped on one or more system nodes.	N/A
No agents are enabled for this ruleset.	N/A

Optional Pega Customer Decision Hub agents are those that are not tied to core functionality and may not be utilized in some installations. Typically, these agents perform channel-specific functionality such as Email, SMS, and Push Notification. Administrators should be aware of the functionality being used by their application, so they can better assess the true impact of failures in these agents.

Understanding channel errors and warnings

The following table explains the scenarios that contribute to the Channel Errors and Channel Warnings lists.

Pega Customer Decision Hub

Failure Scenario	Use Source link to...
Default outbound email account has not been configured.	N/A
There are action email delivery errors.	N/A
Outbound email account could not be validated.	Open the email account's rule form and re-test the connection.
Outbound email account fails to connect.	N/A
Outbound SMS account fails to connect.	N/A
Default Passbook certificate could not be opened.	N/A
No default Push Notification app exists or default app has no defined variants	N/A
Default outbound SMS account has not been configured.	N/A
The Apple Passbook certificate could not be opened.	N/A
Could not write to the template file.	N/A
Inbound SMS account fails to connect.	N/A

Understanding configuration errors and warnings

The following table explains the scenarios that contribute to the Configuration Errors and Configuration Warnings lists.

Pega Customer Decision Hub

Failure Scenario	Use Source link to...
Public Link URL is not specified.	Open the Resource URLs landing page. Administrators should set this URL if their

Failure Scenario	Use Source link to...
	application utilizes the Email channel.
The required mapping License Key for Google Maps is not specified.	N/A
The specified default destination ruleset version is locked.	Open the access group. Administrators should update instance to reference an unlocked ruleset version.
The selected customer Email property used to determine the recipient's email address does not have external mapping.	Open the Customer class defined on the Application Settings page.
The selected prospect Email property used to determine the recipient's email address does not have external mapping.	Open the Prospect class defined on the Application Settings page.
Outbound Default Data Source is not configured or doesn't exist in the system	Open the Configuration Settings landing page tab. Administrators should update the settings to point to a valid data source.
The Outbound Default Data Source fails to connect.	N/A
The default destination ruleset for an access group is not specified.	Open the access group. Administrators should set the Default destination ruleset to their application's artifacts ruleset.
The default destination ruleset for an access group is not locked and no open versions exist for the ruleset.	Open the ruleset. Administrators should add a new unlocked version and update the access group to reference this new version.

Understanding runtime errors and warnings

The following table explains the scenarios that contribute to the Runtime Errors and Runtime Warnings lists.

Pega Customer Decision Hub

Failure Scenario	Use Source link to...
A campaign has failed.	Open the failed work object.

Note: In most cases when the System Health references recent failures, the time duration is configured as one of the criteria in the backing Report Definition (which is marked as extensible). The shipped implementations usually use a time duration of "Greater than or equal to Yesterday" for determining whether a failure is recent.

Reference

This section contains information for administrators.

Pega Customer Decision Hub

- Pega Customer Decision Hub Interaction History data model
- Pega Customer Decision Hub Dynamic System Settings
- Pega Customer Decision Hub access roles and privileges
- Adding new response outcomes
- Disabling the Decision Data Store as action state storage on Pega Cloud
- Pega Customer Decision Hub agents
- Managing Decision Data rules in Pega Customer Decision Hub
- Using a Configuration Modal Window
- Application configuration
- Pega Community Wiki

Pega Customer Decision Hub Interaction History data model

Pega Customer Decision Hub stores all customer responses to specific actions in Interaction History database tables. The recorded interactions can then be used to train adaptive models through delayed learning, or to create business intelligence reports.

Pega Customer Decision Hub

Properties in the *pr_data_ih_dim_application* table

The *pr_data_ih_dim_application* table contains properties that describe the decision path which issued the action. Properties in this table are defined in the *Data-Decision-IH-Dimension-Application* class.

Property	Data type and length	Description	Association
pzid	numeric (19,0)	Unique identifier that is created for each record in the application dimension table	*pxApplicationDimension*
pxcreatedatetime	timestamp without time zone	Time when the record was created	*pxApplicationDimension*
pyapplication	character varying (255)	The name of the application that is calling Pega Customer Decision Hub	*pxApplicationDimension*
pyinteraction	character varying (255)	Name of the Interaction rule. **Note:** Interaction rules are considered deprecated in Pega Customer Decision Hub 8.6	*pxApplicationDimension*
pystrategy	character varying (255)	Name of the final strategy that triggered the creation of an interaction record	*pxApplicationDimension*
pycomponent	character varying (255)	Name of the final strategy that triggered the creation of an interaction record	*pxApplicationDimension*
pyapplicationversion	character varying (16)	Application version	*pxApplicationDimension*

Properties in the *pr_data_ih_dim_channel* table

The *pr_data_ih_dim_channel* table contains properties that describe the channel of the interaction. Properties in this table are defined in the *Data-Decision-IH-Dimension-Channel* class.

Property	Data type and length	Description	Association
pzid	numeric (19,0)	Unique identifier that is created for each record in the customer dimension table	*pxChannelDimension*
pxcreatedatetime	timestamp without time zone	Time when the record was created in the dimension table	*pxChannelDimension*
pydirection	character varying (255)	Direction of the interaction (inbound or outbound)	*pxChannelDimension*

Property	Data type and length	Description	Association
pychannel	character varying (255)	Name of interaction channel	*pxChannelDimension*
pytreatment	character varying (255)	Name of treatment used in the interaction	*pxChannelDimension*
devicetype	character varying (255)	Type of the device used in the interaction	*pxChannelDimension*
uri	character varying (255)	The URI used in the interaction	*pxChannelDimension*
useragent	character varying (255)	Property to hold UserAgent (for example: browser, client) information	*pxChannelDimension*
channelgroup	character varying (255)	A subdivision of the pychannel	*pxChannelDimension*
channelsubgroup	character varying (255)	A subdivision of the channel group	*pxChannelDimension*

Properties in the *pr_data_ih_dim_context* table

The *pr_data_ih_dim_context* table contains properties that describe the decision context of the interaction, that is, the level of customer contact for which the interaction was recorded. Properties in this table are defined in the *Data-Decision-IH-Dimension-Context* class.

Property	Data type and length	Description	Association
pzid	numeric (19,0)	Unique Identifier that is created for each record in the context dimension table	*pxContextDimension*
pxcreatedatetime	timestamp without time zone	Time when the record was created in the dimension table	*pxContextDimension*
pycategory	character varying (255)		*pxContextDimension*
pyreason	character varying (255)		*pxContextDimension*
pydisplaycategory	character varying (255)		*pxContextDimension*
pytoprank	character varying (255)		*pxContextDimension*
controlgroupvalidditystart	timestamp without time zone	Start date time for a control group validity period	*pxContextDimension*
controlgroupvalidityend	timestamp without time zone	End date time for the control group validity period	*pxContextDimension*
mkttype	character varying (255)	Context type	*pxContextDimension*
mktvalue	character varying (255)	Value associated with the context type	*pxContextDimension*
pycontext	character varying (255)		*pxContextDimension*
pycallmode	character varying (255)		*pxContextDimension*
cgvalidditystart	timestamp without time zone	Start date time for the control group validity period	*pxContextDimension*
cgvalidityend	timestamp without time zone	End date time for the control group validity period	*pxContextDimension*
workid	character varying (128)		*pxContextDimension*

Properties in the *pr_data_ih_dim_customer* table

The *pr_data_ih_dim_customer* table contains properties that describe the customer for whom the interaction was recorded. Properties in this table are defined in the *Data-Decision-IH-Dimension-Customer* class.

Property	Data type and length	Description	Association
pzid	numeric (19,0)	Unique Identifier that is created for each record in the customer dimension table	*pxCustomerDimension*
pxcreatedatetime	timestamp without time zone	Time when the record was created in the dimension table	*pxCustomerDimension*
pysubjecttype	character varying (255)	Class value of the subject	*pxCustomerDimension*
pycustomersegment	character varying (255)	The segment to which the customer belongs	*pxCustomerDimension*
pycustomersubsegment	character varying (255)	Subdivision of *pyCustomerSegment*	*pxCustomerDimension*
pygrouptype	character varying (255)		*pxCustomerDimension*
modelcontrolgroup	character varying (255)	Property holding the control group information for a customer in a given outcome optimization path	*pxCustomerDimension*

Properties in the *pr_data_ih_dim_journey* table

The *pr_data_ih_dim_journey* table contains properties that describe the customer journey. Properties in this table are defined in the *Data-Decision-IH-Dimension-Journey* class.

Property	Data type and length	Description	Association
pxcreatedatetime	timestamp without time zone	Time when the record was created in the dimension table	*pxJourneyDimension*
pzid	numeric (19,0)	Unique identifier that is created for each record in the journey dimension table	*pxJourneyDimension*
pyjourney	character varying (255)	Name of the customer journey	*pxJourneyDimension*
pystep	character varying (255)	Step of the customer journey where the interaction was recorded	*pxJourneyDimension*
pystage	character varying (255)	Stage of the customer journey where the interaction was recorded	*pxJourneyDimension*

Properties in the *pr_data_ih_dim_location* table

The *pr_data_ih_dim_location* table contains properties that describe the location of the customer when the interaction took place. Properties in this table are defined in the *Data-*

Decision-IH-Dimension-Location class.

Property	Data type and length	Description	Association
pzid	numeric (19,0)	Unique identifier that is created for each record in the location dimension table	*pxLocationDimension*
pxcreatedatetime	timestamp without time zone	Time when the record was created in the dimension table	*pxLocationDimension*

Properties in the *pr_data_ih_dim_operator* table

The *pr_data_ih_dim_operator* table contains properties that describe the operator linked to the interaction. Properties in this table are defined in the *Data-Decision-IH-Dimension-Operator* class.

Property	Data type and length	Description	Association
pzid	numeric (19,0)	Unique Identifier that is created for each record in the operator dimension table	*pxOperatorDimension*
pxcreatedatetime	timestamp without time zone	Time when the record was created in the dimension table	*pxOperatorDimension*
pyorganization	character varying (255)	Organization name of the operator	*pxOperatorDimension*
pydivision	character varying (255)	Division name of the operator	*pxOperatorDimension*
pyunit	character varying (255)	Unit name of the operator	*pxOperatorDimension*
pyoperator	character varying (255)	Operator ID	*pxOperatorDimension*

Properties in the *pr_data_ih_dim_outcome* table

The *pr_data_ih_dim_outcome* table contains properties that describe the outcome of the interaction. Properties in this table are defined in the *Data-Decision-IH-Dimension-Outcome* class.

Property	Data type and length	Description	Association
pzid	numeric (19,0)	Unique Identifier that is created for each record in the outcome dimension table	*pxOutcomeDimension*
pxcreatedatetime	timestamp without time zone	Time when the record was created in the dimension table	*pxOutcomeDimension*
pyoutcome	character varying (255)	Deprecated in Pega Customer Decision Hub	*pxOutcomeDimension*
pyResponse	character varying (255)	Response (for example: Accepted, Rejected)	*pxOutcomeDimension*
pybehaviour	character varying (255)	Behavior associated with the response (positive, negative, neutral)	*pxOutcomeDimension*

Properties in the *pr_data_ih_fact* table

The *pr_data_ih_fact* table contains additional properties that describe the interaction. Properties in this table are defined in the *Data-Decision-IH-Fact* class.

Property	Data type and length	Description	Association
pxfactid	numeric (19,0)	Unique identifier created for each record inserted into the fact table	none
pysubjectid	character varying (255)	Unique identifier for the subject that the decisioning was made for (for example: CustomerID or Account ID, depending on the context of the decision)	none
pxoutcometime	timestamp without time zone	Timestamp corresponding to when the record was stored in the interaction history tables	none
pzactionid	numeric (19,0)	Identifier used for joining the action dimension	none
pzapplicationid	numeric (19,0)	Identifier used for joining the application dimension	none
pzchannelid	numeric (19,0)	Identifier used for joining the channel dimension	none
pzcontextid	numeric (19,0)	Identifier used for joining the context dimension	none
pzcustomerid	numeric (19,0)	Identifier used for joining the customer dimension	none
pzlocationid	numeric (19,0)	Identifier used for joining the location dimension	none
pzoperatorid	numeric (19,0)	Identifier used for joining the operator dimension	none
pzoutcomeid	numeric (19,0)	Identifier used for joining the outcome dimension	none
pygroupid	character varying (255)	In a multi-level configuration, this is the name of the parent context.	none
<context key>	character varying (255)	For multilevel configurations, on save of context dictionary, properties are added to the table that correspond to each context.	
pxdecisiontime	timestamp without time zone	Time stamp corresponding to when the decision was issued	none

Property	Data type and length	Description	Association
pyexternalid	character varying (255)		none
pylatitude	numeric (19,9)	Latitude	none
pylongitude	numeric (19,9)	Longitude	none
pxpriority	numeric (19,9)	Priority assigned to the action	none
pypropensity	numeric (19,9)	Calculated propensity for the action	none
pxinteractionid	numeric (19,0)	Unique 64-bit identifier for all records that belong to the same interaction	none
pxrank	numeric (18,0)	Rank of the action	none
pypartitionkey	integer	Partition key assigned to the fact	none
pzjourneyid	numeric (19,0)	Identifier used for joining the journey dimension	none
pyrevenue	numeric (19,4)		none
pymaxbudget	numeric (19,4)	Maximum budget assigned to action (used for negotiation)	none
pytargetbudget	numeric (19,4)	Target budget assigned to action (used for negotiation)	none
pyfulfilled	numeric (19,4)		none
pyweight	numeric (19,4)	Value to increase or decrease propensity	none
pyisfactid	numeric (19,0)		none
pxdecisionreference	character varying (255)		none
pyreason	character varying (255)		none
ipaddress	character varying (255)	IP address	none
revenue	numeric (19,4)	Revenue associated with the action	none
cost	numeric (19,4)	Cost associated with the action	none
urlid	numeric (19,0)		none
utm_medium	character varying (255)	The type of link which was used	none
referrerurl	character varying (1024)	The HTTP referrer of the requested website	none
externalaudienceid	character varying (255)	Paid Media Manager uses this property to associate ads on platforms other than Google Ads and Facebook Ads to specific audiences in Pega Customer Decision Hub	none

Properties in the *pr_data_ih_association* table

The *pr_data_ih_association* table contains properties that describe the associated data for the interaction. Properties in this table are defined in the *Data-Decision-IH-Association* class.

Property	Data type and length	Description	Association
pysubjectid	character varying (255)	Identifier of the subject of the decision (customer or account)	none
pyassociatedid	character varying (255)		none
pxcreatedatetime	timestamp without time zone	Timestamp of record creation in the table	none
pyassociationstrength	numeric (19,9)		none

Properties in the *pr_data_ih_dim_action* table

The *pr_data_ih_dim_action* table contains properties that describe the action associated with the interaction. Properties in this table are defined in the *Data-Decision-IH-Dimension-Action* class.

Property	Data type and length	Description	Association
pzid	numeric (19,0)	Unique Identifier that is created for each record in the association dimension table	*pxActionDimension*
pxcreatedatetime	timestamp without time zone	Time the record was created in the dimension table	*pxActionDimension*
pyissue	character varying (255)	Issue the action belongs to.	*pxActionDimension*
pygroup	character varying (255)	Group the action belongs to	*pxActionDimension*
pyname	character varying (255)	Action identifier	*pxActionDimension*
pylabel	character varying (255)	Action name or label	*pxActionDimension*
bundlename	character varying (255)	Bundle name of the action	*pxActionDimension*
bundlehead	character varying (8)	Bundle head of the action	*pxActionDimension*

- Extending Interaction History

Extending Interaction History

Learn about the underlying configuration of Interaction History and how to extend it to match your business objectives.

You can extend Interaction History to record more aspects of customer interactions as property values and use those values to determine the next best action for your customers or as key performance indicators (KPIs) of your decisioning strategy. As an example, Interaction History can be extended with two properties: *ProductID* and *HandleTime*.

Interaction History is the Decision Management data layer that stores all interactions with customers (responses to specific propositions). Information that is collected during each interaction is written into the appropriate Interaction History database table. Each database table contains several default properties that are defined on the corresponding class. Case-sensitive name matching is used between properties in the database table, Strategy Result (SR) properties, and Interaction History properties. Typically, the same Interaction History is used by multiple applications that use a shared implementation.

- Data transform for Interaction History configuration
- Extending Interaction History

Data transform for Interaction History configuration

The *pyInteractionHistoryConfiguration* data transform that is defined in the *Data-Decision-IH-Configuration* class configures Interaction History by setting the properties in this class. In your application ruleset, you can override the *pyInteractionHistoryConfiguration* data transform to customize Interaction History.

The following table provides an overview of the properties that you can set by using the *pyInteractionHistoryConfiguration* data transform.

Properties that can be set by using the pyInteractionHistoryConfiguration data transform

Property	Type	Description
pyFactProperties	Value List (Text)	Lists FACT properties.
pyActionProperties	Value List (Text)	Lists action dimension properties.
pyApplicationProperties	Value List (Text)	Lists application dimension properties.
pyChannelProperties	Value List (Text)	Lists channel dimension properties.
pyContextProperties	Value List (Text)	Lists context dimension properties.
pyCustomerProperties	Value List (Text)	Lists customer dimension properties.
pyLocationProperties	Value List (Text)	Lists location dimension properties.
pyOperatorProperties	Value List (Text)	Lists operator dimension properties.
pyOutcomeProperties	Value List (Text)	Lists outcome dimension properties.
pyMeasurements	Value List (Text)	Lists the subset of *pyFactProperties* that can be used as KPIs.

Note: The *pyOutcome* dimension drives interaction, adaptive learning, and business monitoring results, even if you explicitly set other properties in this dimension.

For more information, see Data transforms.

Extending Interaction History

The process of extending the Interaction History layer in your application consists of the following steps:

1. Adding columns in a database table
2. Adding extensions in application rulesets

Adding columns in a database table

Select the most appropriate database table. It is recommended that you add a column to the FACT table if you expect that the column will contain multiple values that are unique or if the column will represent a property to be used as a KPI. In any other case, consider adding columns to dimension tables. For example, you can extend Interaction History with the *ProductID* and *HandleTime* properties.

- Perform the following steps to add *ProductID*, which is the ID of the offered product, to a dimension database table:
  1. Add the column to the action dimension table, because the list of product IDs consists of a limited number of values, and *ProductID* is part of a proposition.
  2. Specify the data type of the database column as one that can hold characters and numbers, for example, VARCHAR2(255), because *ProductID* has a text value.
- Perform the following steps to add *HandleTime*, which is the duration of the conversation, to the FACT table:
  1. Add the column to the FACT table, because a large number of distinct values is expected and the property is used as a KPI.
  2. Specify the data type of the database column as one that can hold integers, for example, NUMBER(18,0), because *HandleTime* has an integer value.

For more information, see Database tables.

Adding extensions in application rulesets

Add the properties that are extensions of Interaction History to the FACT, dimension, and SR class.

Note: Properties can be shared among multiple applications or they can be application-specific if security or privacy considerations make certain properties unavailable for other applications. Assume that the sample *ProductID* property is shared among call center and outbound marketing applications while *HandleTime* is specific only to the call center application.

- Perform the following actions to extend Interaction History with the *ProductID* property for all applications in the action dimension table:
  1. Add a new property *ProductID* of type Text to the *Data-Decision-IH-Dimension-Action* data model in the application rulesets of the call center and outbound marketing applications.
  2. Add the *ProductID* property of type Text to the *Strategy Result* (*SR*) class of both applications.
  3. Override the *pyInteractionHistoryConfiguration* data transform in the *Data-Decision-IH-Configuration* data model for both application rulesets:
     1. Add a **Set** action and set **Primary.pyActionProperties(<APPEND>)** to **"ProductID"**.
     2. Enable the **Call superclass data transform** setting to ensure that the default configuration is included.
- Perform the following actions to extend Interaction History with the *HandleTime* property for the call center application:
  1. Add a new property *HandleTime* of type Integer to the *Data-Decision-IH-Fact* data model in the application ruleset of the call center application.
  2. Add the *HandleProperty* to the *SR* class of the call center application.
  3. Override the *pyInteractionHistoryConfiguration* data transform in *Data-Decision-IH-Configuration* for the call center ruleset:
     1. Add a **Set** action and set **Primary.pyFactProperties(<APPEND>)** to **"HandleTime"**.
     2. Add a **Set** action and set **Primary.pyMeasurements(<APPEND>)** to **"HandleTime"**. This step is necessary so that you can use the property as a KPI.
     3. Enable the **Call superclass data transform** setting.

Tip: If the new properties are not recognized, save the corresponding database table rule, then log out and log in again.**Tip:** To prevent issues with loading or saving instances of a class, if you add a property to the FACT or dimension classes but that property is not to be considered part of Interaction History, configure the property's persistence settings so that it does not save property data.

Pega Customer Decision Hub Dynamic System Settings

Pega Customer Decision Hub uses Dynamic System Settings to manage various configurable items. Most of these settings are owned by the *PegaMKT-Engine* and *PegaMKT-Install* rulesets. In addition, some settings are owned by the *PegaMKT-Install*, *PegaMKT-Portal*, and *PegaMKT-PushNotification* rulesets. Administrators can edit these settings to modify the behavior of various elements.

Pega Customer Decision Hub

The following tables contain descriptions of these settings, divided according to the setting purpose. The tables contains the setting name, ruleset, and description.

- Installation
- Upgrade

- CRM implementation
- Basic configuration
- Adaptive learning
- Campaign data flow
- Channels
- Containers
- Content repository
- Microsites
- Paid Media Manager
- Segments
- Other configuration

Installation

Dynamic System Settings in the *PegaMKT-Install* ruleset are typically set during application installation or setup. They are listed here for reference only.

Setting name	Owning ruleset	Description
MKTBuildID	*PegaMKT-Install*	The build from which this instance of Pega Customer Decision Hub was created.
MTKCurrentVersion	*PegaMKT-Install*	The ruleset version which you currently have installed.
MKTApplicationLabel	*PegaMKT-Install*	The application label visible in the portals.
MKTApplicationName	*PegaMKT-Install*	The application name.
MKTOrgName	*PegaMKT-Install*	The organization name.
MKTOrgLabel	*PegaMKT-Install*	An optional short description for the organization.
MKTAdminUserEmail	*PegaMKT-Install*	The email of the admin user.
MKTAdminUserName	*PegaMKT-Install*	The name of the admin user.
MKTAdminUserId	*PegaMKT-Install*	The ID of the admin user.
MKTTestOpsPassword	*PegaMKT-Install*	Password for the test user.
MKTUnitLabel	*PegaMKT-Install*	An optional short description for the unit.
MKTDivLabel	*PegaMKT-Install*	An optional short description for the division.
MKTDivName	*PegaMKT-Install*	The division name.
MKTUnitName	*PegaMKT-Install*	The unit name.
MKTWorkGroupManager	*PegaMKT-Install*	The ID of the work group manager.
MKTBuiltOnApp	*PegaMKT-Install*	The application on which your application is built.
MKTDescription	*PegaMKT-Install*	The application description.

Upgrade

Setting name	Owning ruleset	Description
MKTStagingUpgrade0802	*PegaMKT-Engine*	Used in the OutboundMaster agent in ProcessOutboundStaging to determine whether the seamless upgrade for outbound staging has successfully happened.
UpgradeSuccessful	*PegaMKT-Engine*	Initially set to false when installed. Successful run of upgrade scripts will set this to true.

CRM implementation

Setting name	Owning ruleset	Description
ConnectMKTURL	*PegaMKT-Integration*	For implementations of the Pega CRM suite, this is the URL where Pega Customer Decision Hub is deployed. Configure this setting on a remote Event Strategy Manager environment when integrating with events. For more information, see Responding to real-time customer engagement with real-time event triggers.

Basic configuration

Setting name	Owning ruleset	Description
MarketingOperations	*PegaMKT-Engine*	If set to true, the Approval action is available for campaigns. The default value of this setting is true.

Adaptive learning

Setting name	Owning ruleset	Description
ADMShortTimeoutForDelayedLearning	*PegaMKT-Engine*	Duration in seconds to store results for adaptive learning. This setting is used for Real-Time Container scenarios. The default value of this setting is 86400 (1 day).
ADMTimeoutForDelayedLearning	*PegaMKT-Engine*	Duration in seconds to store results for adaptive learning. The default value of this setting is 2592000 (30 days).
MKTOptimizationModelThreshold	*PegaMKT-Engine*	Threshold by which an adaptive model's performance is compared for a self-optimizing campaign. If the model's performance is below this threshold, offer assignment will be handled randomly. Otherwise, the model will be utilized alongside random assignment. The default value of this setting is 0.65.

Setting name	Owning ruleset	Description
decision/adm/modelExecutions/format	*Pega-DecisionEngine*	Determines the common inputs in *pxDecisionResults* to store predictor data. Set to single-level-v1 by default, can be updated to multi-level-v1 when multiple common inputs are needed in a multi-level decision structure.

Campaign data flow

Setting name	Owning ruleset	Description
MKTDDFProgressNewTimeout	*PegaMKT-Engine*	Timeout duration in seconds used internally to determine if data flow execution is stuck in the New state, that is, if execution did not start correctly. The default value of this setting is 60.
MKTDDFProgressRefreshInterval	*PegaMKT-Engine*	Refresh duration in seconds used internally to monitoring the progress of campaign execution data flows. The default value of this setting is 30.
MKTDDFProgressRefreshTimeout	*PegaMKT-Engine*	Timeout duration in seconds used internally to determine if data flow execution is stuck while processing a batch of records. The default value of this setting is 600 (10 minutes).
MKTDFRetryInterval	*PegaMKT-Engine*	Retry duration in minutes used internally to wait before checking data flow availability again if there is no room for more concurrent data flow runs. The default value of this setting is 30.
MKTDFNumRetries	*PegaMKT-Engine*	Maximum number of times a campaign run will attempt to execute its data flow before the run fails. The default value of this setting is 5.
MKTMaxAllowedBatchDFs	*PegaMKT-Engine*	Number of data flows which can run on batch-type nodes at the same time. The default value of this setting is 10.

Channels

Setting name	Owning ruleset	Description
IsReceiptDecodingRequired	*PegaMKT-Engine*	Specify whether SMS delivery receipts decoding is required when connecting to an SMSC. The default value of this setting is false.
MKTChannelQueueStatsFormat	*PegaMKT-Engine*	Configure the format of the log created by the MonitorChannelQueue agent.
MKTEmailPartitionCount	*PegaMKT-Engine*	Number of email table partitions. The default value of this setting is 99.
MKTOutboundEmailMaxBatchSize	*PegaMKT-Engine*	Maximum number of emails sent in one batch. The default value of this setting is 1000.
MKTOutboundEmailMinBatchSize	*PegaMKT-Engine*	Minimum number of emails sent in one batch. The default value of this setting is 250.
MKTOutboundSMSMaxBatchSize	*PegaMKT-Engine*	Maximum number of SMS messages sent in one batch. The default value of this setting is 1000.
MKTOutboundSMSMinBatchSize	*PegaMKT-Engine*	Minimum number of SMS messages sent in one batch. The default value of this setting is 250.
MKTPushPartitionCount	*PegaMKT-Engine*	Number of Push Notification table partitions. The default value of this setting is 5.
MKTSMSPartitionCount	*PegaMKT-Engine*	Number of SMS table partitions. The default value of this setting is 10.
MaxSMSResponseDeliveryReceiptPartition	*PegaMKT-Engine*	Maximum value for PartitionKey in the SMS Response Delivery Receipt table. The default value of this setting is 20.
SMSFailMaxNumberOfRetries	*PegaMKT-Engine*	Maximum number of times the SMS message should be re-sent if it fails. The default value of this setting is 3.
NumberOfPartitions	*PegaMKT-Engine*	Number of buckets used internally to divide Push Notification processing items. Each bucket can be picked up individually by a node where the Push Notification Processor agent is running. The default value of this setting is 5.
RequestorBatchSize	*PegaMKT-Engine*	Number of Push Notification items to process at a time. The Push Notification Processor agent retrieves items until this size is met (or all items have been retrieved) before processing them. The default value of this setting is 1000.

Containers

Setting name	Owning ruleset	Description
MKTEnableCustomerLookup	*PegaMKT-Engine*	Toggle the loading of customer data during container execution. The default value of this setting is true.
MKTEnableCustomerValidation	*PegaMKT-Engine*	Toggle validating if a customer exists in the database during container execution. The default value of this setting is false.
MKTEnableIdentityMatching	*PegaMKT-Engine*	Toggle customer identity matching for container services. The default value of this setting is true.

Content repository

Setting name	Owning ruleset	Description
MarketingImageAssets/AllowedFileSize	*PegaMKT-Engine*	Size limit in KB for the digital assets uploaded to the image library. The default value of this setting is 100.
MarketingImageAssets/BaseURL	*PegaMKT-Engine*	URL of the Content Delivery Network for digital assets. If you leave this setting blank, the images will be rendered directly from the Pega Web Service.
MarketingImageAssets/CustomBaseURL	*PegaMKT-Engine*	URL of the Content Delivery Network which you configured for digital assets. This setting overwrites the value of the setting *MarketingImageAssets/BaseURL*.
MarketingImageAssets/CustomRepositoryName	*PegaMKT-Engine*	Name of the repository rule which points to your S3 repository. This setting overwrites the value of the setting *MarketingImageAssets/DefaultRepositoryName*.
MarketingImageAssets/DefaultRepositoryName	*PegaMKT-Engine*	Name of the default repository rule. The default value of this setting is pegacloudfilestorage.
MarketingImageAssets/Folder	*PegaMKT-Engine*	Name of the folder where Pega Customer Decision Hub will store the image assets. This folder will be located inside the root folder defined in the repository rule. If you do not define a folder for the images, they will be stored directly in the repository root folder. The default value of this setting is pegamarketing/imagelibrary.
MarketingImageAssets/IsRepositoryStorageEnabled	*PegaMKT-Engine*	Set to true if you want Pega Customer Decision Hub to use repository storage for your image library. The default value of this setting is false.

Microsites

Setting name	Owning ruleset	Description
microsite/URL	*PegaMKT-Engine*	URL which the Microsite gadget uses when connecting to Pega Customer Decision Hub. The URL should point to an IAC servlet on the public node. The URL format is http://*<hostname>*:*<port>*/prweb/IAC.
microsite/applicationname	*PegaMKT-Engine*	Application name which the Microsite gadget uses when connecting to Pega Customer Decision Hub. By default, it is the name of the application associated with the access group on the *PegaMKTMicrosite* service package.
microsite/systemid	*PegaMKT-Engine*	System ID which the Microsite gadget uses when connecting to Pega Customer Decision Hub. By default, it is the system ID defined on the system which the URLs point to.
microsite/defaultclass	*PegaMKT-Engine*	If the Site parameter of the Microsite URL is not defined, this setting purpose is used instead. The default value of this setting is PegaMKT-Work-Microsite_NOSITENAMESPECIFIED_.
microsite/parameters/useridentifier	*PegaMKT-Engine*	The user which the Microsite gadget uses to authenticate itself with Pega Customer Decision Hub.
microsite/parameters/skinname	*PegaMKT-Engine*	Name of the skin used by the Microsite.

Paid Media Manager

Setting name	Owning ruleset	Description
paidmedia/syncService/GoogleAdsAudienceAsyncJobFailTime	*PegaMKT-Engine*	Time in hours for Google Audience sync jobs. If this is not set, the default value of this setting is 6 hours.
paidmedia/syncService/GoogleAdsConversionAsyncJobFailTime	*PegaMKT-Engine*	Time in hours for Google Conversion sync jobs. If this is not set, the default value of this setting is 48 hours.
paidmedia/ResponseWaitingTime	*PegaMKT-Engine*	Timeout period in seconds for offers in paid channels. If an individual does not respond to an offer within the specified time period, the system automatically records a negative response. The default value of this setting is 604800 (1 week).
paidmedia/adobe/api/authserver/url	*PegaMKT-Engine*	Auth Server URL for Adobe Audience Manager. The default value of this setting is https://api.demdex.com/oauth/token.
paidmedia/adobe/api/resourceserver/url	*PegaMKT-Engine*	Resource Server Base URL for Adobe Audience Manager. The default value of this setting is https://api.demdex.com:443/v1.
paidmedia/adobe/traitsegment/toplevel/folder/name	*PegaMKT-Engine*	Name of the top level segment and trait folder created under the parent folder in Adobe Audience Manager. The default value of this setting is Paid Audiences.
paidmedia/adobe/file/push/maxfilesizeinbytes	*PegaMKT-Engine*	Maximum size in bytes of synchronization files sent to Adobe Audience Manager. The default value of this setting is 1000000000.
paidmedia/adobe/file/push/retrycount	*PegaMKT-Engine*	Number of times Paid Media Manager automatically tries to resend the synchronization file to Adobe Audience Manager in case of any synchronization problems. The default value of this setting is 3.
paidmedia/adobe/file/directory/base	*PegaMKT-Engine*	Base directory where the Adobe Audience Manager file would be pushed by the data flow run.
paidmedia/adobe/file/push/dir/cleanup/staletimeinms	*PegaMKT-Engine*	Time in milliseconds during which the files are kept in the Adobe Audience Manager file directory. The default value of this setting is 691200000.
paidmedia/linkedin/WaitTimeInMillis	*PegaMKT-Engine*	LinkedIn wait time in milliseconds. The default value of this setting is 60000.
paidmedia/linkedin/api/resourceserver/url	*PegaMKT-Engine*	LinkedIn API resource server endpoint. The default value of this setting is https://api.linkedin.com/v2.

Setting name	Owning ruleset	Description
paidmedia/syncService/requestBatchSize	*PegaMKT-Engine*	Defines the number of records which Paid Media Manager batched within a Paid Media Sync queue item before it prepares an API call to supported networks such as Google or Facebook. The default number of records is 10000. The more records there are in a queue item, the less API calls there will be. Adjust this value depending on the memory capacity available in the cluster.

Segments

Setting name	Owning ruleset	Description
MKTSegCriteriaInsertRequestorCount	*PegaMKT-Engine*	Total number of child requestors that are used during the segment runtime. The default value of this setting is 1.
MKTSegCriteriaQueryTimeout	*PegaMKT-Engine*	Time in seconds for an insert operation to the Criteria query for segment runtime. The default value of this setting is 1200 (20 minutes).
MKTSegDeleteRepopulateQueryTimeout	*PegaMKT-Engine*	Meant for the 'DELETE' portion of query in segments. Different timeouts values are use to have a better control for customers. The default value of this setting is 1200 (20 minutes).
MKTSegDisplayPreAggregateCounts	*PegaMKT-Engine*	Used by segment In List criteria to determine if a second column should be displayed with counts for each distinct value. The default value of this setting is true.
MKTSegGroupByBothColumns	*PegaMKT-Engine*	Enables grouping by both keys for aggregate segment criteria. The default value of this setting is true.
MKTSegIgnoreCaseDefaultSetting	*PegaMKT-Engine*	Default value for the **Ignore Case** setting for String segment criteria. If true, the **Ignore Case** check box is selected by default. The default value of this setting is false.
MKTSegOracleDeleteHint	*PegaMKT-Engine*	Used to specify the DELETE hint used during segment runtime for Oracle DELETE statements.
MKTSegOracleInsertHint	*PegaMKT-Engine*	Used to specify the INSERT hint used during segment runtime for Oracle INSERT statements.
MKTSegOracleSelectHint	*PegaMKT-Engine*	Used to specify the SELECT hint used during segment runtime for Oracle SELECT statements.
MKTSegPartitionCount	*PegaMKT-Engine*	Number of segment table partitions. The default value of this setting is 99.
MKTSegRevalidateCustomers	*PegaMKT-Engine*	Specify whether or not segment groups should be validated against the customer table using an INTERSECT statement. Validating segment groups ensures that all customers in a segment still have a valid ID in the customer table, but may result in a longer segment runtime. The default value of this setting is false.
MKTSegmentRefreshInterval	*PegaMKT-Engine*	Refresh duration in seconds used internally for monitoring the progress of segment execution data flows. The default value of this setting is 10.
ProspectListImportRetries	*PegaMKT-Engine*	Maximum number of times the prospect import should run if it fails. The default value of this setting is 15.

Other configuration

Setting name	Owning ruleset	Description
HelpURI/MarketingHelp	*PegaMKT-Engine*	Defines the baseline URL for context-sensitive help links. These help links are available in various places in the application, for example, as **Get Help** links (in the **Actions** menu) or **Learn more** links. They provide content that is relevant to what the user is doing in the context of a specific landing page or window. The value of this DSS is https://community.pega.com/cdhhelp_v*XX*, where *XX* is the application version, for example, 88. Through a redirect mechanism that is external to the application, the baseline URL ensures that the help links always point to the right version of the content on the documentation site.
MKTConstraintAsGroup	*PegaMKT-Engine*	Specify if a customer constraint should be applied to your whole application. The default value of this setting is false.
MKTDefaultTemplateTextLength	*PegaMKT-Engine*	Default size of text columns created in Database Template tables. This value is only used for those properties which do not have maximum length specified. For individual cases, it is recommended to set the maximum length on the property rule rather than update this setting. The default value of this setting is 200.
MKTDefaultThreadCount	*PegaMKT-Engine*	The default thread count used for multi-requestor processing on a node where the topology has not been configured. The default value of this setting is 2.
MKTDeliveryMatchingMaximumTimeToLive	*PegaMKT-Engine*	Duration in seconds for an unmatched offer or batch table to delivery. The default value of this setting is 300.
MKTDistinctValuesMaxResults	*PegaMKT-Engine*	Maximum number of distinct values to retrieve. The default value of this setting is 500.
MKTFileDelimiter	*PegaMKT-Engine*	Character or characters used by file templates as delimiter. The default value of this setting is , (comma).
		Default value of the **Create funnel analysis reports**

Setting name	Owning ruleset	Description
MKTFunnelAnalysisReportsDefault	*PegaMKT-Engine*	setting for campaign distribution tests. If true, the **Create funnel analysis reports** check box is selected by default when starting a campaign distribution test. The default value of this setting is true.
MappingLicenseKey	*PegaMKT-Engine*	License key required for premium GeofenceMap capabilities.
StrategiesRequireReindex	*PegaMKT-Engine*	Flag to determine whether strategy rules which are shipped in *PegaMKT-Engine* should be re-indexed. Typically used by the *PegaMKTStartupAgent* to check whether missing index records require re-generation. The default value of this setting is false.
WaitAgentReQueueTime	*PegaMKT-Engine*	Duration in seconds used internally to re-queue wait agent processing if a campaign run table still has records that are in a pending state. The default value of this setting is 600 (10 minutes).
database/databases/ExternalMKTData/dataSource	*PegaMKT-Engine*	Name of the JNDI datasource, used when *ExternalMKTData* database rule is set to **Use preferences** mode.
MaxRecentsCount	*PegaMKT-Portal*	Number of items to show in the **Recents** pane in the portal. The default value of this setting is 10.

Pega Customer Decision Hub access roles and privileges

Pega Customer Decision Hub uses access privileges to manage various access rights for operators. As an administrator, you can add or remove privileges to or from access roles to manage the access rights for various access groups in your application.

Pega Customer Decision Hub

Privilege name	Description	
CanSeeMKTManagerDashboard	Grants the ability to access the Manager dashboard.	
CanSeeMKTAdminDashboard	Grants the ability to access the Admin dashboard.	
CanAccessCampaigns	Grants the ability to access campaigns and the **Campaigns** landing page.	
CanAccessContactPolicies	Grants the ability to access contact policies.	
CanAccessDataFlows	Grants the ability to access data flows and the **Data Flows** landing page.	
CanAccessDataMenu	Grants the ability to access the **Data** menu.	
CanAccessMarketingCalendar	Grants the ability to access the Marketing Calendar and the **Marketing Calendar** landing page.	
CanAccessMarketingContent	Grants the ability to access marketing-specific content.	
CanAccessMicrosites	Grants the ability to access microsites and the **Microsites** landing page.	
CanAccessProspectLists	Grants the ability to access prospect lists and the **Prospects** landing page.	
CanChangePortals	Grants the privilege to switch portals.	PegaNBAM_FW:Marketi
CanConfigurePaidMedia	Grants the ability to create and update paid media settings.	
CanCreateMicrosite	Grants the ability to modify the application rule and create new microsites.	
CanCreateNBADTemplate	Grants the ability to create templates in Next-Best-Action Designer.	
CanManageNBA	Grants the ability to modify the Next-Best-Action Designer configuration and model.	
CanPublishToDefault	Grants the ability to configure default settings for the Marketing Dashboard in the Pega Customer Decision Hub portal.	
CanResumeFlow	Grants the ability to resume a flow by using a flow action. This is privilege must be available to Pega Customer Decision Hub agents through the PegaNBAM_FW:Batch access role. For testing purposes, this privilege is also enabled for developer operators.	

Privilege name	Description	
CanViewAppExplorer	Grants the ability to access the Application Explorer in the Pega Customer Decision Hub portal.	
CanViewMKTDeveloperShortcuts	Grants the ability to access developer shortcuts in the Pega Customer Decision Hub portal.	
ConfigureDecisioning	Grants the ability to configure configuration of Decisioning. Used to conditionally show menu items from the navigation rules.	
CDHLoggedIn	Allows to select the setting **Allow direct invocation from the client or a service** for Activity rules when necessary, usually when the activity is called from a script.	CustomerDecisionHub:AdministratorCustomerDecisionHub:AnalystDecisionManager:CDHDecisionManager:PegaMarketingP
MarketingManager	Used in the **MKTApplicationMainMenu** to show Usage History to managers. Used by the **PUMADecisionMainMenu** to show Batch Data to managers.	
MKTApplicationConfiguration	Used in the **Configuration** menu of the Pega Customer Decision Hub portal to enable access to Application Configuration and various admin-level landing pages.	
MKTApplicationConfiguration	Grants the ability to access the Application Configuration landing page.	
MKTNotificationConfiguration	Grants the ability to access the Push Notifications landing page.	
MKTOverrideExclusion	Grants the ability to override the Global Exclusion List configured for the application.	
ReadOnly	This privilege is used as the update privilege to simulate read-only access. It requires no access roles.	
ReadOnly	This privilege is used as the update privilege to simulate read-only access. It requires no access roles.	
ViewAlerts	Grants the ability to access the **Alerts** shortcut in the footer.	
ViewInspectionPrefs	Grants the ability to access the **Inspection Prefs** shortcut in the footer.	
ViewMAOperatorMenu	Grants the ability to access the **Operator** menu in the Pega Customer Decision Hub portal.	
ViewPAL	Grants the ability to access the **Performance** shortcut in the footer.	
ViewPDN	Grants the ability to access the **PDN** shortcut in the footer.	
SocialAdminAccess	Grants the ability to access the Social Engagement landing page.	
pzBiasPolicyConfiguration	Grants the ability to access the Ethical Bias Policy landing page.	

Adding new response outcomes

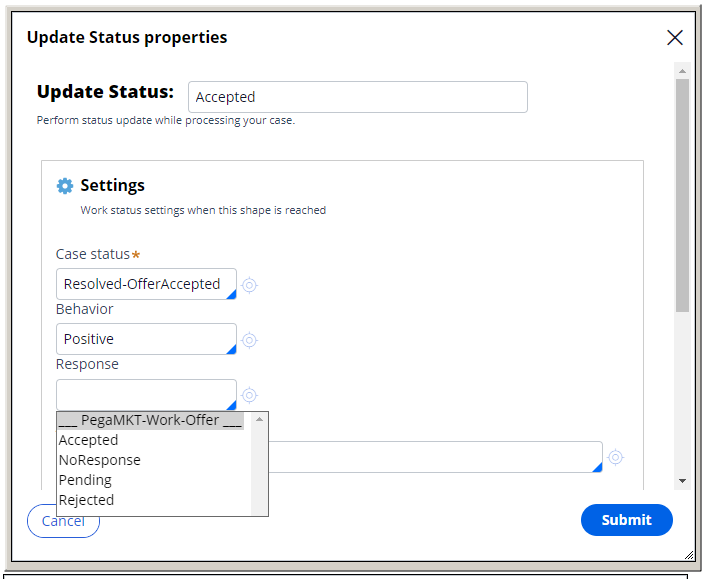
You can add new, customized response outcomes of customer interactions by configuring the Contact policy and Update Status shape in action flows.

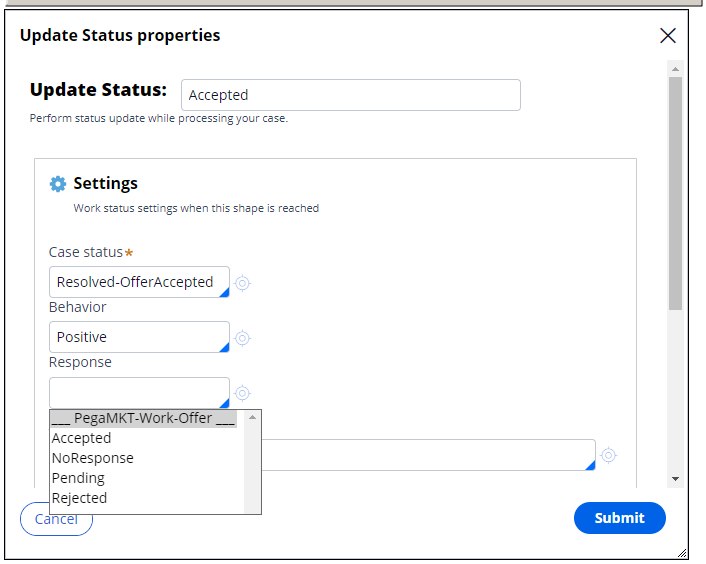
Pega Customer Decision Hub Pega Customer Decision Hub Pega Customer Decision Hub

Customers' responses to actions such as accepted, rejected, and clicked are recorded in the Interaction History as outcomes. Responses to interactions initiated through real-time containers are received through real-time container APIs. In this case, external channels specify the response outcomes: accepted, rejected, clicked, and so n, when they make the API call.

Action flow Update Status shape

Responses are also recorded by the Update Status shape in action flows. This approach is typically used when the action receives a response to an outbound email or SMS.

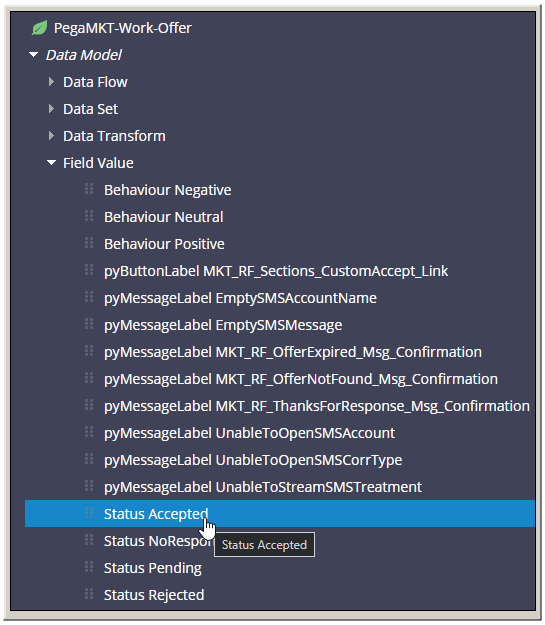


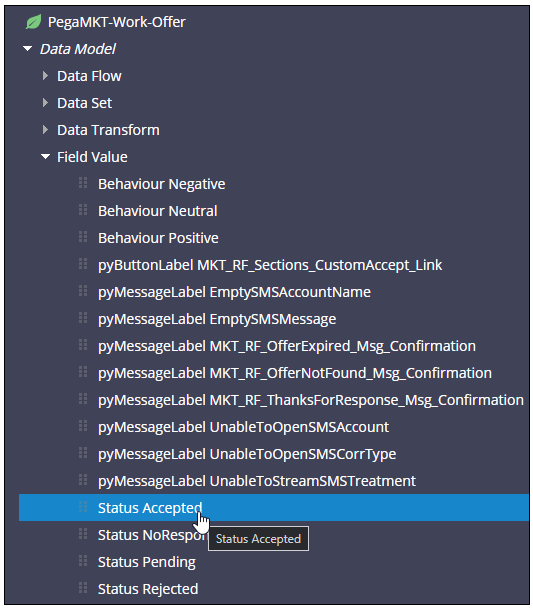


In the Update Status shape, you can type the response value or select the value from a drop-down list. Manually typing the value could inadvertently introduce new values into the Interaction History, hence it is recommended to use the drop-down list.

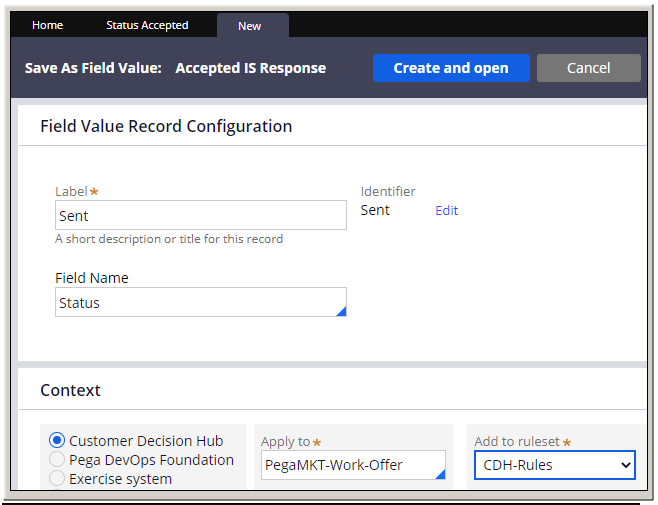
To add new response outcomes to the **Response** drop-down list in the Update Status shape, create a new Field Value on the *Status* property in the *PegaMKT-Work-Offer* class.

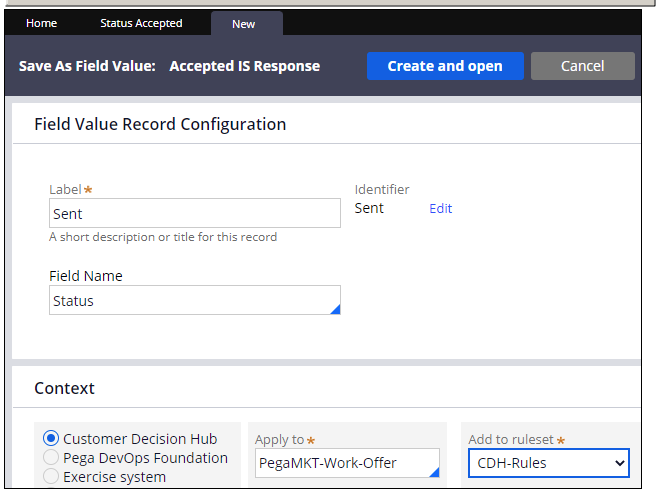
1. Log in to your Pega Customer Decision Hub implementation layer application as an operator with access to Dev Studio.
2. In the Dev Studio, search for and open an existing Field Value on the *Status* property in the *PegaMKT-Work-Offer* class.
   For example **Status Accepted**.



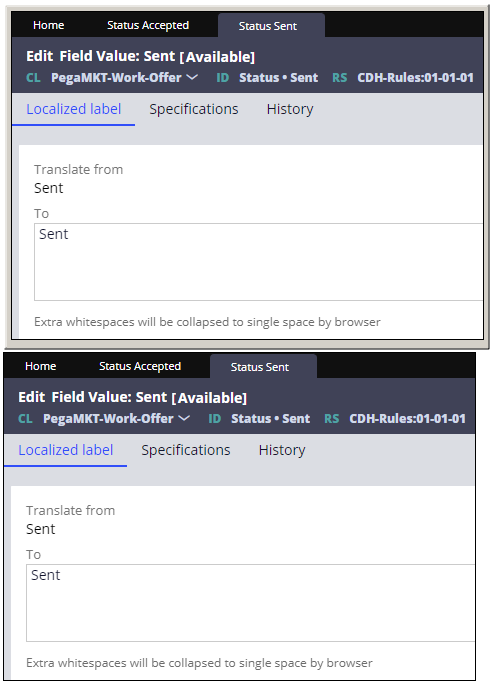


3. Copy the Field Value using **Save As** into your **Rules* ruleset.

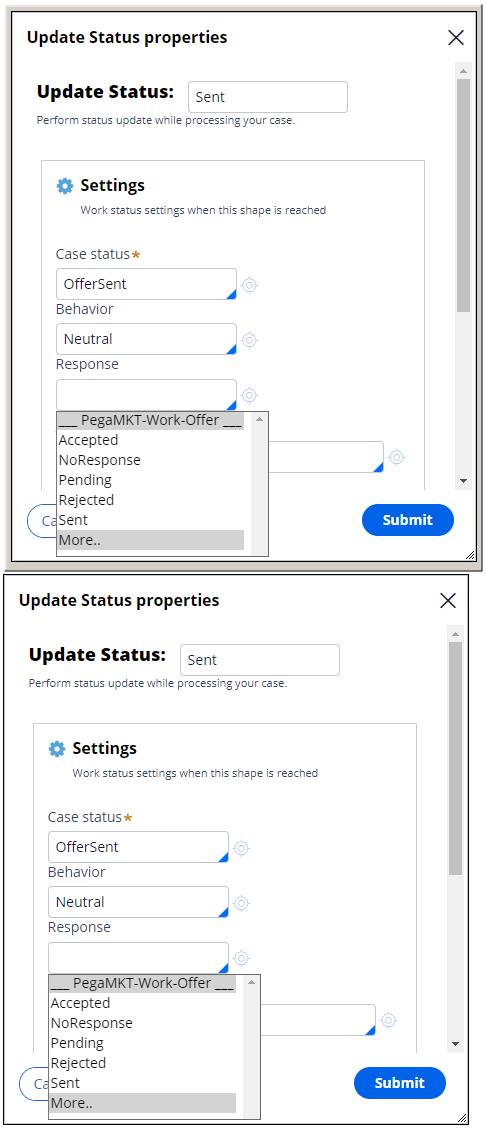




4. **Optional:** Update the **To** value to the new value and check in.
   Note: It is the value in the **Translate from** section, that is displayed in the UI. So, this step is optional and recommended for better housekeeping.

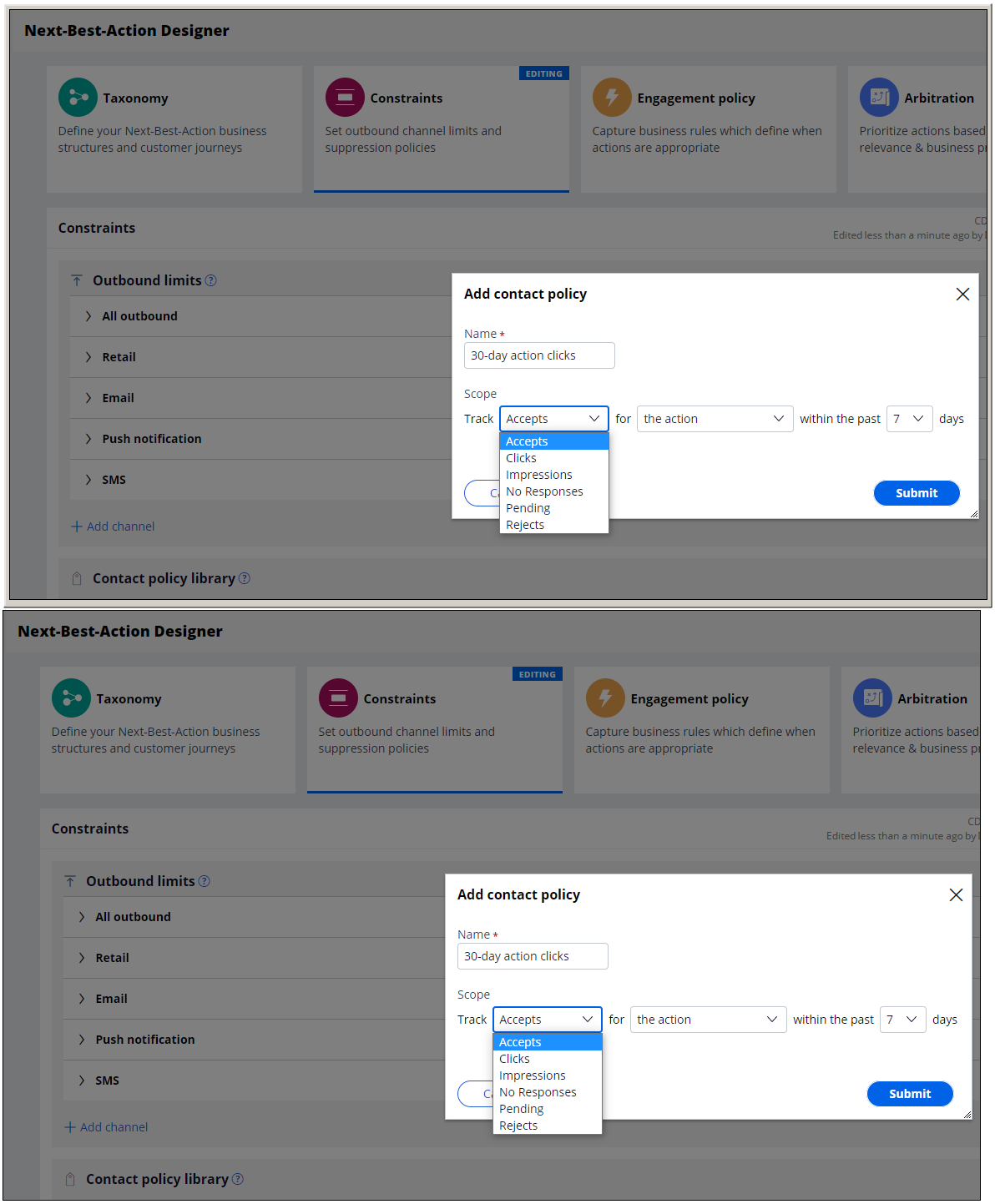


Result: A new response outcome is added to the Update Status shape in the action flow.



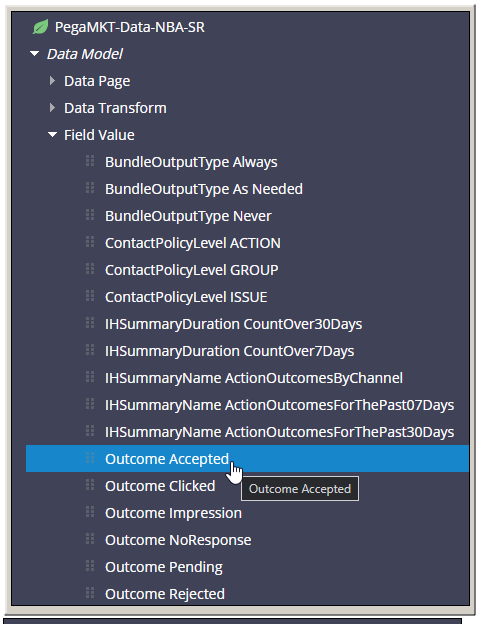
Contact policy

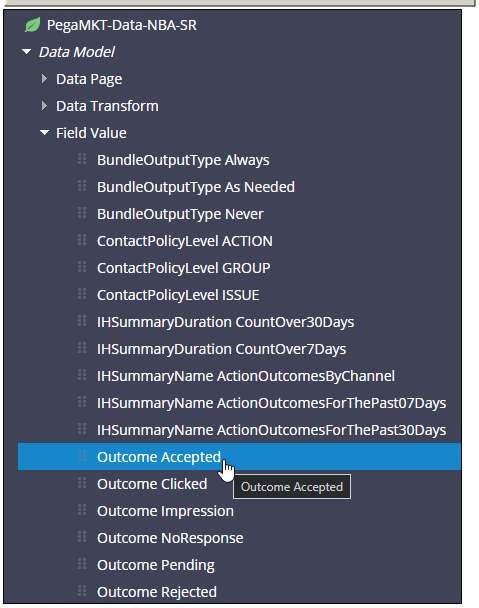
A drop-down list with response outcomes appears when you create a contact policy in the Next-Best-Action Designer.



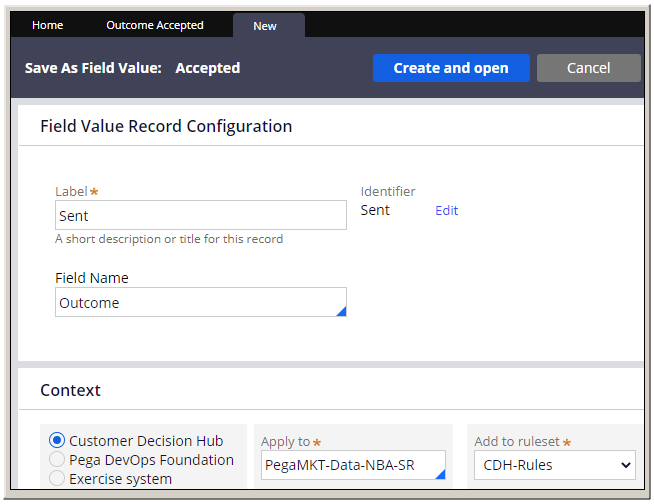
To add new response outcomes to the Contact policy, create a new Field Value on the *Outcome* property in the *PegaMKT-Data-NBA-SR* class.

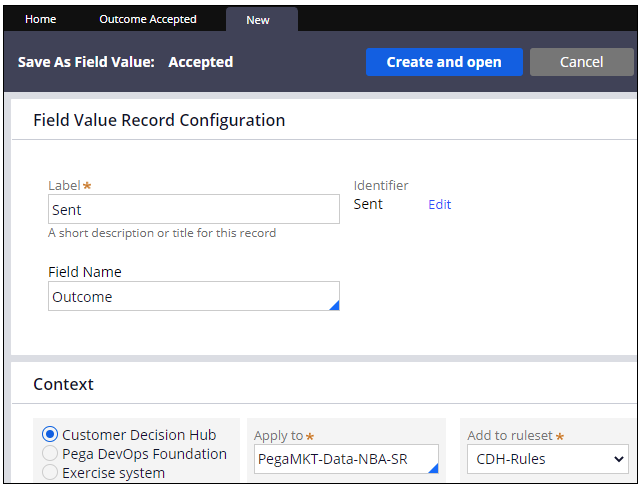
1. Log in to your Pega Customer Decision Hub implementation layer application as an operator with access to Dev Studio.
2. In the Dev Studio, search for and open an existing Field value on the *Status* property in the *PegaMKT-Data-NBA-SR class*.
   For example *Status NoResponse*.



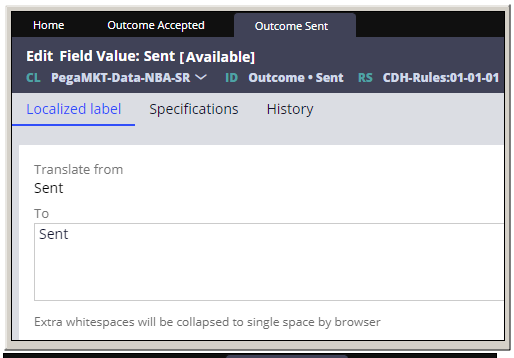


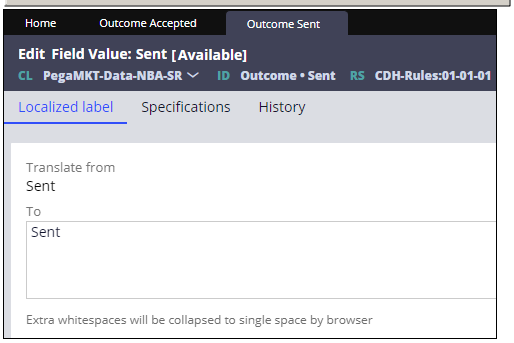
3. Copy the Field Value using **Save As** into your **Rules* ruleset.



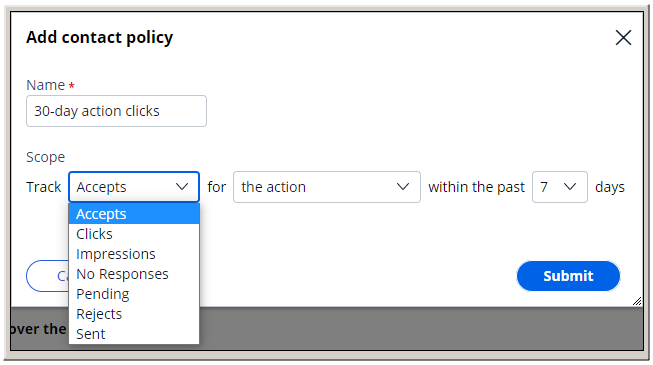


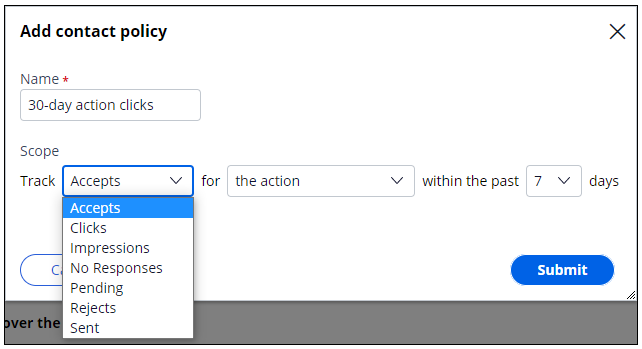
4. **Optional:** Update the **To** value to the new value and check in.
Note: It is the value in the **Translate from** section that is displayed in the UI. So, this step is optional and recommended for better housekeeping.





5. Click **Create and open**. **Result:** A new response outcome is added to the Contact policy **Track** drop-down list.





Disabling the Decision Data Store as action state storage on Pega Cloud

New deployments of Pega Customer Decision Hub on Pega Cloud services are configured to use the Decision Data Store (DDS) to store action state. The DDS data sets manage large and active data from different sources and make it available for real-time and batch processing.

When the DDS is used to store action data, the following features are disabled:

- Volume Constraints
- Wait volume overlays
- Campaign run restarts
- Direct output to template
- Prospects

If you want to use these features, you need to disable the DDS as action state storage, and then configure the campaigns and scheduled runs to write data to database tables.

Pega Customer Decision Hub
Important: Perform this task during a maintenance window.

1. Disable the DDS service as action state storage:
   a. In the navigation pane of App Studio, click **Settings Application settings Customer Decision Hub** .
   b. Clear the **Use Decision Data Store for Action state storage** check box, and then click **Save**.
2. Go to Pega Customer Decision Hub portal to switch existing Campaigns and Next-Best-Action Designer Scheduled runs to write to database tables:
   Note: In progress Campaigns and Scheduled runs continue to run and store Action data in the Decision Data Store.

Choices	Actions
For each existing Scheduled Run (both Primary and Secondary schedules included)	a. If the Scheduled Run is running, suspend it by going to: **Next-Best-Action Designer Channels** . b. Click **More Suspend** . c. Edit the **Recurrence** section and set a valid recurrence schedule. d. Launch the Scheduled run by clicking **More Run** . **Result:** New runs for this Schedule will now store Action state data in database tables.
For each existing Multi-channel Campaign	a. If the Campaign is running, suspend it by going to **Campaigns** menu in the left navigation pane and clicking on **Campaigns** b. Choose the individual campaign you want to suspend, and click **More Suspend** . c. Click the **Edit** button, and then update the **Engagement** section to a valid recurrence schedule if needed. d. **Save** the Campaign. You must edit and re-saved the Campaign even if there is no change required to the schedule. e. Launch the Multi-channel Campaign by clicking the **Run** button. **Result:** New runs for this work object will now store Action state data in database tables.
For each existing Outbound Campaign	a. If the Campaign is scheduled for the future, suspend the Campaign by going to **Campaigns** menu in the left navigation pane and clicking on **Campaigns** b. Choose the individual campaign you want to suspend, and click the **Suspend** button. c. Click the **Edit** button, and then **Save** the Campaign. You must edit and re-saved the Campaign even if there is no change required to the schedule. d. Schedule the Outbound Campaign by clicking **Actions Run** . **Result:** The Campaign Run for this work object will now store Action state data in database tables.

3. Stop the *DF_ProcessWaitingActions* real-time data flow to free up resources.
   Caution: Perform this step only when all runs have been completed and there are no waiting Actions that are written to the Decision data store. Performing this step sooner could result in runs not completing properly and waiting Actions not expiring as expected.
   a. In the header of Dev Studio, in the search box, search for the class name Pega-DM-DDF-Work.
   b. Click **Pega-DM-DDF-Work**.
   c. Filter the ID column by DF_ProcessWaitingActions.
   d. Open the remaining instance with the ID DF_ProcessWaitingActions and then click **Stop**.

Pega Customer Decision Hub agents

On top of the standard Pega Platform agents, exist a set of agents that manage the different aspects of Pega Customer Decision Hub. Review and monitor these agents on a regular basis to ensure that your application is running optimally. In the event of a failure in any of the agents, understanding the agent's role helps to assess the impact of the failure on the application.

Pega Customer Decision Hub

The following tables contains descriptions of these agents, which are divided based on the part of the application that the agent is used in. For each of the agents listed, the Multi-node column contains a recommendation on whether the agent should be run on multiple nodes, based on the purpose of the agent and how it scales for performance.

- Segmentation agents- manage running segments
- Offer runtime agents- manage running offers and real-time events
- Delivery channels agents- manage delivery channels, such as email and SMS
- Paid media job schedulers- manage advertising on ad platforms such as Facebook Ads or Google Ads
- Cleanup and data-mining agents- clean up batch assets that are created while running a program; perform data mining
- Startup and maintenance agent- handles routines, maintenance, and checks that must occur during application server start up
- Deprecated agents

Segmentation agents

Agent name	Multi-node	Description	Node type
		Handles requests to manually run a segment from the Segment rule form. If there are many manual requests for larger segments, then having this agent running on multiple nodes speeds up segment runtime.	

Agent Name	Multi-node	Description	Node type
DeferredSegments	Yes	Queue items for this agent are instances of the *System-Queue-DeferredSegments* class. If there are Broken-Process queue items for this agent, marketers might experience failures or delays in their segments. Review the details on the queue item to determine the cause of the failure.	BackgroundProcessing
SegmentSchedule	Yes	Manages the scheduled run of segments, which is configured in the **Options and Schedule** tab of the Segment rule form. If there are recurring runs, the agent auto-manages its own queue items. There should be only one queue item per segment at any given point in time. Queue items for this agent are instances of the *System-Queue-ScheduledTask* class with the Agent Name set to *PegaMKT-Engine:SegmentSchedule*. Broken-Process queue items for this agent imply failures in scheduled running of segment. Consequently, these segments might have stale or incomplete data.	BackgroundProcessing
ManageMasterSegment		Keeps the Master Segment view in sync with all the segments currently created in the system. The Master Segment is used by Pega Customer Decision Hub 8.2 and older to support the segment filter functionality. In Pega Customer Decision Hub 8.3, this agent still exists but is no longer used for segment filter support.	BackgroundProcessing

Offer runtime agents

Agent Name	Multi-node	Description	Node type
ProcessOverlayAction	Yes	Performs background processing of the Update wait time and Stop action settings on the Action rule. For more information, see Monitoring actions.	BackgroundProcessing
ProcessProgramRun	Yes	Initiates a campaign run at runtime. The agent creates the necessary infrastructure, such as batch classes and tables, and executes the strategy based on the segment input population that is configured. If configured on the campaign, it also triggers the refresh/rerun of the segment before starting a strategy. In a non-partitioned environment, this agent applies volume constraints. Queue items for this agent are instances of the *System-Queue-ProgramRun* class. If the campaign run work object does not show any progress for a long time, review the status of this agent and its associated queue items.	Batch
ProcessEventRun	Yes	Handles the processing of received real-time events. The agent reads the list of received events from a staging table, processes them, and then removes them from the table. Processing an event involves: • Checking the validity of the inbound event data for each of the associated campaigns • Checking the validity of the customer (if configured) • Assembling the appropriate context • Executing the strategy associated with the campaign • Executing offers that the strategy outputs Event campaigns use the *MKT_EVENT_P<ProgramID>* tables instead of the batch output tables.	RealTime
ProcessWaitFlows	Yes	Triggers an action flow for the WaitExpired paths in an action flow. Every campaign run creates one or more (if partitioned) of these queue items to periodically scan the batch table for records whose wait duration has expired. If any such actions are found, this agent resumes their processing along the WaitExpired path. Queue items for this agent are instances of the *System-Queue-WaitFlow* class.	Batch
		When actions are initiated in a non-batch scenario (such as Inbound), the records are stored in the *MKT_DATA_OFFER* table in the data schema. This agent scans through such actions to find any that have	

Agent Name	Multi-node	Description	Node type
ProcessWaitingGenericFlow	Yes	expired and resumes their processing along the WaitExpired path. Queue items for this agent are instances of the *System-Queue-GenericWaitFlow* class.	RealTime
RunHandleResponsesDF		Supports updating the adaptive model responses based on the stages and timeouts configured in the outcome optimization configuration. The agent checks if a real-time data flow run is in progress for the *pxHandleResponses* data flow. If not, it creates one and starts the data flow run.	Batch
ProcessWaitingNBAOffers	No	Periodically checks if there are actions that need Wait Expired configured for the Next-Best-Action Designer triggered events. Primarily operates on the table/class that is mapped to: *PegaMKT-Data-NBAOffer*.	RealTime

Delivery channels agents

Agent Name	Multi-node	Description	Node type
SendEmailsInBulk	No	Periodically reads the email output table PR_DATA_CORR_EMAIL and performs email transmission. The agent removes successfully transmitted emails and, in the event of a failure, updates the status to Failed.	Custom1, Batch
MonitorChannelQueue	Yes	Updates the log file used to monitor the email queue. Pega Customer Decision Hub can log sent, failed, and queued emails, as well as the queue depth, that is, the emails which have not been processed yet. By default, the agent updates the log file at one-hour (3600 seconds) intervals.	BackgroundProcessing
SendSMSInBulk	No	Periodically reads the SMS output table PR_DATA_CORR_SMS and performs SMS transmission. The agent interfaces with the configured SMSC and transmits the SMS message. In the event of a failure, the agent removes any successfully transmitted SMS messages and updates the status to Failed.	Custom2, Batch
InboundSMSReceiver	No	Periodically sets up listeners to handle inbound SMS messages for the configured Inbound SMS Accounts. The agent calls the InboundConnector activity to set up the listener. For each message that is received, the agent creates a new instance of *PegaMKT-Data-Inbound-SMS*, populates the appropriate details, and saves it to the database.	RealTime
InboundSMSOfferFlow	No	Processes instances of *PegaMKT-Data-Inbound-SMS*. For each message, the agent determines the relevant Customer and SMS Account context, parses the inbound message based on the SMS Account configuration, uses this information to determine the behavior and action, and (if appropriate) initiates/resumes the action.	RealTime
OutboundMaster	No	**Note:** Used for backward compatibility and upgrade purposes.A master outbound agent that determines which File/Database Templates need to be processed, and then creates queue items for the WriteOutboundAgent to process them.	Batch
WriteOutboundAgent	Yes	**Note:** Used for backward compatibility and upgrade purposes.A standard agent that performs File and Database output processing based on the details in the queue item created by the OutboundMaster agent. WriteOutboundAgent processes records from the MKT_STAGING table in the Pega Customer Decision Hub schema.	BackgroundProcessing
FileDBSchedule	Yes	**Note:** Used for backward compatibility and upgrade purposes.Performs the scheduled finalization for the File and Database output channels.	BackgroundProcessing
PushNotificationMaster	No	Periodically checks the push notifications table PR_DATA_PUSH_MESSAGE to create new queue items for the PushNotificationProcessor agent in order to transmit notification messages. Each queue item corresponds to a (logically simulated) partition value in the table.	BackgroundProcessing

Agent Name	Multi-node	Description	Node type
PushNotificationProcessor	Yes	Processes messages that are part of a specific partition in the PR_DATA_PUSH_MESSAGE table. The agent transmits these messages based on the configurations for the Push Notification App and Variant (specified on the Push Notifications landing page).	BackgroundProcessing
File/DB template based job schedulers	No	These are not enabled out-of-the-box, but when a file or DB template is configured with schedule options and then saved, the new job schedules with the name of the template rule and the channel appended will be created in the artifacts ruleset. For example, if the name of DB template was OutboundDailyNBA, then the job scheduler will be created, in the configured artifacts ruleset, with the name: OutboundDailyNBA__DB.	BackgroundProcessing

Queue processor

Agent Name	Multi-node	Description	Node type
FinalizeFileDBTemplate		This queue processor is disabled by default to help optimize performance. However, if your file and database templates must be processed at extremely high volume, this dedicated queue processor can be enabled to further increase the performance. For more information, see Optional: Enabling the FinalizeFileDBTemplate queue processor.	BackgroundProcessing

Paid media agents

Note:

Paid media agents are disabled by default. Refer to the table below for information about when to enable each agent.

Agent Name	Multi-node	Description	When to enable	Node type
ProcessPaidMediaRuns	Yes	Initiates a paid media run for paid media queue items created by campaign runs when paid destinations are configured in the Next-Best-Action hierarchy. After the paid media run starts, data flows are created for each paid destination in the hierarchy. Once all data flow runs are either completed or failed, work objects are created for any errors that occurred during the run. For more information, see Extending next-best-action to digital advertising platforms with Paid Media Manager .	After you configure paid destinations in the Next-Best-Action hierarchy.	Batch
ProcessPaidSegmentRuns	Yes	Initiates a paid segment synchronization after the refresh of a segment that has paid destinations enabled. For more information, see Extending next-best-action to digital advertising platforms with Paid Media Manager.	After you configure any paid segment synchronization.	Batch
ProcessPaidMediaReSync	Yes	Processes the resynchronization queue for paid destinations. Resynchronization queue items are created for each update to the PII mappings used by Paid Media Manager or to an individual's PII values, such as an email address, phone number, or device ID. The ProcessPaidMediaReSync agent processes the queue item and triggers the resynchronization for the right destination. By default, this agent runs every 5 seconds. For more information, see Extending next-best-action to digital advertising platforms with Paid Media Manager.	After you configure any paid destinations.	Batch
PaidMediaArtifactsMaster	No	Creates one queue item for each destination where you enabled automatic campaign generation. PaidMediaArtifactsGenerator must be enabled to process the queue items created by this agent. By default, this agent runs every 30 minutes. For more information, see Extending next-best-action to digital advertising	If you enabled automatic campaign generation in the paid destination settings.	BackgroundProcessing

Agent Name	Multi-node	Description	When to enable	Node type
		platforms with Paid Media Manager.		
PaidMediaArtifactsGenerator	Yes	Creates Facebook Ads and Google Ads lookalike audiences as well as campaign and ad set scaffolding. By default, this agent runs every 30 seconds and processes any queue items created by PaidMediaArtifactsMaster. For more information, see Extending next-best-action to digital advertising platforms with Paid Media Manager.	If you enabled automatic campaign generation in the paid destination settings.	BackgroundProcessing
PaidCustomerReSyncMaster	No	Identifies the changes in individual PII data, such as email addresses and phone numbers. If you update an individual's email address or phone number, at the next agent run, the agent triggers a re-synchronization of customer data. For each unique paid destination, the agent creates one CustomerReSync data flow run. Outbound, real-time and segment-based paid audience synchronization is postponed while the CustomerReSync data flow runs. If a synchronization is in progress, the CustomerReSync data flow for that destination is postponed until the current synchronization is completed. By default, this agent runs once every 7 days. For more information, see Extending next-best-action to digital advertising platforms with Paid Media Manager.	After you configure any paid destinations.	Batch
UpdatePaidMediaMatchRate	No	Updates the match rates for paid audiences visible on the Paid Audiences landing page. By default, this agent runs once every hour. For more information, see Extending next-best-action to digital advertising platforms with Paid Media Manager.	After you configure any paid destinations.	Batch

Paid media job schedulers

Note:

Paid media job schedulers are disabled by default. Refer to the table below for information about when to enable each job scheduler.

Job Scheduler Name	Description	When to enable
GetPaidAsyncJobStatus	It polls the status of Google sync Jobs from the Google ads Network. By default, this scheduler runs once every hour.	After you configure any Google paid destinations. For more information, see Extending next-best-action to digital advertising platforms with Paid Media Manager.
PaidNotificationsGenerator	Creates error and notification work objects. By default, this scheduler runs once every hour.	After you configure any paid destinations. For more information, see Extending next-best-action to digital advertising platforms with Paid Media Manager.
TokenValidityChecker	Checks the validity of the tokens used to communicate with paid destinations. If the token is expired or invalid, the scheduler creates an error work object with the default urgency of 80, and displays a notification in the notifications section of Next-Best-Action Designer. For segment synchronization, the notification is displayed in the notifications section of the Paid Sync tab of the segment.	After you configure any paid destinations. For more information, see Extending next-best-action to digital advertising platforms with Paid Media Manager.
OfflineConversionRTTrigger	Runs at regular intervals to read the real-time offline events from the sync staging table and synchronized them to the network. By default runs every hour.	When you want to sync the offline events, captured through *Capture Response* REST API to the network.
RemoveStalePaidMediaEntries	Audience sync creates distributed map entry to avoid the rate-limiting error from other nodes. In 8.5 distributed map entries persisted in Redis DB, even the full cluster restart will not clear these entries. This job scheduler runs daily to identify the entries created by a failed node and removes them from the distributed map.	When running paid media sync the audience to paid network.
PurgeStaleAdobeFolders	Cleans up stale Adobe folders generated during the sync process. It is scheduled to run daily at 11PM EST.	When running paid media sync for Adobe destination.
RemovePaidMediaStaleAudiences	It will run twice in a day and will clean the stale audiences and customers from paid media master dds and audience table. It cleans stale audiences from Facebook, Google and LinkedIn network. Stale Audiences: These are audiences which are deleted from Ad networks but are present in Paid Media	After you configure any paid destinations.

Job Scheduler Name	Description	When to enable
	Manager's master and audience table.	

Cleanup and data-mining agents

Agent Name	Multi-node	Description	Node type
ProcessPurging	No	Cleans up batch assets, such as classes and tables, that are created while running a program. A cleanup occurs when the following conditions have been met: • The ProcessProgramRun agent has completed the processing for all of its actions. The status of all actions must be Completed • The timeframe to purge a work object has lapsed. The timeframe is specified as a number of days. You can view this setting in the Pega Customer Decision Hub portal on the Application settings landing page, in the Delete campaign data older than field. For more information, see Application configuration. Additionally, the ProcessPurging agent configuration uses the *MKTPurgeMaxRecords* setting to determine the maximum number of records that may be purged during a cleanup. By default, the maximum number of purged records is set to 100. To change the default value, you first need to add this setting to the Dynamic System Settings list, and then set the new maximum number.	BackgroundProcessing

Startup and maintenance agent

Agent Name	Multi-node	Description	Node type
PegaMKTStartupAgent	Yes	Handles routines, maintenance, and checks that need to occur during the application server startup. The agent can be used for data instance re-sets or for whenever something must be done during node startup.	BackgroundProcessing
ResilenceHandler	No	Detects segmentation process crashes during runtime and recovers the long running and not recoverable processes to a healthy state. The processes can be re-run or edited again.	BackgroundProcessing

Deprecated agents

Agent Name	Multi-node	Description	Node type
AnalysisSchedule	Yes	Similar functionality to the SegmentSchedule agent, but instead of segments, it runs Analysis Project rules according to the schedule.	BackgroundProcessing
APNs Feedback Service	No	Uses the Apple APNs Feedback service to periodically retrieve uninstalled push tokens and calls Pega Customer Decision Hub's Push Notification Un-registration service to remove these uninstalled push tokens from the registered list.	BackgroundProcessing
ProcessCampaignRun	Yes	This agent is only available if Field Marketing is available as part of your Pega Customer Decision Hub application. Executes the actions for the selected list of customers in a Field Marketing campaign.	Batch
RefreshCampaignOfferStats	No	This agent is only available if Field Marketing is available as part of your Pega Customer Decision Hub application. Performs data mining by periodically refreshing a materialized view in the database to summarize the Interaction History data for Field Marketing reporting purposes.	BackgroundProcessing

Managing Decision Data rules in Pega Customer Decision Hub

Pega Customer Decision Hub uses Decision Data rules as the backing of several types of artifacts, such as treatments, Next-Best-Action taxonomy or Context Dictionary definitions, and others. You can view, manage, and define these Decision Data rules directly in the Pega Customer Decision Hub portal.

Pega Customer Decision Hub

1. To create a new property, in the Pega Customer Decision Hub portal, click **Next-Best-Action Designer Taxonomy Properties** .
2. Click **Add Add property** , and then configure the required property.
   For more information about configuring properties, see Properties.
3. To add the new property to the Decision Data rule form, in the Pega Customer Decision Hub portal, click **Intelligence Decision Data** .
4. Find and open the Decision Data rule that you want to edit. **For example:** To customize the form used by your email treatments, open the *EmailTreatments* rule.

5. In the Form fields section, review the list of properties which are available as fields on the treatment form.
    - To add a field, click the **Add field** button.
    - To remove a field, click the **Remove property** icon.

Using a Configuration Modal Window

The configuration modal window is utilized in multiple places within Pega Customer Decision Hub. This appendix outlines the components of a configuration window and how to use the window to select items.

Pega Customer Decision Hub

Configuration Window Components

The configuration modal window has the following components:

1. **Items list** - The left half of the window has a list of selectable items. This component has the following usage patterns:
    - The modal presents the twenty most recently updated rules (e.g. Strategies) in the system.
    - If there are more than twenty results, use the **View 20 more results** link at the bottom to load the next set of results.
    - Use the search box to search for rules by name or description.
    - The refresh icon refreshes the list of results, retaining any search criteria that was previously entered.
    - Use the **Create** link (if available in the window) to create a new rule. The system pre-populates relevant fields on the rule's create form. After creating the rule, refresh the list of results in the modal to see your new rule.
    - Click an entry in the list to select it. Upon selection, the system displays information about the selected entry in the details panel on the right.
2. **Item details panel** - The upper portion of the right half of the window lists pertinent details about the currently selected item. Use the link in this panel (if present) to open the corresponding rule.
3. **Selected items panel** - The lower portion of the right half of the configuration window lists the items that have been selected so far. Clicking an item link in this section opens the corresponding rule. This panel also displays an informational message (either Only 1 can be added or Multiple can be added) to indicate how many elements can be selected using this configuration window.

Selecting Items

Based on the element being configured and its intended usage, users can either select a single item from the list or multiple items.

Click **Add** to include the corresponding item as a selection for the configuration. In the case of a single-select configuration modal, the **Add** button is disabled on all other rows once an item has been selected. To select a different item, first remove the currently added item by either clicking **Remove** in the item's row or by clicking the corresponding trashcan icon in the selected items panel.

After adding the desired items, click **Apply** to confirm your selection changes and close the configuration window. Alternately, click **Cancel** at any time to discard selection changes and close the configuration window.

Note: When a configuration window is triggered from an already open configuration modal window, the new window slides in. Upon application of selected values or cancellation, the secondary window slides out and returns the user back to the original configuration window.

Application configuration

This appendix outlines the various configuration options that are available for the administrator to configure the application. These settings should be configured as part of setting up the application and can be adjusted later, as the need arises.

Pega Customer Decision Hub

The Application landing page is available only to administrators with access to App Studio, and lists various configuration options. This landing page is accessible in App Studio via **Settings Application** .

To change any of these settings, click the **Edit** button on the particular setting's landing page. In edit mode, click **Save** to save and apply your changes to the settings; or click **Cancel** to discard your changes.

Note: These settings are stored in a node-level data page. Any changes made to these settings on a particular node should be immediately visible to all processes running on that node. In a multi-node environment, changes may take up to an hour to reflect across other nodes. In such situations, it is advisable to manually flush the data page on each of the nodes after changing these settings. To do so, open the Declare_MKTSystemSetting data page (under PegaMKT-System-Setting) and click the **Clear Data Page** button on the Load Management tab.

Configuration Settings

This landing page contains settings pertaining to the overall execution infrastructure.

Outbound Default Data Source	Displays the database instance where the customer table resides and where the application will store execution data (such as data pertaining to Segment and Campaign execution).
Outbound DB Preview File Size	Displays the number of records displayed in the preview file when previewing the contents of the data written for a Database Template.
Batch Size	Displays the size (number of records) into which back-end processing is batched, in order to leverage performance improvements from vertical and horizontal scalability.
Delete Campaign Run data older than X days	Displays the threshold (in number of days) for purging Campaign execution data. This only applies to data for Campaign Runs where all actions are complete.
Output File Date Format	Displays the default format to be applied to date fields in a File Template.
Global Exclusion List Segment	Displays the Segment that represents the global exclusion list. Customers in this list will be excluded from outbound communication. When no value is specified, the exclusion list feature is disabled.**Note:** Whenever a change is made to this setting, all event-based Campaigns need to be suspended and re-submitted for execution.
Use Simulation Data	Specifies whether Dashboard widgets should use simulated data. Enable this setting to review the Dashboard and its capabilities in the absence of real data.
Use Decision Data Source	Specifies whether a decision data store (such as Cassandra) should be utilized to retrieve any customer-centric data which is not required for segmentation. Using decision data sources requires the configuration of a DNode in your system. Review the Pega 7 DNode Cluster Management landing page help topic for details.

Artifacts

This landing page contains settings pertaining to the behavior and display of artifacts.

Campaigns	
Disable campaign action validation	Specifies whether Campaigns should validate the existence of Action rules utilized by the Campaign Strategy .

Require key codes	Specifies whether key code entry is required for artifacts (such as campaigns, actions, and treatments). If this setting is enabled, it will over-ride the user preference for specifying key codes.
Number of runs to schedule at a time for open-ended campaigns	Displays the number of Campaign Runs that the system will schedule at a time for open-ended Campaigns (i.e. those Campaigns where an end date is not set).
Campaign contacts edit mode	Determines the mode to use when managing contacts in Field Campaigns. This setting is only visible if the application includes Pega Field Marketing.
Email responses	
Default Response Page	Displays the Rule-HTML-Section rule that will be used as the default response page for Email responses.
Default Action Expired Page	Displays the Rule-HTML-Section rule that will be used as the default page for Email responses to expired actions.
Default Action Not Available Page	Displays the Rule-HTML-Section rule that will be used as the default page for Email responses to actions that are no longer available.
Geofences	
Mapping License Key	Displays the license key to use for the Google Maps mapping service.
Default Customer Location Properties	Displays the properties on the customer class to be used as the defaults for the customer's location (latitude and longitude) when filtering on Geofences in Strategies.

Pega Community Wiki

Learn more out-of-the-box capabilities and ways to address common design patterns of Pega Platform and Applications such as Pega Customer Decision Hub on the Pega Community Wiki.

Pega Customer Decision Hub

You can access the website using the link Pega Community Wiki.