



Guidewire PolicyCenter™

Advanced Product Designer Guide

Release 2021.04

© 2021 Guidewire Software, Inc.

For information about Guidewire trademarks, visit <http://guidewire.com/legal-notice>.

Guidewire Proprietary & Confidential — DO NOT DISTRIBUTE

Product Name: Guidewire PolicyCenter

Product Release: 2021.04

Document Name: Advanced Product Designer Guide

Document Revision: 14-June-2021

Contents

Guidewire Documentation	7
About PolicyCenter documentation	7
Support	9
1 About Advanced Product Designer	11
Project team.	11
Creating insurance products.	12
Conceptualize	12
Visualize	13
Finalize	13
Continuing product design.	14
Advanced Product Designer versus Product Designer	15
2 System settings for Advanced Product Designer.	17
Define Gosu package for product line files	17
3 Conceptualizing products in mind maps	19
Mind map structure.	19
Labels on product and product line	20
Labels for Short Name (*PD).	21
Markers in mind map	22
Guidewire markers for XMind	23
Legacy support for XMind markers	24
Language support in mind maps	25
Display languages in mind maps	25
Japanese mind maps.	26
Working with XMind	26
Installing XMind	26
Add Guidewire markers to XMind.	26
Create a product mind map	27
Display labels in XMind.	27
Risk objects in mind map	27
Modeling locations on risk objects.	28
Exposures in mind map	29
Clauses in mind map.	29
Events on clauses in mind map	30
Covered costs on clauses in mind map	30
Defining option clause terms in mind map	31
Defining package clause terms in mind map	31
Defining subclauses in mind map	32
Schedules in mind map	33
Attributes in mind map	34
Drop-down lists in mind map	35
Nested drop-down lists with dependent relationships	35
Involved party in mind map	36
Usage and availability notes in mind map	37
Extract usage and availability notes from mind map.	37
Pricing in mind map	38
Generate mind map from product	39

4 Visualizing products in PolicyCenter	41
Product design modes in Advanced Product Designer	41
Editing visualized products	42
Create visualized products	42
Create visualized product from mind map	42
Create visualized product directly in PolicyCenter	43
Create visualized product from template	43
Import mind maps for additional languages	43
Start policy submission for visualized product	44
Remove visualized product	44
Update visualized product	44
Export visualized product to template	45
5 Working with visualized products	47
Products and product lines	47
Product fields on Product Definition screen	48
Product line Details tab	49
Fields tab	50
Field naming conventions	51
Risk objects in visualized product	51
Location risk object in visualized product	51
Create risk object types in visualized product	52
Risk Object Definition popup	53
Splitting risk objects in visualized products	54
Split risk object by rating period in Product Definition	54
Split risk objects in a policy transaction	55
Clauses in visualized product	55
Manage Clause Categories popup	56
Coverages, Exclusions, and Conditions tabs	56
Clause terms in visualized product	57
Option clause terms in visualized product	57
Package clause terms in visualized product	59
Events on coverages	60
Covered costs in visualized product	61
Subclauses in visualized product	61
Create subclause in visualized product	62
Subclauses in a visualized policy transaction	63
Schedules on clauses	63
View or add schedule in visualized product	63
Schedule term configuration	64
Attributes	64
Drop-down lists in visualized product	65
Drop-down lists in developer mode	66
Modeling dependent relationships in drop-down lists	68
Involved party in visualized product	69
Product model properties	70
Tags	70
Usage and availability in visualized product	70
Specifying additional availability in Product Designer	72
Default, minimum, and maximum values for product model properties	73
Subclauses with numeric and option clause terms	74
Add default and minimum and maximum values	74
Add minimum or maximum value to numeric or option clause term	75
Add field usage	76
Exposures in visualized product	77

Splitting exposures in visualized products.	78
Exposure Fields popup	78
Define exposure as split by rating period.	78
Split exposures in a policy transaction.	79
Editions for product lines.	79
Editions tab.	81
Edition property popup.	81
Create an edition	81
Change product model properties in an edition	82
Delete an edition	82
Pricing for rating in visualized product	83
Add pricing to a risk object.	84
Pricing screen in submission in User mode	85
Adding costs in Pricing	85
Itemized for Billing	87
Separate Collection.	90
Basis attribute	93
Cost codes	97
Modify cost code definitions	98
Multi-line products	98
Add product lines to a multiline product.	98
Remove or delete product line from multi-line product.	99
6 Finalizing and generating the product.	101
Metadata	102
Adding Short Name (*PD)	103
Add metadata	104
Review product content.	105
Validated fields	105
Validate product in Designer mode	106
Validate product in Developer mode	106
Move product.	107
Generate and deploy the product	107
Generate Product Code	108
7 After generating product code.	109
Loading editions into generated product	109
Editions high-level workflow	110
Load and activate editions in generated product	111
Activating editions in development or test mode	112
Changing visualized product after code generation	113
Regenerating the product.	113
Merging changes in generated product	114
Generated product files	114
System files	115
Resources.	115
Product model.	116
Policy line patterns	117
Risk object entities	119
Clause entities.	120
Schedule entities	120
Display names.	121
Typelists.	121
PCF files	121
Gosu classes	122
Displaying Risk Details in generated product.	123

Risk objects and coverables	123
8 Upgrade generated APD products	125
9 Benefits of APD for externally-managed products	127
View a visualized version of your product	127
Generate EDGE APIs for externally-managed products.	128
Migrate externally-managed product to Advanced Product Designer	128
Extract APD representation of externally managed product	129
10 Example mind maps	131
11 Advanced Product Designer glossary	133
attribute	133
basis-scalable	133
category	133
class code	133
clause	133
clause term	134
clause term option	134
condition	134
coverable	134
coverage	134
coverage category	134
exclusion	134
exposure	134
form	134
generated product	134
mind map	135
modifier	135
offering	135
package	135
policy line	135
product	135
product definition	135
product line	135
product model properties	136
question set	136
rate-scalable	136
reference date	136
risk object	136
schedule	136
surcharge	136
template	137
visualized product	137

Guidewire Documentation

About PolicyCenter documentation

The following table lists the documents in PolicyCenter documentation:

Document	Purpose
<i>InsuranceSuite Guide</i>	If you are new to Guidewire InsuranceSuite applications, read the <i>InsuranceSuite Guide</i> for information on the architecture of Guidewire InsuranceSuite and application integrations. The intended readers are everyone who works with Guidewire applications.
<i>Application Guide</i>	If you are new to PolicyCenter or want to understand a feature, read the <i>Application Guide</i> . This guide describes features from a business perspective and provides links to other books as needed. The intended readers are everyone who works with PolicyCenter.
<i>Upgrade Guide</i>	Describes the overall upgrade process, and describes how to upgrade your configuration and database. The intended readers are implementation engineers who must merge base application changes into existing application extensions and integrations.
<i>Configuration Upgrade Tools Guide</i>	Describes the tools and functionality provided by the Guidewire InsuranceSuite Configuration Upgrade Tools. The intended readers are implementation engineers who must merge base application changes into existing application extensions and integrations. Visit the Guidewire Community to access the <i>Configuration Upgrade Tools Guide</i> , which is available for download, separately from the main documentation set, with the Configuration Upgrade Tools.
<i>Installation Guide</i>	Describes how to install PolicyCenter. The intended readers are everyone who installs the application for development or for production.
<i>System Administration Guide</i>	Describes how to manage a PolicyCenter system. The intended readers are system administrators responsible for managing security, backups, logging, importing user data, or application monitoring.
<i>Configuration Guide</i>	The primary reference for configuring initial implementation, data model extensions, and user interface (PCF) files for PolicyCenter. The intended readers are all IT staff and configuration engineers.
<i>PCF Format Reference</i>	Describes PolicyCenter PCF widgets and attributes. The intended readers are configuration engineers. See the <i>Configuration Guide</i> .
<i>Data Dictionary</i>	Describes the PolicyCenter data model, including configuration extensions. The dictionary can be generated at any time to reflect the current PolicyCenter configuration. The intended readers are configuration engineers.
<i>Security Dictionary</i>	Describes all security permissions, roles, and the relationships among them. The dictionary can be generated at any time to reflect the current PolicyCenter configuration. The intended readers are configuration engineers.
<i>Globalization Guide</i>	Describes how to configure PolicyCenter for a global environment. Covers globalization topics such as global regions, languages, date and number formats, names, currencies, addresses, and phone numbers. The intended readers are configuration engineers who localize PolicyCenter.
<i>Rules Guide</i>	Describes business rule methodology and the rule sets in Guidewire Studio for PolicyCenter. The intended readers are business analysts who define business processes, as well as programmers who write business rules in Gosu.
<i>Guidewire Contact Management Guide</i>	Describes how to configure Guidewire InsuranceSuite applications to integrate with ContactManager and how to manage client and vendor contacts in a single system of record. The intended readers are PolicyCenter implementation engineers and ContactManager administrators.
<i>Best Practices Guide</i>	A reference of recommended design patterns for data model extensions, user interface, business rules, and Gosu programming. The intended readers are configuration engineers.
<i>Integration Guide</i>	Describes the integration architecture, concepts, and procedures for integrating PolicyCenter with external systems and extending application behavior with custom programming code. The intended

Document	Purpose
	readers are system architects and the integration programmers who write web services code or plugin code in Gosu or Java.
<i>REST API Client Guide</i>	Describes how to use the InsuranceSuite REST API Client to make outbound HTTP calls to internal or third-party REST services.
<i>Java API Reference</i>	Javadoc-style reference of PolicyCenter Java plugin interfaces, entity fields, and other utility classes. The intended readers are system architects and integration programmers.
<i>Gosu Reference Guide</i>	Describes the Gosu programming language. The intended readers are anyone who uses the Gosu language, including for rules and PCF configuration.
<i>Gosu API Reference</i>	Javadoc-style reference of PolicyCenter Gosu classes and properties. The reference can be generated at any time to reflect the current PolicyCenter configuration. The intended readers are configuration engineers, system architects, and integration programmers.
<i>ISBTF and GUnit Testing Guide</i>	Describes the tools and functionality provided by InsuranceSuite for testing application behavior during an initial implementation or an upgrade. The guide covers functionality related to Behavior Testing Framework, GUnit, and Gosu functionality designed specifically for application testing. There are two sets of intended readers: business analysts who will assist in writing tests that describe the desired application behavior; and technical developers who will write implementation code that executes the tests.
<i>Glossary</i>	Defines industry terminology and technical terms in Guidewire documentation. The intended readers are everyone who works with Guidewire applications.
<i>Advanced Product Designer Guide</i>	Advanced Product Designer is a tool that helps you, a business user, design, simulate, and deploy an insurance product. The intended readers are business analysts who understand insurance products, business systems analysts who liaise between business analysts and IT, project managers, and IT who provides technical expertise in areas such as programming, testing, and databases.
<i>Product Model Guide</i>	Describes the PolicyCenter product model. The intended readers are business analysts and implementation engineers who use PolicyCenter or Product Designer. To customize the product model, see the <i>Product Designer Guide</i> .
<i>Product Designer Guide</i>	Describes how to use Product Designer to configure lines of business. The intended readers are business analysts and implementation engineers who customize the product model and design new lines of business.
<i>REST API Framework</i>	Describes the Guidewire InsuranceSuite framework that provides the means to define, implement, and publish REST API contracts. It also describes how the Guidewire REST framework interacts with JSON and Swagger objects. The intended readers are system architects and integration programmers who write web services code or plugin code in Gosu or Java.

Conventions in this document

Text style	Meaning	Examples
<i>italic</i>	Indicates a term that is being defined, added emphasis, and book titles. In monospace text, italics indicate a variable to be replaced.	A <i>destination</i> sends messages to an external system. Navigate to the PolicyCenter installation directory by running the following command: <pre>cd installDir</pre>
bold	Highlights important sections of code in examples.	<pre>for (i=0, i<someArray.length(), i++) { newArray[i] = someArray[i].getName() }</pre>
narrow bold	The name of a user interface element, such as a button name, a menu item name, or a tab name.	Click Submit .

Text style	Meaning	Examples
monospace	Code examples, computer output, class and method names, URLs, parameter names, string literals, and other objects that might appear in programming code.	The getName method of the IDoStuff API returns the name of the object.
<i>monospace italic</i>	Variable placeholder text within code examples, command examples, file paths, and URLs.	Run the startServer <i>server_name</i> command. Navigate to <code>http://server_name/index.html</code> .

Support

For assistance, visit the Guidewire Community.

Guidewire customers

<https://community.guidewire.com>

Guidewire partners

<https://partner.guidewire.com>

About Advanced Product Designer

Advanced Product Designer is a business tool that helps you design, simulate and deploy an insurance product.

With Advanced Product Designer, a Business System Analyst can:

- Define an insurance product, including coverages, exclusions, conditions, risk details, and other attributes
- View the product in PolicyCenter and create a policy submission

You, a Business System Analyst, can prototype insurance products in mind maps created in XMind. You import the mind map into PolicyCenter, creating a visualized product. In the visualized product, you can create submissions and other policy transactions and navigate through the product screens.

Note: XMind is mind mapping and brainstorming software developed by XMind Ltd.

The project team works iteratively on the product definition, making changes to the mind map, and importing those changes to the visualized product. In PolicyCenter, you modify and enhance the visualized product. When the visualized product is complete, you can generate the product. The generated product includes entities, and product model and user interface elements that can be further customized using Product Designer and Guidewire Studio.

Project team

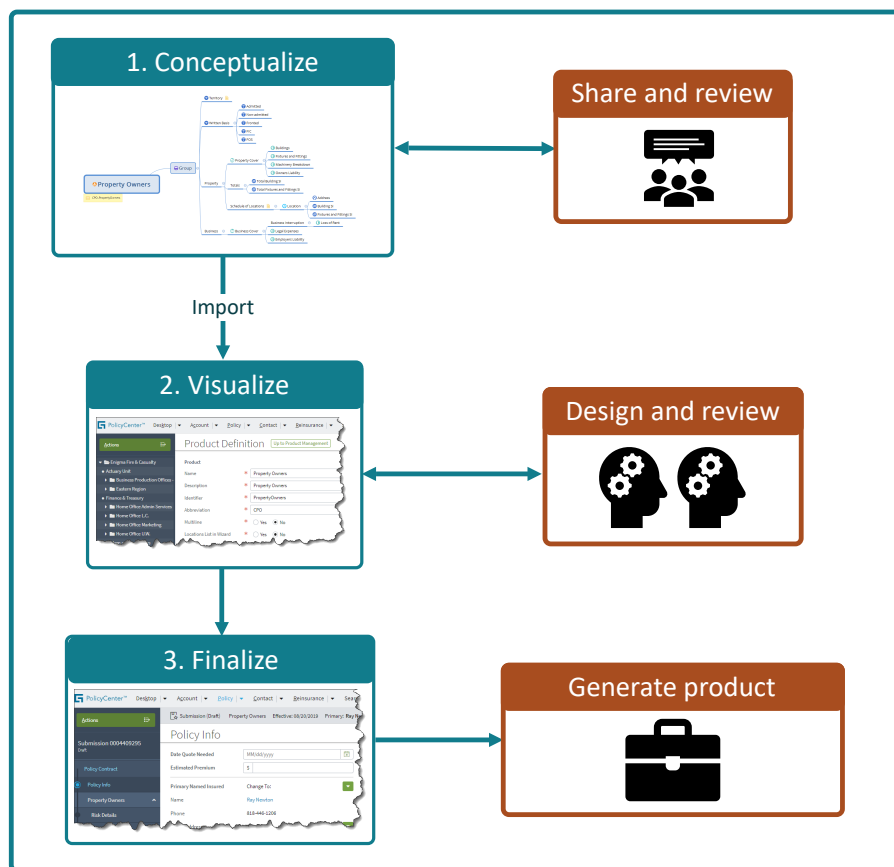
The project team developing the insurance product needs people with skills in the following roles:

Role	Description
Business Analyst	Business Analysts are insurance industry experts who understands the business needs. Business Analysts are given general requests and transform them into specific business requirements.
Business Systems Analyst	Business Systems Analysts are the liaison between Business Analysts and IT, the technical development team. Business Systems Analysts work with Business Analysts to conduct needs analysis and write business directives for IT. Business Systems Analysts take business directives and create precise, actionable requirements, specifying fields, content, behaviors and business rules. Business Systems Analysts work with IT to test and validate requirements after implementation.

Role	Description
Product Manager	Product Managers define product offerings, determine underwriting criteria, and ensure compliance with jurisdictional regulations.
IT	Information Technology provides technical expertise in areas such as programming, databases, system administration, and testing.

Creating insurance products

The steps for creating a product using Advanced Product Designer are:



Developing a product is an iterative process. You can prototype a product through the conceptualize, visualize and finalize steps and generate the product. You continue development by adding product details in each step. Even after the product is in production, you can continue to modify the product details in each step as business requirements change.

Conceptualize

Business System Analysts, Business Analysts, and Product Managers are involved in conceptualization. Users of the product, such as underwriters and actuaries, may provide input.

You *conceptualize* the product in XMind by creating a mind map that captures the product requirements. The product structure is mapped into a tree-like hierarchy, with the product as the root of the tree.

The nodes of the tree model the different parts of the product, such as risk objects, coverages, exclusions, and conditions. The team works iteratively on the mind map, defining the product requirements until achieving a level of stability.

If you prefer not to use XMind, then you can create the product hierarchy directly in Advanced Product Designer screens in PolicyCenter.

Share and review

During conceptualization, you share and review the product mind map with team members and incorporate their feedback.

Import

After incorporating feedback, you are ready to import the mind map into PolicyCenter.

Visualize

Business System Analysts are involved in visualization. IT is involved when the product nears finalization. Business Systems Analysts may consult with Product Managers and Business Analysts. Users of the product, such as underwriters and actuaries, may provide input.

When the mind map is imported into PolicyCenter, it becomes a *visualized product*. If you created the product directly in Advanced Product Designer screens in PolicyCenter, then you already have your visualized product.

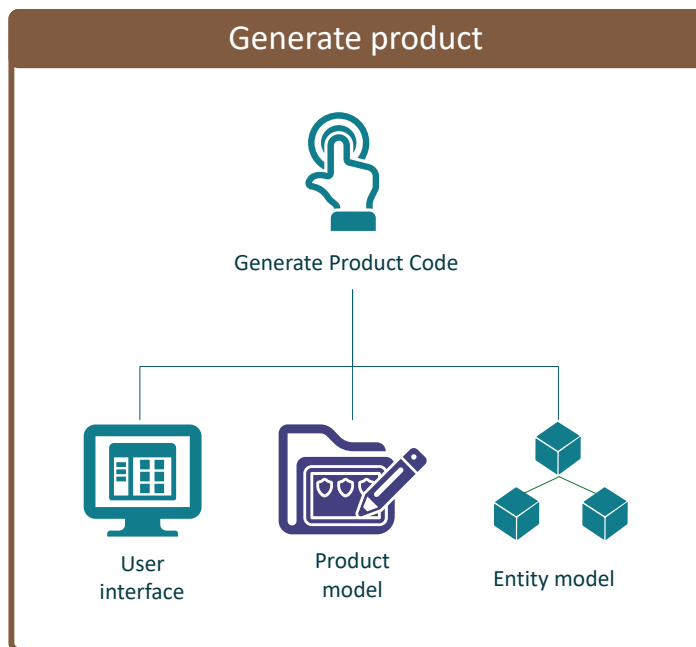
In a visualized product, you can navigate through a submission or other policy transaction just like you do with products installed in PolicyCenter. Visualization enables you to work in PolicyCenter with your new product without needing to develop a prototype. In a policy transaction, you can navigate through the screens, adding risk objects, selecting coverages, and entering policy data.

In the visualized product, you can continue to add product details on Advanced Product Designer screens in PolicyCenter. You can also make changes to the mind map, and reimport those changes.

Finalize

IT and Business System Analysts are involved in finalization. Product Managers may be consulted.

When you have refined and completed entering product details in the visualized product, you are ready to *finalize* the product. In finalization, Advanced Product Designer generates the product from the visualized product, creating the artifacts needed for PolicyCenter including entities, product model, and user interface elements.



Continuing product design

As you continue to define and design the product, you continue to make changes to the mind map, visualized product, and generated product.

In the mind map, you make changes, such as corrections or additions, and update the visualized product. If you change the name of a topic, this becomes an addition when you update the visualized product. If you delete a topic from the mind map, it is not deleted when you update the visualized product. You must delete them manually from the visualized product.

Alternatively, you can make changes directly in the visualized product in PolicyCenter. If you wish to keep the mind map updated, you can manually update the mind map to reflect these changes. You can also generate a mind map from a product in PolicyCenter.

Some product details must be entered directly in the visualized product. The details include parts of pricing, minimum and maximum values, usage properties, and default values. You can enter these details in the mind map. However, these are not brought in to the visualized product when you import the mind map.

In the generated product, you can enhance the product using the development tools provided by PolicyCenter, such as Guidewire Studio. You can create underwriting rules, rating, and forms management. You can integrate with other applications, and make other changes to meet your needs. You incorporate changes made to the visualized product by regenerating it and merging changes.

Continue this iterative process until you have captured the product requirements.

When possible, make changes in Advanced Product Designer

When working with an APD managed product, make changes in Advanced Product Designer on the **Product Management** screens if possible. For example, always add new structural elements, such as clauses, risk objects, or attributes, in Advanced Product Designer rather than adding them in Guidewire Studio or Product Designer. If the mind map is your source of truth, add these elements in the mind map, and update the product with the mind map.

If you add structural elements in Guidewire Studio or Product Designer, you cannot alter them in the visualized product in Advanced Product Designer.

For example, after generating the product, you find that you need an additional Building attribute. In Advanced Product Designer, add that attribute in the visualized product on the **Product Management** screens, and regenerate the

product. You can make this change even if the product is already in production. Do not add the attribute in Product Designer. Unlike attributes defined in the visualized product, it will not be accessible to view or modify in the visualized product on the **Product Management** screens.

Documenting the product definition

You may wish to keep the mind map as your documentation of the product definition, and update the map with changes you make in the visualized product. Alternately, you can document the product definition elsewhere. Ultimately, the generated product is the implementation of the product definition.

Advanced Product Designer versus Product Designer

You can start developing your product with Advanced Product Designer, and finalize your work on the product model using Product Designer.

Product Designer provides greater access to the product model. Product Designer provides additional capabilities such as the ability to:

- Use a script to determine whether a clause can be added to the policy
- Perform an action when a clause is added or removed

While Advanced Product Designer lets you model availability for product model properties, Product Designer provides additional availability features, such as:

Availability scripts

Gosu scripts that can capture complex logic.

Grandfathering

Continue to offer a coverage or other pattern to existing customers, even if the pattern is not otherwise available.

Offerings

Enable or disable product model patterns for different sales or customer situations.

Availability lookup tables

Specify whether or not a given pattern is available based on date, jurisdiction, industry code, transaction type, underwriting company, among others.

Note: When possible, use the features provided in Advanced Product Designer. Only use Product Designer for requirements that Advanced Product Designer does not provide. For example, specify minimum and maximum values and availability in Advanced Product Designer when possible.

See also

- *Product Designer Guide*
- “Specifying additional availability in Product Designer” on page 72

System settings for Advanced Product Designer

Before you begin developing products, set up your development system for Advanced Product Designer.

Server mode

Set the server to development mode. You can use the **Product Management** screen to create products when the server is in development or test mode.

See also

- *System Administration Guide*

Define Gosu package for product line files

Before you start generating products, define the Gosu package that will contain the Gosu classes for your product lines.

About this task

The `APDProductLinesGosuPackage` script parameter defines the package in which PolicyCenter generates Gosu files for your product lines. The package is created in `gsrc`, and the default value is `ext.lob`. You can change the package name to a name appropriate for your implementation, such as `xyzcorp` or `ext.xyzcorp`.

See also

- “Gosu classes” on page 122

Procedure

1. Access the `APDProductLinesGosuPackage` parameter in one of the following ways:
 - If PolicyCenter is not running and the database is empty, edit `ScriptParameters.xml` in Guidewire Studio.
 - Otherwise, go to **Administration**→**Utilities**→**Script Parameters** in PolicyCenter.
2. If you like, change the value of `APDProductLinesGosuPackage` from `ext.lob` to a name appropriate for you implementation, such as `xyzcorp`.

Gosu classes for your product line will be created in the `xyzcorp` package. In Studio, you can access these files in `configuration/gsrc/xyzcorp`.

- 3.** If you have previously generated product lines, move the Gosu classes for those product lines from the old to the new package.
 - a.** Stop PolicyCenter if it is running.
 - b.** In Guidewire Studio or in the file system move the Gosu classes from the old to the new package.
 - c.** Restart PolicyCenter.

Conceptualizing products in mind maps

You can conceptualize the product or product line in XMind by creating a mind map that captures your requirements. In the map, the emphasis is on creating a logical structure that provides the conceptual model of the product in insurance business terms. Product implementation is not the focus at this stage.

Advanced Product Designer in PolicyCenter can read mind maps created in XMind. In XMind, you map the product structure into a tree-like diagram. The root of the tree is either a product or a product line.

The nodes of the tree model different parts of the product or product line. The project team works iteratively on the mind map, defining and fine-tuning the product requirements until achieving agreement on the model and achieving product stability.

Continuing product design

As you continue to define and design the product, you can continue to make changes to the mind map.

In the mind map, you make changes, such as corrections or additions, and update the visualized product. If you change the name of a topic, this becomes an addition when you update the visualized product. If you delete a topic from the mind map, it is not deleted when you update the visualized product. You must delete them manually from the visualized product.

Mind map structure

When importing a mind map, Advanced Product Designer reads the hierarchical structure, annotations, and markers to interpret the mind map. Only elements with specific marker are incorporated into the product visualization in PolicyCenter.

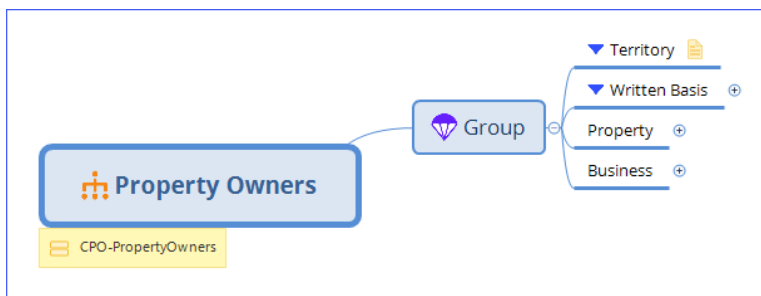
You can add any amount of detail in the mind map with notes, callouts, hyperlinks, among others. For example, XMind has Notes that you can attach to elements. You can use notes to describe business properties, pricing properties, and other details. The details that Advanced Product Designer does not support are dropped when importing to PolicyCenter. However, you can keep these details in the map if they are useful for product conceptualization.

The mind map must meet the following criteria:

- A mind map can contain only one product. A product is not required. On import, a default product is added if the mind map does not contain a product.
- Each mind map must include at least one product line, and product lines must be subordinate to the main product..
- Each product or product line must have an XMind label.
- Product lines contain the risk objects, risk objects may include attributes, exposures, clause categories, and more risk objects. Clause categories may have coverages, exclusions, and conditions. Attributes and terms may be added to these objects.
- A clause is always subordinate to a clause group.
- A clause can only apply to one risk object.
- A risk object that is a scheduled item definition can only be linked to attribute nodes like text, number, decimal, true/false, among others.

The mind map must adhere to this structure. During import, Advanced Product Designer provides validation checks and only critical errors stop the import, such as no product or line definition. All other invalid structures are ignored in the same way as unsupported annotations are ignored. If this occurs, you may not get everything imported that you expected. You can fix these problems in the mind map and import to update the product.

The following illustration shows a mind map in XMind for a property owners line. Property Owners line has a Group risk object. The children of Group are Territory drop-down list, Written Basis drop-down list, Property attribute, and Business attribute.



Labels on product and product line

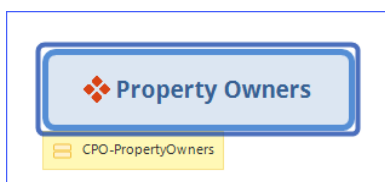
To import the mind map into PolicyCenter, Advanced Product Designer requires that products and product lines have a **Label** in the map.

To add a label, right-click on the topic, and select **Insert→Label**.

Product label

The product label is of the format *<Abbreviation>-<Identifier>*. The *Abbreviation* is three characters, the first of which must be an uppercase letter and the remaining letters, digits, \$, or _. The *Identifier* must start with an uppercase letter followed by letters, digits, \$, or _.

For example, the following **Property Owners** product has the label **CPO-PropertyOwners**:



Line label

The line label is of the format <LinePrefix>-<Identifier> where *LinePrefix* is three characters, the first of which is an uppercase letter followed by letters, digits, \$, or _. The *Identifier* starts with an uppercase letter followed by letters, digits, \$, or _.

Visualized product in PolicyCenter

When Advanced Product Designer creates the visualized product in PolicyCenter:

- The 3-digit code becomes an **Abbreviation**. In a line template, the code becomes the **Line Prefix**.
- The identifier becomes the **Identifier** for the line appended with the word **Line**.

Labels for Short Name (*PD)

If **Product Design Mode** is **Developer**, the **Product Definition** screens have **Short Name (*PD)** fields. These fields must be filled in before generating the product. For drop-down lists, you specify the short name in the **Typelist (*PD)** field.

When designing a product, create short names that follow best-practice naming conventions agreed upon by the organization or team.

Mind map label for short name

For all topics with markers (except Product  and Product Line ) in the XMind mind map, you can specify the short name in a label. The label becomes the **Short Name (*PD)** in the visualized product.

Short name requirements

Short names must:

- Start with an uppercase letter
- Include only letters, digits, \$ or _
- Clauses and clause categories – Be unique within the product line
- Clause terms – Be unique within the parent clause
- Drop-down list entries – Can start with a lowercase letter or digit

Short Name (*PD) fields have character limits that are enforced in the user interface. If you import a mind map or template where short names exceed the character limits, the short names are truncated. **Short Name (*PD)** is visible if **Product Design Mode** is **Developer**.

The character limits for short names are:

Fields	Limit	Description
Clauses	45	
Clause categories	45	
Clause terms	30	
Risk objects	22	
Exposures	22	
Attributes or fields	30	
Drop-down list values	45	
Product line	22	On the Product Definition screen, PolicyLine Subtype , excluding prefix (*PD)

Words to avoid

The short name cannot be one of the following keywords. The check is case-insensitive.

- Java keywords
- Gosu keywords – See *Gosu Reference Guide*
- The following SQL keywords:
 - use
 - type

The name can include these words. For example, `abstract` is a Java and Gosu keyword, so the short name cannot be `ABSTRACT`, but can be `abstractPhrase`.

Words to avoid for exposures and coverables

For exposures and coverables, field short names cannot match certain property names on PolicyCenter entities. Coverables are risk object that have coverages, including product lines. Using one of these words results in a warning, but does not block you from completing code generation. Guidewire recommends that you change the name so that no warning is issued.

On exposures and coverables, including the product line, do not use these words as field short names:

- | | |
|-----------------|------------------|
| • Changed | • ExpirationDate |
| • Class | • New |
| • DisplayName | • TYPE |
| • EffectiveDate | |

On coverables, including the product line, do not use this word as field short names:

- State

On the product line, do not use these words as field short names:

- AdditionalInsureds
- BaseState
- Branch

Note: These are the words to avoid that you are most likely to use. This is not a complete list.

Defaulting Short Name (*PD)

When you create or update a visualized product from a mind map, you can choose to create default short names. Choose **Default additional fields** on the **Load Product** popup. **Design Mode** must be set to **Developer**.

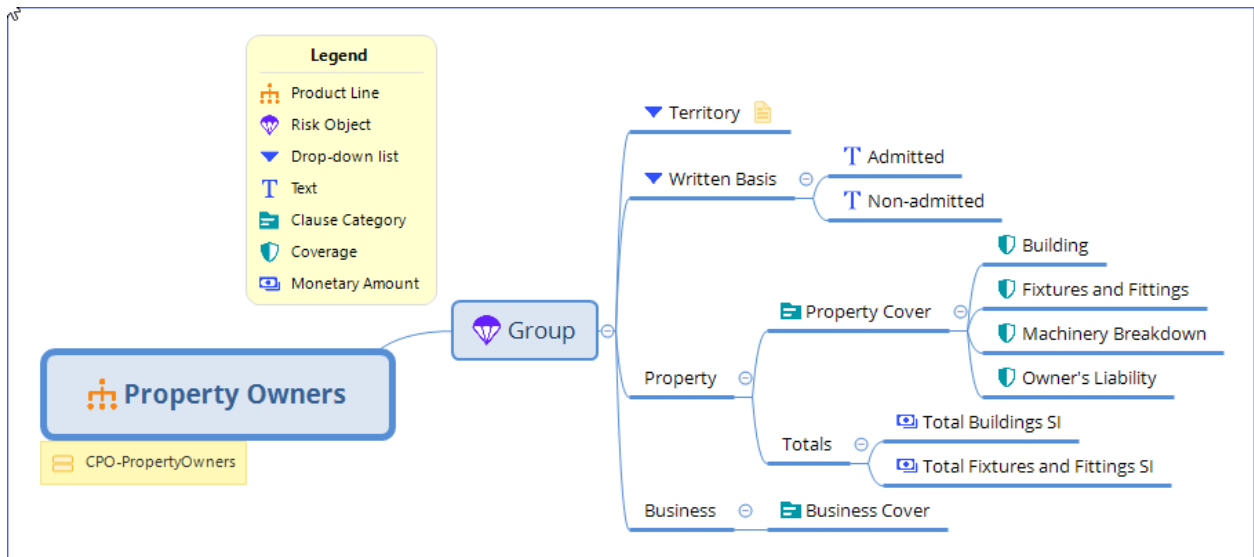
If **Product Design Mode** is **Developer**, the **Product Definition** screens have **Short Name (*PD)** fields. These fields must be filled in before generating the product. For drop-down lists, you specify the short name in the **Typelist (*PD)** field.

When designing a product, create short names that follow best-practice naming conventions agreed upon by the organization or team.

Markers in mind map

XMind provides **Markers** to describe and classify the topics your mind map. Guidewire provides XMind markers to tag the elements of your map that are part of the product model.

The following screen shot from XMind shows a product line with Guidewire markers identifying the product line, risk object, coverages, attributes, and other parts of the product. These markers provide information that Advanced Product Designer uses when importing the map.



See also

- “Add Guidewire markers to XMind” on page 26

Other XMind markers

Besides the Guidewire markers and the legacy markers, you can use other XMind markers for any purpose, such as flagging items that need further review. Excepting the Guidewire and legacy markers, these are ignored when importing into PolicyCenter.

Do not add both Guidewire marker and legacy marker to a topic

To avoid unexpected results when importing the mind map, do not add both a Guidewire marker and a legacy marker to a topic. If the markers do not represent the same type, the import will use either the Guidewire marker or legacy marker.

Guidewire markers for XMind

Marker type	Marker	Represents
Product model		Product
		Product Line
		Risk Object
		Exposure
		Location Risk Object
		Risk Attribute Category
		Event
		Covered Cost
		Clause Category

Marker type	Marker	Represents
		Coverage
		Condition
		Exclusion
Data types		Text
		Number
		Decimal
		True/false
		Date-time
		Drop-down list
		Monetary Amount
		Involved Party
		Location
Other		Pricing

Legacy support for XMind markers

Use of the legacy markers is deprecated. Do not use the legacy markers in mind maps that use the Guidewire markers. You can import mind maps that use legacy markers created for a prior implementation, the *Advanced Product Delivery Accelerator*.

Marker type	Marker	Represents
Product model	 Red star	Product
	 Orange star	Product Line
	 Blue star	Risk Object
	 Green star	Exposure
	 Question symbol	Risk Attribute Category
	 Exclamation point	Event
	 Plus sign	Covered Cost
	 Purple star	Clause Category
	 Smile face	Coverage
	 Laugh face	Condition

Marker type	Marker	Represents
Data types	 Angry face	Exclusion
	 Quote symbol	Text
	 Bar Chart symbol	Number
	 Line Graph symbol	Decimal
	 Right symbol	True/false
	 Thermometer symbol	Date-time
	 Pie Chart symbol	Drop-down list
	 Money symbol	Monetary Amount
	 Contact symbol	Involved Party
	 Flight symbol	Location
	 Medal symbol	Pricing

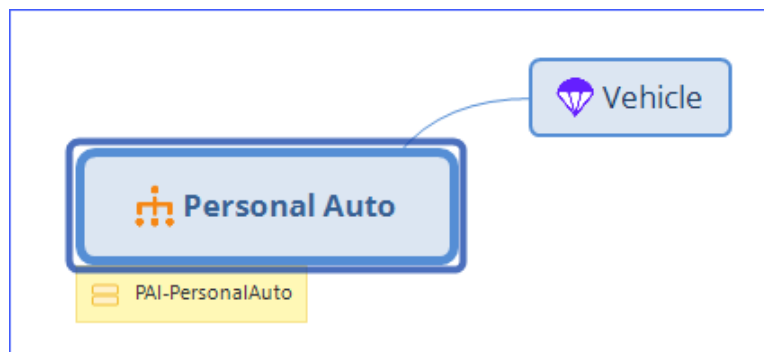
Location risk object

To mark a topic as a location risk object, mark it as a risk object  and a location .

Language support in mind maps

Display languages in mind maps

In the mind map, the topic text, *Personal Auto* and *Vehicle*, becomes the label or name of the object in the visualized product. When you run a policy transaction in PolicyCenter, these labels and names appear in the user interface. If a mind map has only one language, the product is always imported as US English (en_US).



You can define mind maps with text for two languages, with English as the second language. In the map, use a caret (^) to separate the text for each language. Enter the English text after the caret.

To add additional languages, create mind maps for each language. For a product that needs French and German, create a French/English mind map and a German/English mind map. The en_US English is used to match the names.

See also

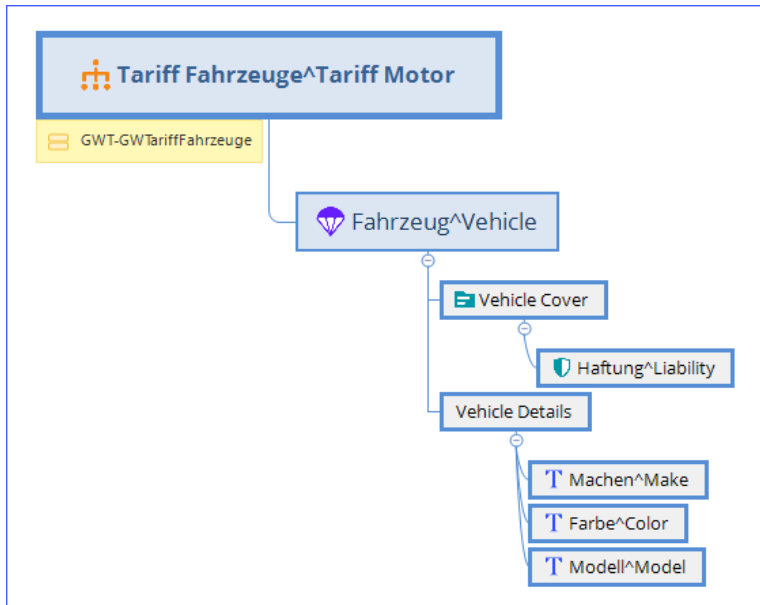
- “Import mind maps for additional languages” on page 43

German/English mind map

In a German/English mind map, the product line name appears first in German (Tariff Fahrzeuge) and then in English (Tariff Motor):

Tariff Fahrzeuge^Tariff Motor

German and English text are provided for objects that appear in the product user interface. Text that does not appear in the user interface, such as Vehicle Cover and Vehicle Details, may be English only.



Japanese mind maps

To create XMind mind maps with Japanese text, you must have the Meiryō UI font installed.

Working with XMind

Installing XMind

Advanced Product Designer can import mind maps created with the free version of XMind 8. XMind is mind mapping and brainstorming software developed by XMind Ltd.

You can download XMind from their website at xmind.net.

Add Guidewire markers to XMind

Guidewire provides XMind markers to tag the elements of your map that are part of the product model and visualization. Use these instructions to import the Guidewire markers into XMind.

Procedure

1. In XMind, create a new blank map by selecting **File→New Blank Map**.
2. In the right sidebar, select **Marker** to display the **Markers** panel.
3. At the top of the Markers panel, select **View Menu→Import Markers**.
4. Download the resource bundle file that contains the Guidewire markers, `GWResourceBundle.xrb` from “Example mind maps” on page 131.
5. In the **Import XMind Resource** dialog, enter the path to `GWResourceBundle.xrb`.

6. Click **Finish**.

The **Guidewire** markers appear in the **Markers** panel.

Create a product mind map

XMind provides a blank canvas on which you can model your product structure. At a minimum, the mind map must contain a product or product line.

Procedure

1. Open XMind.

XMind displays a new, blank map.

2. Click **New Blank Map** to create a new **Central Topic** in an Untitled mind map.

3. In the right side-bar, click to display the **Markers** panel.

4. Change the name of this topic to the name of your product or product line.

5. With the topic selected, add a Product or Product Line marker from the **Markers** panel.

6. Right-click the topic and select **Insert→Label**.

Product and product line labels have the format *<Abbreviation>-<Identifier>*.

<Abbreviation>

Three characters, the first of which must be an uppercase letter and the remaining letters, digits, \$, or _.

Identifier

Starts with an uppercase letter followed by letters, digits, \$, or _.

In the visualized product in PolicyCenter:

- The 3-digit code becomes an **Abbreviation**. In a product line topic, the code becomes the **Line Prefix**.
- The identifier becomes the **Identifier** for the line appended with the word **Line**.

Display labels in XMind

About this task

In a product mind map, you attach labels to the product and product line. Display the labels in the mind map to easily view their contents.

Procedure

1. In XMind, open a map.

2. Click in a blank area of the map.

3. Select **Window→Format** to display the **Sheet Format Panel**.

4. Select **Information Card→Show Label**.

Risk objects in mind map

A risk object captures information about the risks covered by a policy. The risk object can be a tangible object such as a car or a building. The risk object can be a location, which is special type of risk object. Or the risk object can be an intangible object such as a jurisdiction or the policy line. A risk object may or may not have attached coverages.

In the mind map, create risk objects by adding a risk object marker . Use a clause category to attach coverages, exclusions, and conditions, if any, to the risk object.

In the visualized product, you can add one or more instances of a risk object to a policy. For example, you have a cell phone risk object in a mind map. In a policy transaction, you can add one or more cell phones to the policy.

Risk attributes

In the mind map, you create topics for each risk attribute, and attach these topics to the risk object. You can also group these topics under a common Risk Attribute Category topic. For example, add a Policy Attributes subtopic to the product line, add the Risk Attribute Category ? marker to the topic. Add subtopics for the risk attributes. Add markers indicating the type of attribute.

The first text T or drop-down list ▼ attribute on a risk object becomes the identifier. The identifier helps to identify one risk object from another in a list of risk objects.

Modeling locations on risk objects

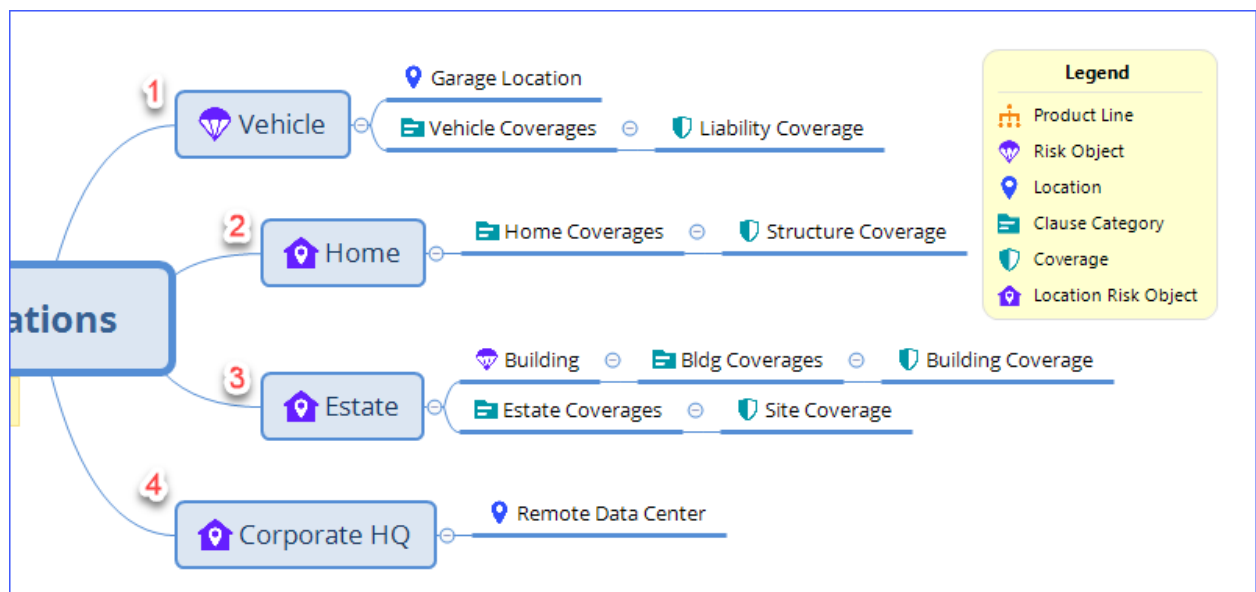
A location risk object is a special type of risk object that is a location. In PolicyCenter, location risk objects automatically include a location field that cannot be removed. An example of a location risk object is the home in a homeowners policy.

In addition, a risk object that is not a location can have an associated location field. For example, as a vehicle risk object has an associated garage location. The garage is not a risk object.

In the XMind, you can model locations on risk objects in the following ways:

1. The risk object is not a location but has a location field. Define the risk as a risk object , and add a child location . For example, the vehicle garage.
2. The location is the risk object. Define it as a location risk object . For example, a home. In the visualized product, the location field is created automatically.
3. The location is the risk object, but has multiple building risk objects . For example, an estate that has a main house, guest house, and pool house.
4. The risk object is a location and has more than one location field. Define the risk object as a location risk object , and add additional location fields. For example, a company's corporate headquarters and their remote data center.

The following mind map shows how to model these:



Buildings

There are various ways to model buildings in the mind map.

In the home example, the location and the building are modeled as a single location risk object .

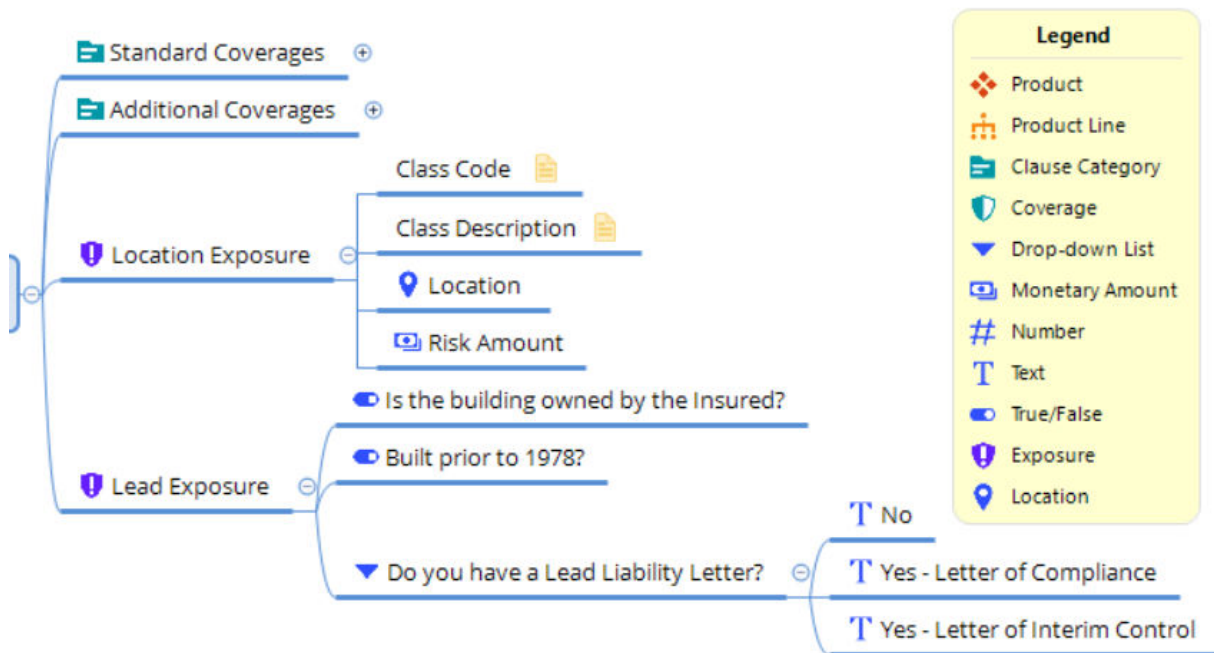
In the estate example, the estate is a location risk object , and has buildings child risk objects .

Exposures in mind map

An exposure represents a potential for accident or loss due to a risk, such as lawsuits or hazardous materials.

In the mind map, you can attach exposures  to the product line  and risk objects . You can attach any of the data types to an Exposure .

The following illustration shows exposures on the product line in General Liability. There are two exposures on the product line: Location Exposure and Lead Exposure. Location Exposure has fields for Class Code, Class Description, Location, and Risk Amount. Class Code and Class Description will be lookup tables, and are not implemented in the mind map. Lead Exposure has two True/False questions and a drop-down list that collect information about lead exposure.



Clauses in mind map

Coverages, conditions, and exclusions are types of clauses. You define clauses and clause terms on risk objects in the mind map. In the visualized product, you add additional information, such as:

- Availability
- Usage for clause terms
- Minimum, maximum, and default values for clause terms
- Metadata for product generation
- Pricing

To add clauses to a risk object in the XMind mind map, add a clause category  to the risk object. Create separate clause categories for each clause type. For example, group coverages in one clause category, and exposures in another clause category. Create clauses as subtopics of the clause category, and add coverage , condition , or exclusion  markers.

Create clause terms on a clause by attaching attribute subtopics.

Clause terms

Clause terms are attributes attached to coverages, exclusions, and conditions.

In the XMind mind map, create clause terms by attaching subtopics to clauses. Use the following markers to create clause terms:

Marker	Represents
T	Text
	Boolean corresponds to True/false
#	Integer corresponds Number
.00	Decimal
	Money corresponds to Monetary amount
▼	Drop Down
	Date-Time

You can use **Drop Down** ▼ clause terms to represent option and package coverage terms, as well as lists.

Events on clauses in mind map

To create events in the XMind mind map, add Event  markers to topics. Events are subtopics of clauses. To group events under a clause, create a subtopic of the clause, and add the events. Do not add a marker to the subtopic. For example, in a Content coverage, add an Events subtopic that contains Fire and Flood events.

You can reference the predefined events in your mind map by entering the name.

If you import a new event from a mind map, the event is added to PolicyCenter. The name of the event is case-sensitive and must be unique.

Predefined events

There are a number of predefined events that are available to all products. The case-sensitive names are:

- Animal Bite
- Animal Damage
- Collision
- Damage by Third Party
- Earthquake
- Fire
- Flood
- Hail
- Landslip
- Landslip
- Lightning
- Loss of Keys
- Mallorca Protection
- Own Property Liability
- Riot and Strike
- Rodent Damage
- Sonic Boom
- Special Equipment Damage
- Storm
- Tap Water
- Theft
- TP Bodily Injury
- TP Property Damage
- TP Vehicle Damage
- Vandalism/Malicious Damage
- Windscreen Damage

Covered costs on clauses in mind map

You define covered costs on clauses. The insured recoups covered costs through claims.

To create a covered cost in the XMind mind map, add a Covered Cost  marker to a topic. Covered costs are subtopics of clauses. To group covered costs under a clause, create a subtopic of the clause, and add the covered costs. Do not add a marker to the subtopic. For example, in a Legal Expenses coverage, add a Costs subtopic that contains Contract Disputes and Legal Defense covered costs.

You can reference the predefined covered costs in your mind map by entering the name.

You can add new covered costs to PolicyCenter by adding them to the mind map and importing the mind map. The name of the covered cost is case-sensitive and must be unique.

Predefined covered cost

There are a number of predefined covered costs that are available to all products. The case-sensitive names are:

- Age/Disabled Equipment
- Alternate Accommodation
- Car Valet
- Compensation (general)
- Legal
- Locks and Keys
- Loss of Earnings
- Loss of Rent
- Medical
- Property Repairs
- Property Replacement
- Rental Vehicle
- Rental Vehicle
- Site Clearance
- Vehicle Recovery
- Vehicle Repairs
- Vehicle Replacement
- Windscreen

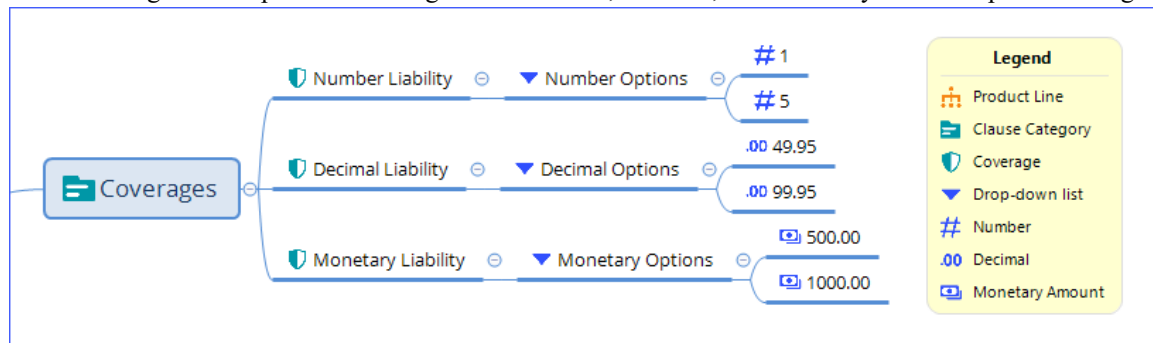
Defining option clause terms in mind map

Option clause terms let you choose from a predefined list of options. Option terms can have labels like 100, 500, or Statutory. Each option term has a numeric value. In the visualized product you can set the return value for each term. The term labeled 100 might return 100 or a multiplier of 100. The Statutory term also returns a numeric value.

In a mind map, you can define option clause terms in a drop-down list by attaching a drop-down list to a Coverage, Condition, or Exclusion. Add the option clause terms as topics under the drop-down list. The terms can be Monetary Amount, Number, or Decimal, but must all be of the same type.

If the list item marker type is Money in the mind map, the clause term **Value** defaults to **Money** type in the visualized product. If the marker type is Number or Decimal, the clause term **Value** defaults to **Other**. You can change the **Value** to a more appropriate type such as **Count**, **Days**, **Hours**, **Money**, or **Percent**.

The following mind map shows coverages with Number, Decimal, and Monetary Amount option coverage terms.



Text marker for option clause terms

For option clause terms with text labels, Guidewire recommends that you use the marker appropriate for the value, such as Number, Decimal, or Monetary Amount. Text items do not import by default to option clause terms in the visualized product. However, if the mind map has Text markers for the option clause terms, you can change them to option clause terms in the visualized product.

Defining package clause terms in mind map

Package clause terms are terms offered as a group. Each term in the package returns a numeric value. In the visualized product, you can set the numeric value for each term. For example, a Liability - Bodily Injury and Property Damage coverage on a car offers package coverage terms, that group three terms. The first package offers 50, 100, 50 terms. The term labeled 50 might return 50 or a multiplier of 50. Like the other package terms, the Statutory package term corresponds to three terms.

- 50/100/50
- 50/200/50
- Statutory

In a clause term, the package terms can have different numbers of values. For example, the following clause term has packages with three, two, and one terms:

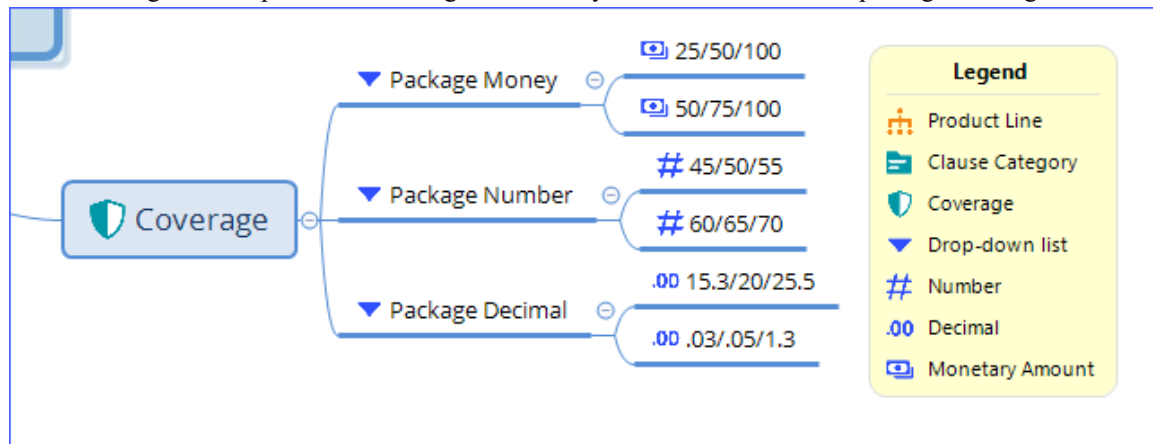
- 50/100/50
- 1000/1000
- 100K CSL

In a mind map, you can define package clause terms in a drop-down list. For the package clause terms, use a slash character (/) to separate the values. Attach a drop-down list to a Coverage, Exclusion, or Condition. Add the package clause terms as topics under the drop-down list. Add Money, Number, or Decimal markers to the package clause term topics:

- 25/50/100
- 50/75/100

If the list item marker type is Money in the mind map, the clause term **Value** defaults to **Money** type in the visualized product. If the marker type is Number or Decimal, the clause term **Value** defaults to **Other**. You can change the **Value** to a more appropriate type such as **Count**, **Days**, **Hours**, **Money**, or **Percent**.

The following mind map shows a coverage with Money, Number, and Decimal package coverage terms.



Text marker for package clause terms

For package clause terms with text labels, Guidewire recommends that you use the marker appropriate for the value, such as Number, Decimal, or Monetary Amount.

See also

- “Changing Text markers to option clause terms” on page 58

Defining subclauses in mind map

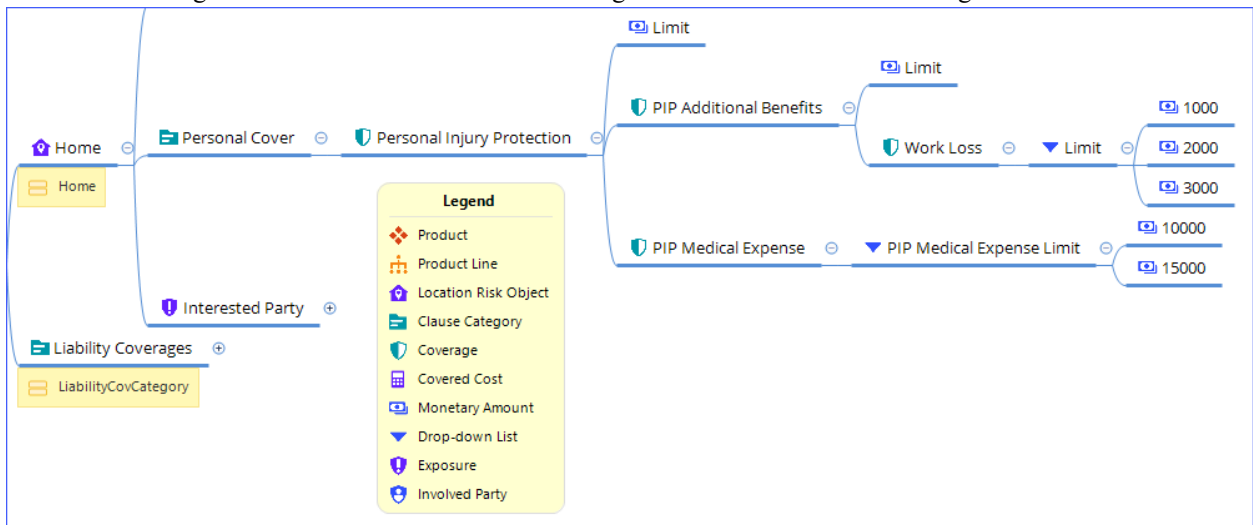
You can create subclauses, or clauses with parent-child relationships, in a mind map. For example, a coverage can have child coverages.

- A clause can have zero or one parent clause
- A clause can have multiple subclauses
- You can create multiple levels of subclauses
- A subclause and its parent must be of the same clause type.

For example, an exclusion cannot be a subclause of a coverage. PolicyCenter displays a validation error if it encounters this when importing a mind map.

The following mind map shows a Personal Injury Protection coverage that is a parent to PIP Additional Benefits and PIP Medical Expense Limit coverages. The child coverages are nested directly underneath the Personal Injury

Protection coverage. The PIP Additional Benefits coverage has a Work Loss child coverage.



Schedules in mind map

Schedules capture information about scheduled items. For each scheduled item, each piece of information is a *schedule term*, such as a name or description. Schedules are often referred to during underwriting and are included in forms.

Some schedule terms are passed to rating and potentially affect the cost of the policy. These terms are often, but not necessarily, Decimal, Monetary Amount, Option, or Package schedule terms.

If a schedule term is modeled in the mind map as a drop-down list with Number, Decimal or Monetary Amount items, then the list becomes an Option drop-down type in the visualized product. If the value contains the slash character (/), it is imported as a Package drop-down type. In the visualized product, you can change the drop-down type to Typelist or another value.

In schedules, dependent attribute relationships in drop-down lists are not supported. They are dropped when you import the mind map.

Jewelry schedule example

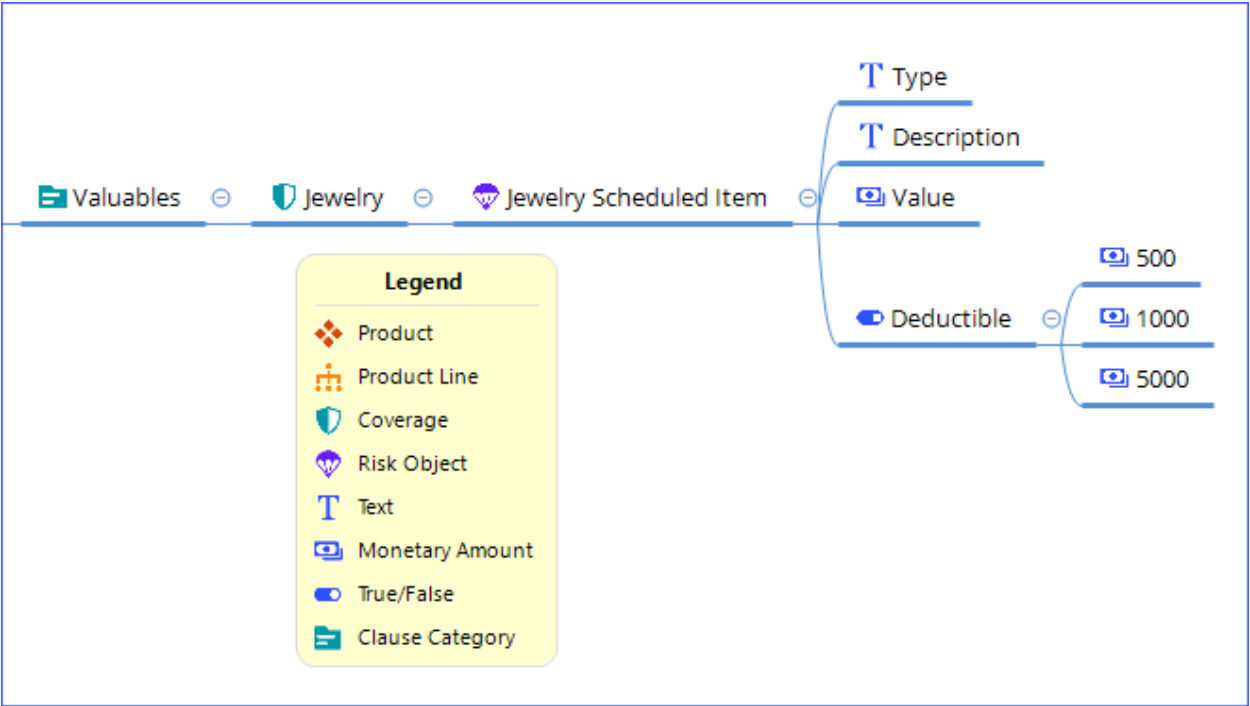
For example, a Jewelry Scheduled Item has the following schedule terms:

- Type – Type of jewelry as Text
- Description – Description of the piece of jewelry as Text
- Value – Value of the jewelry as a Monetary Amount
- Deductible – Drop-down list with \$500, \$1,000, and \$5,000 Monetary Amount choices

In a policy, the Jewelry schedule contains the following information:

Type	Description	Value	Deductible
Pearl necklace	Single strand, 16 inches, white pearls	\$4,500	\$500
Diamond engagement ring	3 carats of diamonds in 18 karat white gold	\$17,000	\$5,000
Wedding band	18 karat white gold	\$7,250	\$1,000

To create a schedule in XMind, create a clause (Jewelry coverage) with a risk object (Jewelry Scheduled Item). On the risk object, attach schedule terms of any of the data types, such as Text, Monetary Amount, Drop-down list. The following illustration shows the mind map for a jewelry schedule.



Attributes in mind map

When writing a policy, you may require specific information or answers to question about parts of the policy. You can model this information using attributes. For example, in a small business policy, you need to know whether the insured has been refused insurance in the past. On a building, you may need to know the number of stories. Attributes can be attached to various parts of the product, such as the product line, risk objects, and exposures. Attributes can impact the pricing of a policy and drive underwriting decisions.

In the mind map, you create topics for each attribute. To organize attributes, you can group these as subtopics under a common topic. Add markers to each topic indicating the type of attribute.

You can group related questions under a common topic. For example, a Vehicle Questions subtopic on the Vehicle coverage contains attributes that are questions about the vehicle. This is similar to, but not the same as, creating a *question set*. You can use Product Designer to create question sets.

The attribute marker types are:

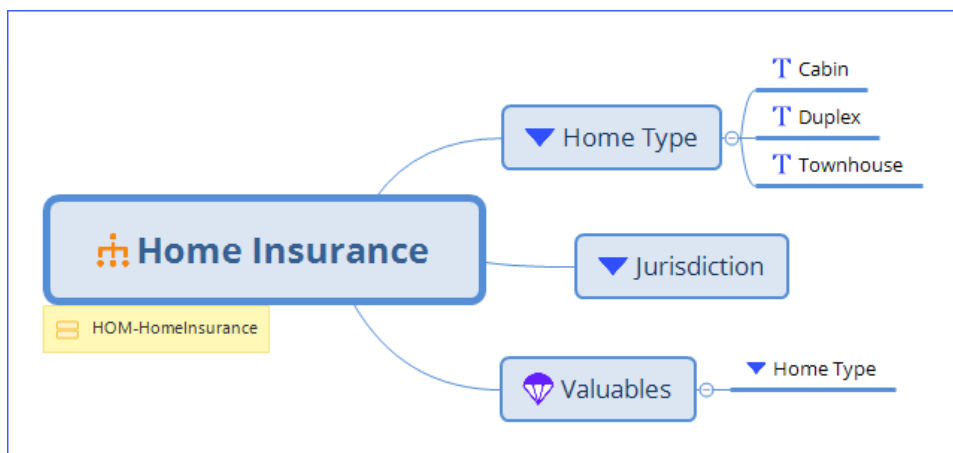
Marker type	Marker	Represents
Data types	T	Text
	#	Number
	.00	Decimal
		True/false
		Date-time
		Drop-down list
		Monetary Amount

Marker type	Marker	Represents
		Involved Party
		Location

Drop-down lists in mind map

Drop-down lists let you choose an option from a predefined list of choices. You can add drop-down lists to risk objects and exposures. You can also add drop-down lists to clauses (coverages, conditions, and exclusions), to create option and package clause terms.

In the mind map, add a Drop-down list ▼ marker to the topic, and attach subtopics for each item in the list. Add Text T, Number #, Decimal .00, and Money \$ markers to the subtopics. All list items must be of the same type.



Coverage terms

You can use drop-down lists to model coverage terms.

See also

- “Defining option clause terms in mind map” on page 31
- “Defining package clause terms in mind map” on page 31

Use existing drop-down list or PolicyCenter typelist

You can create a drop-down list that gets its values from another drop-down list. This list can be:

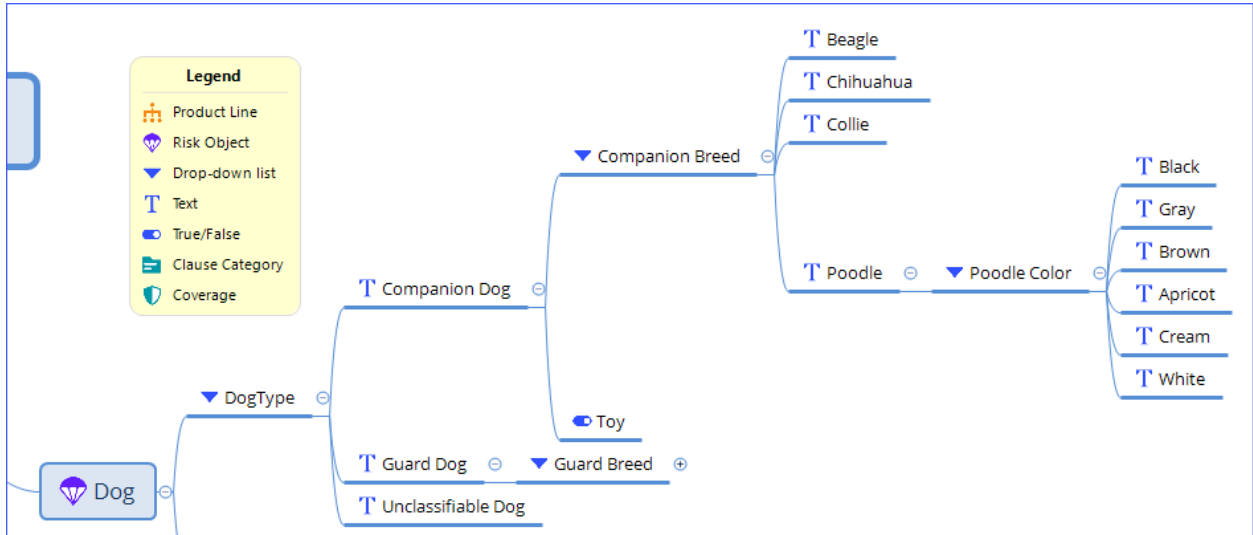
- A drop-down list defined in the current product – In the example mind map, Home Type on Valuables is an example.
- One of the many PolicyCenter typelists. PolicyCenter includes general purpose typelists such as Country, Jurisdiction, and Currency. PolicyCenter also includes typelists defined in generated products. In the example mind map, Jurisdiction is an example of a general purpose typelist. After you generate the Home Insurance product, Home Type becomes a typelist that other products can use.

To create a drop-down list that gets its values from another list, create a drop-down list ▼ topic with the name of the list. Do not add list items.

Nested drop-down lists with dependent relationships

You can model dependent relationships in mind maps by attaching attributes to drop-down list items. For example, in a Pet Insurance product, the Dog risk object has a Type drop-down list where you can specify whether the dog is a Companion Dog, Guard Dog, or Unclassifiable. If you select Companion Dog, you can further specify the breed and

whether the dog is a toy. The breed is a drop-down list, and Poodle includes a nested Poodle Color drop-down list. Companion Dog also has a Boolean Toy attribute. You cannot attach attributes to the Toy attribute.



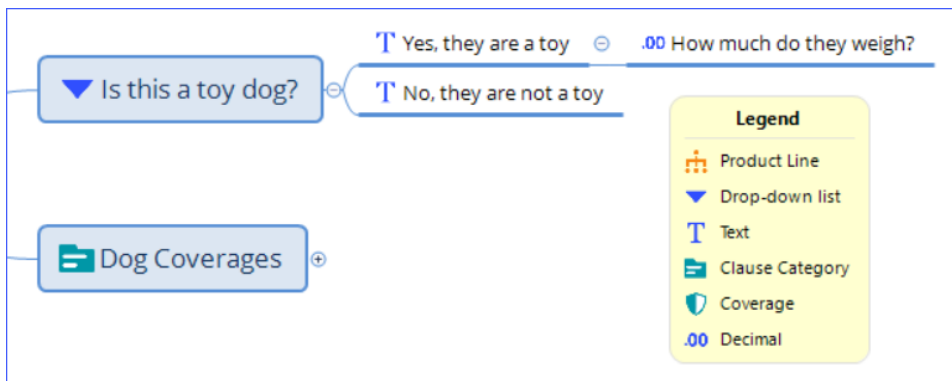
A drop-down list can have Text, Number, Money, and Decimal attributes. Beneath any of these list items, you can attach data type attributes, such as Text, Drop-down list, and Number, among others. If you attach a drop-down list, you can continue attaching attributes on the list items. This relationship is preserved when you import the product to PolicyCenter.

In schedules, dependent attribute relationships in drop-down lists are not supported. They are dropped when you import the mind map.

Use drop-down list for follow-up to True/False questions in mind map

You can model follow-up questions to a True/False (binary choice) question in mind maps by using a Drop-down list. This is because True/False topics cannot have subtopics. They are deleted when you import the mind map, and you cannot model this relationship in Advanced Product Designer.

The following example has a binary choice question, *Is this a toy dog?*, that is modeled as a drop-down list. The *Yes, they are a toy* choice has a follow-up question *How much do they weigh?*



Yes, they are a toy, *No, they are not a toy* are text. *How much do they weigh?* is decimal.

See also

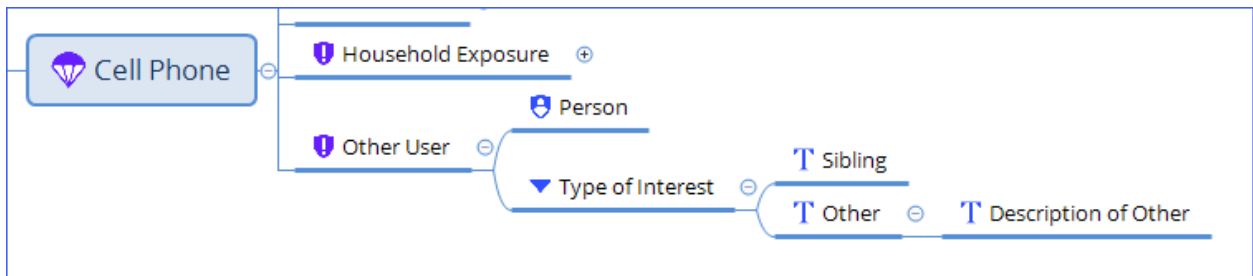
- “Use drop-down list for follow-up to true/false questions in visualized product” on page 69

Involved party in mind map

An involved party is a contact (person or company) who has an interest on the policy. In the product model, you can attach involved parties to the policy line, risk objects, and exposures.

Note: If you attach involved parties to objects others than policy line, risk objects, or exposures, they are imported as children of the nearest ancestor that is a policy line, risk object, or exposure.

For example, in a mind map, a cell phone risk object has an Other User exposure where you can specify contacts that are other users of the phone in the involved party  Person field.



Usage and availability notes in mind map

Specify whether a product attribute is derived, hidden, unavailable, or user entered by default. You can also specify conditional exceptions that change usage and availability for special circumstances.

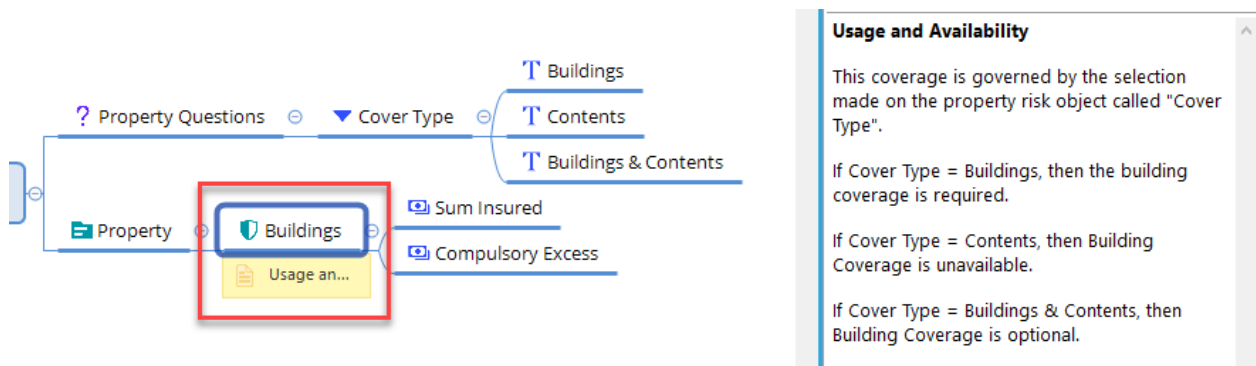
During the conceptualization phase in XMind, you can identify usage and availability properties for attributes, and describe these properties in notes in the mind map. Usage properties can apply to any attribute. Availability properties apply to clauses and drop-down list items.

Note: In the mind map, usage and availability properties are informational only; they are dropped when you import the mind map into PolicyCenter.

To easily identify these properties, you can add the words **Usage** and **Availability** at the top of the Note before adding your description.

Refer to the this note when you implement the usage or availability property in the visualized product.

The following mind map models a product line with a Buildings coverage. The availability of the Buildings coverage is governed by the Cover Type drop-down list attribute.



The availability for the Buildings coverage is described in the Note on Buildings:

Usage and Availability
 This coverage is governed by the selection made on the property risk object called Cover Type.

If Cover Type is Buildings, then Building Coverage is required.
 If Cover Type is Contents, then Building Coverage is unavailable.
 If Cover Type is Buildings & Contents, then Building Coverage is optional.

Extract usage and availability notes from mind map

You can extract the usage and availability notes from a mind map into HTML format. You can use these notes as a guide for adding usage and availability to the visualized product.

Procedure

1. In XMind, select **File**→**Export**.
2. Select **Document**→**HTML File**, then click **Next**.
3. In the Generate HTML File dialog, select **Overview Picture** and **Separate Overview Pictures**.
4. Under **All**, select **Notes**. Enter a file name in **To file**, and click **Finish**.

XMind generates the HTML file, and displays it in the default browser.

Pricing in mind map

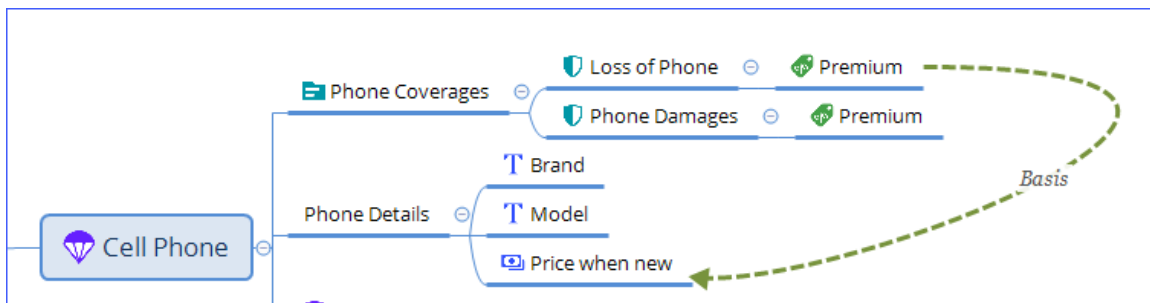
In the mind map, you can model the following pricing relationships.

Note: Pricing in the mind map is informational only; it is not imported into the visualized product.

Modeling pricing to basis

In the mind map, you can add a Pricing marker to a topic, and add a relationship between the Pricing topic and its basis.

For example, a Cell Phone risk object has a Loss of Phone coverage with a Phone Premium pricing topic. The basis of Phone Premium is Price when new. Price when new is a monetary amount topic under Phone Details.



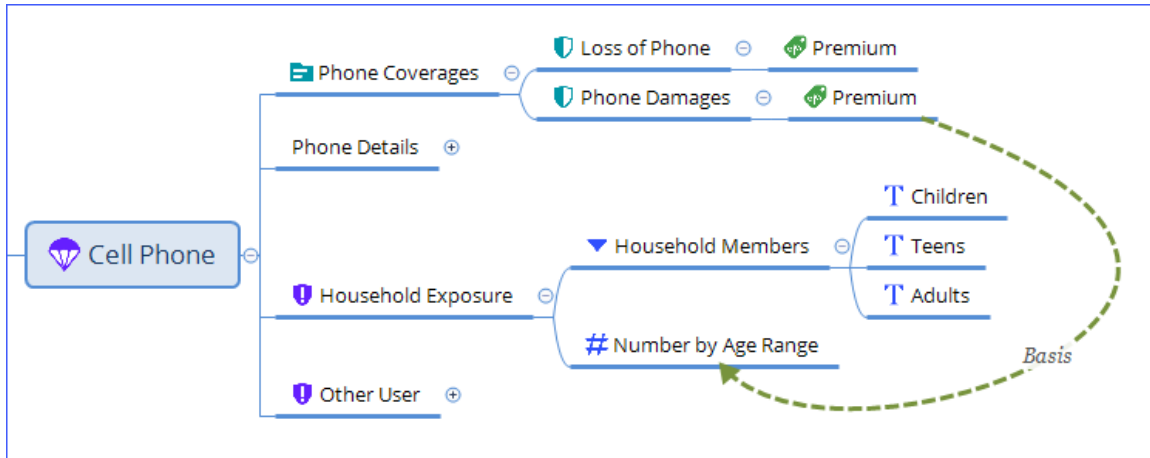
When you import a mind map, Pricing markers and their relationships are dropped. You can use the pricing relationships from the mind map as a guide to adding pricing information to your visualized product.

Modeling exposure pricing

With exposure pricing, exposure to risk provides the basis for rating. For example, the cost of Phone Damages coverage is based on household members of various age ranges.

In the mind map, define an exposure and exposure fields. In this example, the Cell Phone risk object has a Household Exposure with Household Members drop-down list. Household members can be Children, Teens, or Adults. The exposure also has a Number by Age Range number topic where you can specify the number of household members in a particular age range.

The Phone Damages coverage has a Premium pricing topic that will be based upon Number by Age Range. When you import the mind map, pricing topics and their relationships are dropped. You recreate this relationship in the visualized product.



Generate mind map from product

You can generate a mind map in XMind format from a visualized or externally managed product in PolicyCenter.

Procedure

1. In PolicyCenter, save the product to template.
 - Installed product – Go to the **Installed Products** tab on the **Product Management** screen, and click **Generate Template**.
 - Visualized product– Go to the **Visualized Products** tab on the **Product Management** screen, edit the product and click **Export to Template**.
2. In a Web browser, go to <https://tools.app.cloud.guidewire.net/>.
3. Click **APD Mind Map Generation**.
4. Follow the instructions on that page.

Visualizing products in PolicyCenter

You conceptualize your product in an XMind mind map. When the mind map has reached a level of stability, you use Advanced Product Designer to import it into PolicyCenter.

Advanced Product Designer creates a *visualized product* in PolicyCenter. You can demonstrate, maintain, and finalize the visualized product in PolicyCenter. You can create a policy submission and navigate through the product screens.

Product design without a mind map

You do not need a mind map to define your product requirements. You can design the product directly on **Visualized Products** screens in PolicyCenter.

Saving your work

Export your visualized product to *template* format to save your work.

Guidewire recommends that you use template format for saving and archiving your visualized products. You can use a version control system to store the template which is in XML format. The template contains product details from the **Product Definition** screens.

Product design modes in Advanced Product Designer

Advanced Product Designer adds a **Product Design Mode** setting to PolicyCenter. This offers increasing levels of interaction with visualized products.

To set **Product Design Mode**, select **Preferences** from the Options menu. The modes are:

Mode	Description
Hidden	Visualized products are hidden.
User	Visualized products can be demonstrated without revealing design elements.
Designer	You can add and modify visualized products in the Product Management screen. In submissions and other policy transactions, you can make changes to the visualized product. This is the default value.
Developer	Same as Designer , and fields related to product code generation are available. You can generate the product.

Production system

Advanced Product Designer is unavailable when PolicyCenter is in production mode. In production mode there is no **Product Design Mode** setting, and the **Product Management** screen does not let you create or modify visualized products.

Editing visualized products

You can edit visualized products in PolicyCenter:

- From the **Product Definition** screen
- In a policy submission or other policy transaction

Edit in Product Definition screen

Set **Product Design Mode** to **Designer**.

You can access the **Product Definition** screen by navigating to the **Product Management**→**APD Managed** tab, selecting a product, and clicking **Edit Product**. These screens are designed specifically for viewing product details and making changes.

Edit in submissions and other policy transactions

You can use a submission for demonstrating the product to others. To see the product as a user, set **Product Design Mode** to **User**. In the submission you can view product details, choose coverages, and set values on the **Risk Details** and **Pricing** screens. You can also try out the visualized product in other policy transactions.

In a policy transaction, you can make changes to the visualized product. Set **Product Design Mode** to **Designer**. The **Risk Details** and **Pricing** screens display extra fields, such as **Minimum** or **Maximum**, for changing the product model. For example, if a minimum or maximum value is in error, you can change the value.

In a policy transaction, you can use the **Toggle User Mode** button to switch back and forth between **User** and **Designer** modes. When demonstrating a product submission, toggle to **User** mode to view screens as an end user. If someone requests a change, toggle to **Designer** mode and change the product. Then toggle back to **User** mode.

Create visualized products

Create visualized product from mind map

Create a visualized product by selecting **Import From XMind** to import a mind map into PolicyCenter.

Before you begin

- Set **Product Design Mode** to **Designer** or above.

Procedure

1. Navigate to one of the following screens:
 - On the **Product Management** screen, the **APD Managed** tab, select **Import From XMind**.
 - On the **New Submissions** screen, the **Visualized Products** tab, select **Add Product**→**From XMind**.
2. In the **Load Product** popup, browse to select your mind map.

Default additional fields? specifies whether default values are created for metadata fields labeled (*PD).

If **Product Design Mode** is **Developer** and you are importing a mind map, **Default additional fields?** appears. The default setting is **No**. If you select **Yes**, default values are created for metadata fields labeled (*PD).

Only set **Default additional fields?** to **Yes** to automatically create the (*PD) fields so that you can quickly generate a product for demonstration or testing. When designing a product for actual use, you do not want to automatically create these fields. When designing a product, create short names that follow best-practice naming conventions agreed upon by the organization or team.

3. Click **Update**.

PolicyCenter imports the mind map and adds the product to the table.

Create visualized product directly in PolicyCenter

You can create a visualized product directly in PolicyCenter, without starting from a mind map.

Before you begin

- Set **Product Design Mode** to **Designer** or above.

Procedure

1. In PolicyCenter, select **Administration**→**Product Management**.
PolicyCenter displays the **Product Management** screen with **APD Managed** tab.
2. On the **APD Managed** tab, select **New Product**.
PolicyCenter displays the **Load Product** screen.
3. Enter values for **Product** fields as described in “Product fields on Product Definition screen” on page 48.
4. Enter values for **Product Line** fields as described in “Product line Details tab” on page 49.
5. When you have complete filling out product and line details, click **Update**.
The visualized product is created and added to the table.

Create visualized product from template

You can move a visualized product from one PolicyCenter environment to another by saving the visualized product to template and importing the template to the other environment.

Before you begin

- Set **Product Design Mode** to **Designer** or above.

Procedure

1. Navigate to one of the following screens:
 - On the **Product Management** screen, the **APD Managed** tab, select **Import From Template**.
 - On the **New Submissions** screen, the **APD Managed** tab, select **Add Product**→**From Template**.
2. In the **Load Product** popup, browse to select your mind map.
3. Click **Update**.
PolicyCenter imports the template and adds it to the table.

Import mind maps for additional languages

Before you begin

Create mind maps with multiple languages as described in “Display languages in mind maps” on page 25.

About this task

Advanced Product Designer can import mind maps that provide display text for two languages, with English (en_US) as the second language. For more than two languages, create additional mind maps for those languages. For a product that needs French and German, create a French/English mind map and a German/English mind map, where the English is the same. Importing merges the maps based on matching the English names.

When you generate the product, display properties are generated for every language you have defined.

Procedure

1. In PolicyCenter, enable the display languages needed for the product. See *Globalization Guide*.
2. In PolicyCenter, go to the **APD Managed** tab of the **Product Management** screen.
3. Select **Import From XMind** to import the first mind map with two languages.
4. In the **Local Language** field of the import screen, select the additional language, such as German or French.
5. To import more languages, use **Import From XMind** to import the mind maps with those languages.

Start policy submission for visualized product

A submission for a visualized product lets you navigate through the screens of a policy submission. You can add coverages, specify policy details, quote, bind, and issue a policy.

Procedure

1. In PolicyCenter, navigate to an account.
You can use a pre-existing account. For example, if you have loaded the small sample data set, you can use the Ray Newton account. Otherwise, you can create a new account.
2. From the **Actions** menu, select **New Submission**.
3. On the **New Submissions** screen, select the **APD Managed** tab, and click **Select** for one of the visualized products.
4. You can navigate through the product screens, fill in policy information, add coverages, and then issue the policy.

Remove visualized product

Before you begin

- Set **Product Design Mode** to **Developer**.

About this task

Note: Removing a visualized product withdraws open policy transactions, such as submissions. You can still view withdrawn policy transactions after removing the product, but policy data, such as coverages, is missing.

Procedure

1. In PolicyCenter, select **Administration**→**Product Management**.
PolicyCenter displays the **Product Management** screen with **APD Managed** tab.
2. On the **APD Managed** tab, select **Remove Product**.
Remove Product is unavailable if there are bound policy transactions on the product, or if the product has policy transactions that cannot be automatically withdrawn. For example, this occurs if you do not have withdraw permission.

Update visualized product

You can update a visualized product from its mind map or template.

Before you begin

- You have:
 - A mind map created in XMind or template file of a product
 - A visualized product created from that mind map
- Set **Product Design Mode** to **Designer** or above.

About this task

In the mind map, you make changes, such as corrections or additions, and update the visualized product. If you change the name of a topic, this becomes an addition when you update the visualized product. If you delete a topic from the mind map, it is not deleted when you update the visualized product. You must delete them manually from the visualized product.

When importing a template with conditions or variants, the priority of conditions or variants in the template supersedes the priority in the visualized product.

When importing a template, all changes, including corrections, additions, and deletions, are reflected in the visualized product.

Procedure

1. In PolicyCenter, select **Administration**→**Product Management**.
PolicyCenter displays the **Product Management** screen with **APD Managed** tab.
2. In the table at the top of the **APD Managed** tab, select the product you wish to update.
The visualized product appears in the **Details** tab.
3. Select **Update Product** and choose **From Template** or **From XMind**.
4. In the **Load Product** popup, browse to select your mind map or template.
Default additional fields? specifies whether default values are created for metadata fields labeled (*PD).
If **Product Design Mode** is **Developer** and you are importing a mind map, **Default additional fields?** appears. The default setting is **No**. If you select **Yes**, default values are created for metadata fields labeled (*PD).
Only set **Default additional fields?** to **Yes** to automatically create the (*PD) fields so that you can quickly generate a product for demonstration or testing. When designing a product for actual use, you do not want to automatically create these fields. When designing a product, create short names that follow best-practice naming conventions agreed upon by the organization or team.
5. Click **Update**.
The visualized product is updated.
All visualized products are validated to prevent duplicates in the product Abbreviation and other fields. The validation does not check against products on the **Externally Managed** tab.

Export visualized product to template

From PolicyCenter, export a visualized product to a template in XML format.

Before you begin

- You must have at least one visualized product.
- Set **Product Design Mode** to **Designer** or above.

Procedure

1. In PolicyCenter, select **Administration**→**Product Management**.
PolicyCenter displays the **Product Management** screen with **APD Managed** tab.
2. In the table at the top of the **APD Managed** tab, select the product you wish to export.

The visualized product appears below in the **Details** tab.

3. Click **Export Template** to export the current product to a template.

The template is downloaded to your system. The file name begins with the product Identifier and has an XML file extension.

Working with visualized products

These topics describe how to model product details in Advanced Product Designer screens. The product may have been imported from a mind map.

Details not imported from mind map

You can model product details such as pricing relationships in the mind map. However, these relationships are not supported for import and are dropped. After you have created a visualized product, you can add these details that are not imported from the mind map. You can:

- Edit the product line, risk objects, coverages, exclusions, conditions, and other parts of the product model
- Add lines to a product
- Define pricing
- Create usage properties that specify how a value is obtained and when it is available
- Set minimum and maximum values

Products and product lines

A *product* is an insurance product, such as personal auto or commercial package, that an insurance company offers for sale to its customers. A product sale results in a policy. A *product line*, sometimes referred to as *line*, is contained in a *product*. Multiline products contain one or more product lines.

Mind map

In XMind, you map the product structure into a tree-like diagram where the root of the tree is either a product or a product line. Add a Product  or Line  marker to the topic.

See also

- “Create a product mind map” on page 27
- “Labels on product and product line” on page 20

Visualized product

In PolicyCenter, you can view and modify the product on the **Product Definition** screen. Information about the product line is on the **Details** tab.

Product fields on Product Definition screen

These are the fields in the **Product** section of the **Product Definition** screen.

Field	Description
Name	Name for the new product.
Description	Description of the product.
Identifier	A unique identifier for the new product. For example, SmBus for a Small Business product. See “Labels on product and product line” on page 20.
Abbreviation	Define a unique, 3-character abbreviated code for the new product. If this is a single line product, then PolicyCenter uses this code for the Line Prefix in Product Line . See “Labels on product and product line” on page 20.
Multiline	Does this product contain more than one product line? If No , you can define the product line’s policy details in the Product Line section below. If Yes , you define the product line details within the product.
Locations List in Wizard	Can this product insure more than one location? If Yes , then include a screen for defining multiple locations. A Locations step appears on the left, at the same level as Policy Info in a submission.
Coinsured	Does coinsurance apply to this product? If Yes , you can define coinsurance details in the product.
Currencies	Specify currency in the product. Values are: • Domestic - Policies are single currency and use the currency defined in config.xml. • Single - A policy will be in a chosen currency. However, any number of currencies can be defined in the system. • Multiple original - The policy is multi-currency. Premiums and sums insured can be defined in any number of currencies. There is a single settlement currency into which all amounts are converted for billing (this cannot be the main policy currency). • Multiple settlement and original currencies - Original currencies as above, but may bill in a number of settlement currencies (typically based on the countries of risk). PolicyCenter must be configured to support this.
Account Type	Specify the type of account that can take out policies for this product: • Any – Both companies and people • Company • Person This affects which product are listed for a new submission.
Enabled for REST API	If there are both visualized product artifacts and finalized product artifacts, this field identifies which set of artifacts will be referenced by the system APIs. For more information, see the <i>Configuration Guide</i> .

Product line Details tab

When creating a new product line or editing an existing product line, you can edit and add line details in the **Details** tab of the **Product Definition** screen.

Name	Name for the new product line.
Description	Description of the product line.
Line Prefix	A unique 3-character code for the product line. See “Labels on product and product line” on page 20.
Identifier	A unique identifier for the new product line. For example, use SmBus for a Small Business product line. See “Labels on product and product line” on page 20.
Currencies	Specify currency in the product. Values are: • Domestic - Policies are single currency and use the currency defined in PolicyCenter. • Single - Policies will be in a chosen currency. However, any number of currencies can be defined in the system. • Multiple original - Policies are multicurrency. Premiums and sums insured can be defined in any number of currencies. There is a single settlement currency into which all amounts are converted for billing. (The settlement currency cannot be the main policy currency.) • Multiple settlement and original currencies - Multiple original currencies as above, but may bill in a number of settlement currencies (typically based on the countries of risk). PolicyCenter must be configured to support this.
Product Line Type	Defines the coverage types as well as where those coverages apply (policy or risk object covering single or multiple risks, for example) that will be available for the defined product line. For all choices except Liability Only – Single Risk , the Has Child Risks field appears.
Has Child Risks	Does the line have only a single risk, or does the line have other risks, modeled as child risk objects? If Yes , define the Children’s UI Label .
Children’s UI Label	Appears if you answer Yes to Has Child Risks . Enter a label for the screen that contains the risks. This screen contains tabs for Coverages , Additional Coverages , and Exclusions and Conditions . For example, Equipment.
Multiple Exposure For Insureds’ Risk	Can more than one exposure can be defined for each risk? If Yes , define the UI Label for the exposures. If No , a risk is always associated with a single exposure.
Exposure’s UI Label	Appears if you answer Yes to Multiple Exposure For Insureds’ Risk . Enter a label for the exposures.
Menu Label for Generated UI (*PD)	Define the Product Name label that appears on the New Submissions screen when you are creating a new policy. After you generate the product, the product uses this label. It appears as the label in the job wizard and policy file step. It does not appear in the visualized product.
PolicyLine Subtype, excluding prefix (*PD)	The name of the entity generated for this product line. The entity is a subtype of PolicyLine. This field appears if you are in Developer mode.

Fields tab

Describes the **Fields** tab in **Product Definition** screens.

Note: Fields required for product generation are marked (*PD).

Field	Description
Label	Label that appears in the user interface.
Type	The type of data, such as text, number, monetary amount, or date-time. Involved party creates a contact field on the policy. Contact details include name, phone number, address, and other information. When viewing the Contacts for the policy, these contacts have the Involved Party role. The Involved Party  marker in a mind map becomes an involved party contact.
Identifier	The identifier helps to identify one risk object from another in a list of risk objects. The Risk Details screen in a policy transaction lists the risks on the policy. In the list of risk objects (Risk Name), the identifier appears after name of the risk object. For example, on Pet risk objects, the Pet Name text field is the identifier. In a policy with pets named Fido and Peony, Risk Name shows the two risks: • Pet: Fido • Pet: Peony Identifier fields directly on a policy line do not appear in a policy transaction.
Jurisdiction	The jurisdiction for location fields on child risk objects. If you select this, then the risk object become a location risk object. Does not appear on product line fields because the product line cannot be a location risk object.
Ordering	The order of the items in the user interface, from first to last.
Field Usage	How the product uses the field. Field values can be: • Derived – Value is derived programmatically, and is read-only in the user interface. In the generated product, create code in Guidewire Studio that derives the value. • Hidden – Value is stored in the database but is not displayed to the user. • Unavailable – The field is not available and cannot be added to the policy. • User Entered – Value is entered by the user. • User Entered: Required for Bind – Value is entered by the user and is required to bind the policy. Has no effect in a visualized submission. • User Entered: Required for Issue – Value is entered by the user and is required for issuance. Has no effect in a visualized submission. • User Entered: Required for Quote – Value is entered by the user and is required at quote. Has no effect in a visualized submission.
Default Value	Default value for the field.
Minimum Value	Minimum value for the field.
Maximum Value	Maximum value for the field.

Field	Description
Description (*PD)	Description of the field. This field is required for product generation and displays if Product Design Mode is Developer .
Short Name (*PD)	This field is required for product generation and displays if Product Design Mode is Developer . Unique name that identifies the field in the generated product. This name must be unique in the product line.
Category	This field is not used. Do not enter a value.
Split By Rating Periods	This field can have split rating periods. Split rating periods create separate rating periods for dates selected by the user.
Scalable	Whether to use basis-scalable rating. When selected, the basis is scaled automatically before and after the split date. You can also manually scale the basis by editing the field values. This field is enabled if you select Split By Rating Periods , and Type is Number , Decimal (14,2) , or Monetary amount .

Field naming conventions

In PolicyCenter, when you define new fields, the following naming conventions apply:

Name

Must start with an uppercase letter and can only include letters, digits, \$, _, and the space character.

Identifier

Must start with an uppercase letter and can only include letters, digits, \$, _, and the space character.

Abbreviation

Must start with a letter or digit and can only include letters, digits, and the characters \$ or _.

You cannot use special characters in **Typelists** and **Line**, **Exposure**, and **Risk Object/Coverable** fields.

Risk objects in visualized product

A risk object captures information about the risks covered by a policy. The risk object can be a tangible object such as a car or a building. The risk object can be a location, which is special type of risk object. Or the risk object can be an intangible object such as a jurisdiction or the policy line. A risk object may or may not have attached coverages.

You can use *risk attributes* to obtain specific information or answers to question about parts of the policy. For example, in a small business policy, you need to know whether the insured has been refused insurance in the past. On a building, you may need to know the number of floors.

In the visualized product, you can access risk objects from the **Children** tab at various levels of the product hierarchy. Click a risk object to view or modify it on the **Risk Object Definition** popup.

The product line is a type of risk object.

Note: In PolicyCenter, if you make changes to risk object fields that were imported from the mind map, remember to update the mind map with the corresponding change.

Attributes on the product line and risk objects appear on the **Fields** tab of the **Product Definition** screen. In a submission on the **Risk Details** screen, these attributes appear on the **Fields** tab.

You can specify the risk attribute category in the **Category** column on the attribute.

Location risk object in visualized product

A location risk object is a special type of risk object that is a location. In PolicyCenter, location risk objects automatically include a location field that cannot be removed. An example of a location risk object is the home in a homeowners policy.

When viewing the location risk object on **Product Definition** screens, the **Details** tab on the **Risk Object Definition** popup includes a **Location/Jurisdiction** field. For a location risk object, this field is set to **Risk includes location reference** by default. The other values in the drop-down list for this field are: **Risk is at parent's location** and **Risk includes location reference**.

On the **Fields** tab, a location is created by default. The **Label** is the **Name** of risk object appended with **Address**. The **Type** is **Location** and cannot be changed. The location cannot be removed.

In a submission or other policy transaction, you can add locations on the **Risk Objects** tab.

Create risk object types in visualized product

You can add new risk object types, including location risk objects, to a visualized product.

Before you begin

- Set **Product Design Mode** to **Designer** or above.

About this task

Risk objects can be attached to other risk objects, including the product line, which is a special type of risk object.

Note: If you create risk objects directly in the visualized product, Guidewire recommends that you add these objects manually to the mind map. This ensures that the mind map is keep in sync with the product in PolicyCenter.

Rather than create risk object directly in the visualized product, you can create risk objects in the mind map and update the product from the mind map.

These instructions add a risk object from the **Product Definition** screen. You can also add risk objects in a submission or other policy transaction that displays the **Risk Details** screen.

Procedure

1. Select a product at the top of the **APD Managed** tab.
The selected product appears in the **Details** tab.
2. Click **Edit Product**.
PolicyCenter displays the **Product Definition** screen.
3. Navigate to the **Children** tab of the risk object onto which you wish to add the new risk object.
If you are adding the risk object to the product line, the **Children** tab is on the **Product Definition** screen. Otherwise, it is on the **Risk Object Definition** popup.
If the **Children** tab is not visible on the **Product Definition** screen, go to the **Details** tab, and select **Has Child Risks**. On the **Risk Object Definition** popup, select any value in **Type of Risk Object** except **Liability Only - Single Risk** and select **Has Child Risks**.
4. Select **Add**.
A **<New>** row is added to the table.
5. Click **<New>** to display the **Risk Object Definition** popup.
You can define new categories, coverages, terms, conditions, exclusions and exclusion and condition categories.
Note: You cannot reuse attributes defined for other risk objects because this is a different risk object type.

Create location risk object

6. (Optional) To create a location risk object, set **Location/Jurisdiction** to **Risk includes a location reference** on the **Details** tab.

This creates a **Location** field on the **Fields** tab. This field type cannot be changed or removed.

Update mind map

7. In the mind map for this product, add the new risk object and any other new coverages, conditions, and other objects.

You must update the mind map in XMind. Advanced Product Designer does not have an automated way to update the mind map.

Risk Object Definition popup

You can access risk objects from the **Children** tab on the **Product Definition** screen. The **Risk Object Definition** screen has its own **Children** tab if the risk object can contain child risk objects.

The table on the **Children** tab lists the names of the risk objects defined for this product or risk object. Click the **Name** of the risk object to modify its properties on the **Risk Object Definition** popup. This popup has **Details**, **Fields**, **Exposures**, **Coverages**, **Exclusions**, and **Conditions**, and **Children** tabs.

Field	Description
Name	Name of the risk object.
Description (*PD)	Description of the field. This field is required for product generation and displays if Product Design Mode is Developer .
Number Name	Whether the risks are numbered.
Type of Risk Object	Select one of the risk types. This field is for your information only. It has no effect on the product definition.
Location/Jurisdiction	Specify the relationship between the risk object and the location or jurisdiction.
Has Child Risks	If Yes , specify a user interface label that groups the risks in a policy in Children's UI Label . The Children's UI Label is only used in submissions and other policy transactions in the visualized product. It is not used in the generated product.
Multiple Exposures for Name Risk	Specify whether this risk object has exposures. If Yes , specify a user interface label that groups the exposures in a policy in Exposure's UI Label . The Exposure's UI Label is only used in submissions and other policy transactions in the visualized product. It is not used in the generated product.
Menu Label for Generated UI (*PD)	The label that appears in the user interface in the generated product. It appears as the label in the job wizard and policy file step. It does not appear in the visualized product.

Risk object Fields

On the **Fields** tab, enter metadata for all fields on the risk object.

Field	Description
Label	Label that appears in the user interface.
Type	Specify the type. If drop-down list, you can define the list.

Field	Description
Identifier	The identifier helps to identify one risk object from another in a list of risk objects. The Risk Details screen in a policy transaction lists the risks on the policy. In the list of risk objects (Risk Name), the identifier appears after name of the risk object. For example, on Pet risk objects, the Pet Name text field is the identifier. In a policy with pets named Fido and Peony, Risk Name shows the two risks: • Pet: Fido • Pet: Peony Identifier fields directly on a policy line do not appear in a policy transaction.
Field Usage	Specify whether a product attribute is derived, hidden, unavailable, or user entered by default. You can also specify conditional exceptions that change usage and availability for special circumstances.
Default Value	Default value for the field.
Minimum Value	Minimum value for the field.
Maximum Value	Maximum value for the field.
Tags	Add a tag to a product attribute to mark it for special processing or handling in the generated product or external applications.
Split By Rating Periods	This field can have split rating periods. Split rating periods create separate rating periods for dates selected by the user.
Scalable	Whether to use basis-scalable rating. When selected, the basis is scaled automatically before and after the split date. You can also manually scale the basis by editing the field values. This field is enabled if you select Split By Rating Periods , and Type is Number , Decimal (14,2) , or Monetary amount .
Category	This field is not used. Do not enter a value.
Description (*PD)	Description of the field. This field is required for product generation and displays if Product Design Mode is Developer .
Short Name (*PD)	This field is required for product generation and displays if Product Design Mode is Developer . Unique name that identifies the field in the generated product. This name must be unique in the product line.
Ordering	The order of the items in the user interface, from first to last.

Splitting risk objects in visualized products

In visualized products, you can define risk objects as splittable into multiple rating periods. Risk objects are splittable if one or more of their fields can be split.

Note: You cannot define risk objects as splittable in the mind map.

Split risk object by rating period in Product Definition

About this task

In the product definition, define a risk object as split by rating period.

Procedure

1. In the **Product Definition** screen, go to the **Children** tab and click to view a risk object.
2. On the risk object, select **Fields**.
3. Select **Split By Rating Periods** on the field.
4. If the field is a number, decimal, or monetary amount, select **Scalable** if the field is scalable.

Split risk objects in a policy transaction

About this task

In a policy transaction, you split the risk object and can modify independently the values of the split fields. Any number of splits may be performed on risk objects. Different risk objects may have different numbers of splits or different split dates.

Note:

Because the **Split** button is on the risk objects tab, you can only split child risk objects of the product line and not the line itself.

Procedure

1. In a policy transactions for a policy line that has risk objects that can be split, advance to **Risk Details** and select the risk objects tab.
2. Add the risk object that can be split.
For risk objects that can be split, **Effective Date** and **Expiration Date** appear in the list view and the **Split** button appears above the list.
3. Select one or more of these risk objects to enable the **Split** button.
4. Click **Split**.

The **Split Risks** popup appears.

5. Choose the **Date** on which to split the risks and click **Split**.

An error appears if the date is outside the policy period range or if a split already exists on the given date.

After a successful split, there are two versions of the risk object. You can click to view either version of the risk object but only fields that can be split are editable in the second version.

Both versions initially have the same data, except for scalable fields which are prorated based on the split date. You can edit splittable fields independently in both versions. Fields that cannot be split are editable only in the first version. Upon save, changes made in the first version propagate to the second version.

You can **Remove** or **Split** the first version of an risk object only. Removing a risk object removes all versions. There is no way to remove a single version. There is no way to undo the split to a risk object.

Note:

The risk object popup lets you edit other product elements besides fields, such as clauses, exposures and child coverables. In the base implementation, these elements are independently editable in both versions of the risk object.

Clauses in visualized product

Coverages, conditions, and exclusions are types of clauses. You define clauses and clause terms on risk objects in the mind map. In the visualized product, you add additional information, such as:

- Availability
- Usage for clause terms
- Minimum, maximum, and default values for clause terms
- Metadata for product generation

- Pricing

In the **Product Definition** screens of a visualized product, you can edit, add, or delete clauses and clause categories by clicking **Manage Clause Categories** on the **Coverages**, **Conditions**, or **Exclusions** tab. If you make changes, remember to update the mind map.

For individual clauses, you can also add details, such as usage properties, and minimum and maximum values, that are not imported from the mind map.

In a submission or other policy transaction that displays **Risk Details**, each clause category has its own tab on the risk object. View the tab to access the clauses in the clause category.

When specifying **Clause Availability** on a clause, you can choose the following values:

- **Optional** – Clause is optional and can be added to the policy.
- **Required** – Clause is required and is automatically added to the policy. It cannot be removed.
- **Suggested** – Clause is added automatically to the policy but can be removed.
- **Unavailable** – The clause is not available and cannot be added to the policy.

Manage Clause Categories popup

You can define clause categories on the **Manage Clause Categories** popup. The categories defined for this product are listed in the table at the top of the screen. You can add or remove categories. Select a category to display its values the following fields:

Field	Description
Label	Label that appears in the user interface.
Description (*PD)	Description of the field. This field is required for product generation and displays if Product Design Mode is Developer .
Display Categories	Select Yes to display categories on the Risk Details screen as separate views on the product line.
Clauses Included	Lists the clauses that this category includes.

Coverages, Exclusions, and Conditions tabs

Describes these tabs in **Product Definition** screens.

Note: Fields required for product generation are marked (*PD).

Field	Description
Label	Label that appears in the user interface.
Terms	Click to display the Clause Terms popup where you can define terms.
Details	Click to display the Definition popup where you can provide detail such as Events Covered and Cost Items Covered .
Category	Select a clause category. You can also edit and add a new one.
Clause Availability	Specify whether a clause is optional required, suggested, or unavailable by default. You can also specify conditions.
Tags	Add a tag to a product attribute to mark it for special processing or handling in the generated product or external applications.
Schedule?	Select if this clause has a schedule.
Schedule Details	Enabled if you select Schedule .

Field	Description
Description (*PD)	Description of the field. This field is required for product generation and displays if Product Design Mode is Developer .
Ordering	The order of the items in the user interface, from first to last.

Clause terms in visualized product

Clause terms are attributes attached to coverages, exclusions, and conditions.

To view clause terms in PolicyCenter, navigate to the **Coverages**, **Exclusions**, or **Conditions** tab, and click **Terms** for a clause. You can select the following types:

- **Text**
- **True/false**
- **Number**
- **Decimal (14,2)**
- **Monetary amount**
- **Drop-down list**
- **Date-time**

You can model clause terms in a product mind map and import them into Advanced Product Designer.

Drop-down lists

In the visualized product, drop-down lists on clause terms can be of the following types:

Type	Description
Option	Numeric values in a predefined range. You can specify that the values represent count, days, hours, money, or percent. While the values must be numeric, the Label does not have to be. For example, a deductible or limit where you can choose: • 100 • 500 • 1000
Package	Collection of numeric values, such as limits or deductibles, selected as a group from a list of predefined choices. You can specify that the values represent count, days, hours, money, or percent. While the values must be numeric, the Label does not have to be. For example, an automobile liability package covering bodily injury per claimant, bodily injury per accident, and property damage per accident where you can choose: • 10000/20000/50000 • 10000/20000/100000 • 25000/50000/100000
Typelist	Values are defined in a typelist. Use typelists if the terms are text, and option and package cannot be used.

Configuration

In Advanced Product Designer, option types are defined in the `CovTermModelVal` typelist that you can edit in Studio. You can add values to this typelist, but they must be integers. For example, you can add values for number of months and weeks.

Option clause terms in visualized product

Option clause terms let you choose from a predefined list of options. Option terms can have labels like 100, 500, or Statutory. Each option term has a numeric value. In the visualized product you can set the return value for each term. The term labeled 100 might return 100 or a multiplier of 100. The Statutory term also returns a numeric value.

To view option clause terms in a visualized product, go the coverage, condition, or exclusion, and select **Terms** to display the **Clause Terms** popup. The terms are listed in a table. Select **Definition** in a clause term to view the **Drop-Down Definition** popup. The option clause terms have a **Value** column. The label **Value** is editable. Change **Value** to an appropriate short name for the clause term options.

Drop-Down Definition [Return to Clause Terms](#)

Label Decimal Options

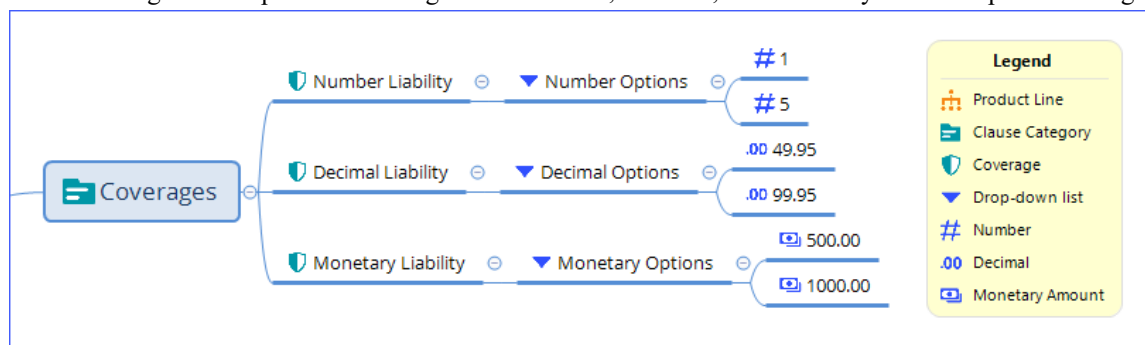
Drop-Down Type *

[Add](#) [Remove](#)

Label *	Availability	Tags	Currency	Value	(*PD)	Description (*PD)
Other						
☐ 49.95	Available	<none>	<none>	49.95	49.95	49.95
☐ 99.95	Available	<none>	<none>	99.95	99.95	99.95

Mind map

The following mind map shows coverages with Number, Decimal, and Monetary Amount option coverage terms.



See also

- “Defining option clause terms in mind map” on page 31

Visualized product

When you import or update the mind map into PolicyCenter, the Money, Number, and Decimal list items are imported as option clause terms.

If you select **Default Additional Fields** when you import or update the mind map, the topic text of option coverage terms becomes the **Value** and **Description**. For example, 49.95 and 99.95 become the **Value** and **Description** of the drop-down list items. If the marker type is Money in the mind map, **Currency** is set to the default currency.

In the **Drop-Down Definition** popup, the **Drop-Down Type** is **Option**. If the list item marker type is Money in the mind map, the clause term **Value** defaults to **Money** type in the visualized product. If the marker type is Number or Decimal, the clause term **Value** defaults to **Other**. You can change the **Value** to a more appropriate type such as **Count**, **Days**, **Hours**, **Money**, or **Percent**.

Changing Text markers to option clause terms

For option clause terms with text labels, Guidewire recommends that you use the marker appropriate for the value, such as Number, Decimal, or Monetary Amount. Text items do not import by default to option clause terms in the visualized product. However, if the mind map has Text markers for the option clause terms, you can change them to option clause terms in the visualized product.

If the marker type is Text in the mind map, then the drop-down list defaults to a typelist in the visualized product. Change this to an option clause term on the **Drop-Down Definition** popup by changing **Drop-Down Type** from **Typelist** to **Option**. The label **Value** is editable. Change **Value** to an appropriate short name for the clause term options.

Package clause terms in visualized product

Package clause terms are terms offered as a group. Each term in the package returns a numeric value. In the visualized product, you can set the numeric value for each term. For example, a Liability - Bodily Injury and Property Damage coverage on a car offers package coverage terms, that group three terms. The first package offers 50, 100, 50 terms. The term labeled 50 might return 50 or a multiplier of 50. Like the other package terms, the Statutory package term corresponds to three terms.

- 50/100/50
- 50/200/50
- Statutory

In a clause term, the package terms can have different numbers of values. For example, the following clause term has packages with three, two, and one terms:

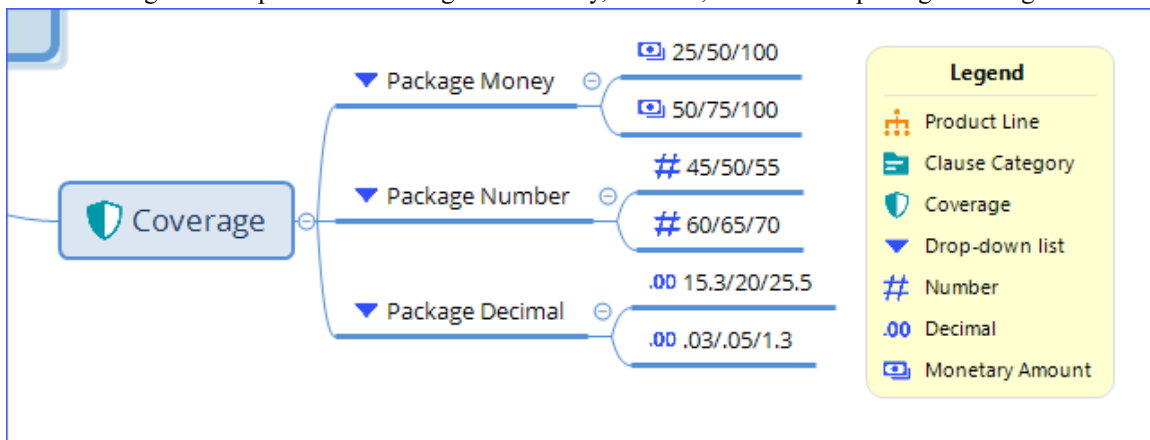
- 50/100/50
- 1000/1000
- 100K CSL

To view package clause terms in a visualized product, go the coverage, condition, or exclusion and select **Terms** to display the **Clause Terms** popup. The terms are listed in a table. Select **Definition** in a clause term to view the **Drop-Down Definition** popup. The package clause terms are split into individual **Money** or other **Other** fields. If the fields are **Other**, you can change them to a more appropriate type such as **Count**, **Days**, **Hours**, **Money**, or **Percent**. The label **Value#** is editable. Change **Value#** to an appropriate short name for the clause term.

Currency	Value	(*PD)	Value1	(*PD)	Value2	(*PD)	Description (*PD)
	Money		Money		Money		
>> USD	25		50		100		25/50/100
>> USD	50		75		100		50/75/100

Mind map

The following mind map shows a coverage with Money, Number, and Decimal package coverage terms.



See also

- “Defining package clause terms in mind map” on page 31

Visualized product

When you import or update the mind map into PolicyCenter, these drop-down lists are imported as package clause terms. In the **Drop-Down Definition** popup, the **Drop-Down Type** is **Package**. If the marker type is Money in the mind map, the clause terms default to Money type. If the marker type is Number or Decimal, the clause terms default to Other.

If you select **Default Additional Fields** when you import or update the mind map, the values are extracted. For example, the values 25, 50, and 100 are extracted from 25/50/100. If the marker type is Money in the mind map, **Currency** is set to the default currency.

In the **Drop-Down Definition** popup, the **Drop-Down Type** is **Package**. If the list item marker type is Money in the mind map, the clause term **Value** defaults to **Money** type in the visualized product. If the marker type is Number or Decimal, the clause term **Value** defaults to **Other**. You can change the **Value** to a more appropriate type such as **Count, Days, Hours, Money, or Percent**.

Changing Text markers to package clause terms

For package clause terms with text labels, Guidewire recommends that you use the marker appropriate for the value, such as Number, Decimal, or Monetary Amount.

If the marker type is Text in the mind map, the **Drop-Down Type** defaults to **Package** in the visualized product. Set the **Value** to the appropriate type and enter values. The label **Value#** is editable. Change **Value#** to an appropriate short name for the clause term.

Events on coverages

Events are incidents for which a coverage provides protection.

To view events in PolicyCenter, navigate to the **Coverages** tab, and click **Details** for a coverage.

On the **Definition** popup, events are listed under **Events Covered**. Each event has the following fields:

Field	Description
Events	Select the name of the event.
Fixed Limit	Enter a number or decimal for the fixed amount of the limit.
Fixed Deductible	Enter a number or decimal for the fixed amount of the deductible.
Limit Attribute	Select the attribute that stores the limit.
Deductible Attribute	Select the attribute that stores the deductible.
Limit Override	Select the attribute that stores the override of the limit.
Deductible Override	Select the attribute that stores the override of the deductible.

The choices available for **Limit Attribute**, **Deductible Attribute**, **Limit Override**, and **Deductible Override** fields are Monetary Amount attributes on siblings or ancestors of the event.

Predefined events

There are a number of predefined events that are available to all products. The case-sensitive names are:

- Animal Bite
- Animal Damage
- Collision
- Damage by Third Party
- Earthquake
- Fire
- Flood
- Hail
- Landslip
- Lightning
- Loss of Keys
- Mallorca Protection
- Own Property Liability
- Riot and Strike
- Rodent Damage
- Sonic Boom
- Storm
- Tap Water
- Theft
- TP Bodily Injury
- TP Property Damage
- TP Vehicle Damage
- Vandalism/Malicious Damage
- Windscreen Damage

- Landslip
- Special Equipment Damage

Adding events

If you import a new event from a mind map, the event is added to PolicyCenter. The name of the event is case-sensitive and must be unique.

The predefined events are listed in the `APD_Perils` system table. These events are accessible to all products. You can directly add or remove values from this system table in Studio.

Covered costs in visualized product

You define covered costs on clauses. The insured recoups covered costs through claims.

To view covered costs in the visualized product, navigate to the **Coverages** tab, and click **Details** for a coverage.

On the **Definition** popup, covered costs are listed under **Cost Items Covered**. Each covered cost has the following fields:

Field	Description
Category	Select the name of the covered cost.
Type	
Restricted To	

Predefined covered costs

There are a number of predefined covered costs that are available to all products. The case-sensitive names are:

- Age/Disabled Equipment
- Loss of Earnings
- Rental Vehicle
- Alternate Accommodation
- Loss of Rent
- Site Clearance
- Car Valet
- Medical
- Vehicle Recovery
- Compensation (general)
- Property Repairs
- Vehicle Repairs
- Legal
- Property Replacement
- Vehicle Replacement
- Locks and Keys
- Rental Vehicle
- Windscreen

Adding new covered costs

If you import a new covered cost from a mind map, the covered cost is added to PolicyCenter.

The predefined covered costs are listed in the `APD_ClaimCostCategory` system table. These covered costs are accessible to all products. You can directly add or remove values from this system table in Studio.

Subclauses in visualized product

You can create subclauses, or clauses with parent-child relationships, directly in Advanced Product Designer or in a mind map that you import into Advanced Product Designer. For example, a coverage can have child coverages.

- A clause can have zero or one parent clause
- A clause can have multiple subclauses
- You can create multiple levels of subclauses
- A subclause and its parent must be of the same clause type.

For example, an exclusion cannot be a subclause of a coverage. PolicyCenter displays a validation error if it encounters this when importing a mind map.

For example, a Personal Injury Protection coverage can have PIP Additional Benefits and PIP Medical Expense Limit child coverages. These child coverages appear as **Subclauses** on the Personal Injury Protection coverage. The PIP Additional Benefits coverage has a Work Loss child coverage.

Numeric option clause terms with calculated values

On parent clauses, terms of type number, decimal, or monetary amount, the minimum and maximum values can be fixed or calculated. If fixed, you specify the value. If calculated, the value can be the minimum, maximum, or sum of specified subclause terms of the same type underneath the parent clause.

On parent clauses, the minimum and maximum values for option clause terms can only be calculated. You must specify how the value is calculated.

See also

- “Subclauses with numeric and option clause terms” on page 74
- “Add minimum or maximum value to numeric or option clause term” on page 75

Generated code

Subclause relationships are not part of the generated product. You must import these relationships in editions.

Create subclause in visualized product

About this task

Create a child clause of a parent clause by creating a subclause. The child clause must be of the same type (coverage, condition, or exclusion) as its parent.

Procedure

1. On the **Product Management** screen, select a product, and click **Edit Product** on the **Details** tab.
2. Navigate to the **Coverages**, **Exclusions**, or **Conditions** tab.
3. Click **Subclauses** on the clause where you wish to add a subclause.

The **Subclauses** popup appears. In this popup, you can:

- Add a new subclause or select a subclause from existing clauses of the same type on that risk object
- Unassign a subclause from its parent clause
- Remove a subclause

Add a new or existing clause as a subclause

4. To add a new subclause, select **Add→New *Clause*** to add a new subclause, where *Clause* is **Coverage**, **Condition**, or **Exclusion**.

The popup is in the context of the parent clause. It is automatically assigned to the same clause category, and you cannot change its category. The new clause created in this popup is linked to the parent clause as its child clause.

5. To add an existing clause as a subclause, select **Add→Existing *Clauses***, where *Clauses* is **Coverage**, **Condition**, or **Exclusion**, and select the clause from the list.

The list includes all clauses of the same clause type on that risk object. The list excludes the current clause, its ancestors, and subclauses.

Existing *Clauses* is unavailable if there are no clauses that can be subclauses.

Note: If this results in a clause being assigned a new parent, its category is automatically changed to match the category of the new parent.

Unassign a subclause from its parent clause

6. On the **Subclauses** popup, select a clause, and click **Unassign**.

When you unassign a subclause from its parent, the subclause moves to the top ancestor level. For example, if you unassign a doubly nested subcoverage, it appears on the **Coverages** under which it was nested.

Remove a subclause

7. On the **Subclauses** popup, select a clause and click **Remove**.

This removes the clause and the subclauses it defines from the product.

Subclauses in a visualized policy transaction

In policy transactions based on the visualized product, in user mode, the **Subclauses** button is available if a clause has subclauses. In the policy transaction, advance to a clause that has subclauses.

- Click **Subclauses** to display the **Subclauses** popup. This popup displays the subclauses that are children of the current clause.
- If one of the children has its own child clauses, the **Subclauses** button appears for that child. Click **Subclauses** to display the **Subclauses** popup for the next level of subclauses.

In policy transactions based on the visualized product, in designer or developer mode, the **Subclauses** button is always available so that you can define subclauses on the current clause.

Schedules on clauses

In the XMind mind map and in Advanced Product Designer, you can create schedules on clauses (coverages, conditions, and exclusions).

In Advanced Product Designer, schedules appear under the **Clause Schedule Terms** listview on the **Clause Terms** popup. Access this popup by clicking **Terms** in the clause. You can add scheduled item properties to the schedule.

Schedules capture information about scheduled items. For each scheduled item, each piece of information is a *schedule term*, such as a name or description. Schedules are often referred to during underwriting and are included in forms.

Some schedule terms are passed to rating and potentially affect the cost of the policy. These terms are often, but not necessarily, Decimal, Monetary Amount, Option, or Package schedule terms.

If a schedule term is modeled in the mind map as a drop-down list with Number, Decimal or Monetary Amount items, then the list becomes an Option drop-down type in the visualized product. If the value contains the slash character (/), it is imported as a Package drop-down type. In the visualized product, you can change the drop-down type to Typelist or another value.

View or add schedule in visualized product

After you import a mind map into Advanced Product Designer, you can view schedules in **Product Definition** and submissions.

Before you begin

These instructions reference the mind map in “Schedules in mind map” on page 33.

About this task

You can view and add a schedule to a clause (coverage, condition, or exclusion).

For example, a Jewelry Scheduled Item has the following schedule terms:

- Type – Type of jewelry as Text
- Description – Description of the piece of jewelry as Text
- Value – Value of the jewelry as a Monetary Amount
- Deductible – Drop-down list with \$500, \$1,000, and \$5,000 Monetary Amount choices

In a policy, the Jewelry schedule contains the following information:

Type	Description	Value	Deductible
Pearl necklace	Single strand, 16 inches, white pearls	\$4,500	\$500

Type	Description	Value	Deductible
Diamond engagement ring	3 carats of diamonds in 18 karat white gold	\$17,000	\$5,000
Wedding band	18 karat white gold	\$7,250	\$1,000

Procedure

View or add

1. View the product in the **Details** tab of the **Product Management** screen.
2. Edit the product to view the **Product Definition** screen and go to the **Clause** tab. In this example, the schedule is on the Jewelry coverage, so go to the **Coverages** tab.
3. In the clause that has or will have a schedule, click **Terms**.
The **Clause Terms** popup appears. At the bottom of the screen, the **Clause Schedule Terms** listview displays the schedule terms: **Type**, **Description**, **Value**, and **Deductible**.
4. Optionally, click **Add** to add new schedule terms.

Visualize in a submission

5. To see the schedule in visualize mode, create a submission with this product and go to the **Risk Details** screen.
6. Go to the clause category that contains the schedule. In this example, go to the **Valuables** tab.
The Jewelry schedule appears in the **Clause Schedule** listview. You can add scheduled items to the policy.
7. If you are in **Designer** mode or higher, you can modify the schedule definition by adding, removing, or changing the schedule terms.

Schedule term configuration

Terms on linked clause

Some schedule terms are passed to rating and potentially affect the cost of the policy. In the **Clause Schedule Terms** listview, mark these schedule terms as **Generate As Clause Term**. When you import or create a schedule with Decimal, Monetary Amount, Option, and Package schedule terms, **Generate As Clause Term** is always **Yes**.

Location and Involved party schedule terms cannot be generated as clause terms.

Generate As Clause Term is visible when you are in **Developer** mode.

When you import a template that has linked clauses, the linked clause terms of any field type are marked as **Generate As Clause Term**.

For example, the jewelry schedule has Value monetary amount and Deductible monetary amount option schedule terms provide information that is used to rate the policy. It also has Type and Description text schedule terms.

When **Generate As Clause Term** is selected, Value and Deductible become clause terms for a linked clause that is generated under the schedule coverable (Jewelry). Type and Description become text properties on the Jewelry coverable.

Generic schedules

Schedules in the visualized product become generic schedules in the generated product.

Attributes

When writing a policy, you may require specific information or answers to question about parts of the policy. You can model this information using attributes. For example, in a small business policy, you need to know whether the insured has been refused insurance in the past. On a building, you may need to know the number of stories. Attributes can be attached to various parts of the product, such as the product line, risk objects, and exposures. Attributes can impact the pricing of a policy and drive underwriting decisions.

In a visualized product, attributes associated with a product model element appear below or near that element. Attributes can be:

- Text
- Monetary amount
- Drop-down list
- Number
- True/false
- Location
- Decimal
- Date-time
- Involved party

Attributes on the product line and risk objects appear on the **Fields** tab of the **Product Definition** screen. In a submission on the **Risk Details** screen, these attributes appear on the **Fields** tab.

Drop-down lists in visualized product

Drop-down lists let you choose an option from a predefined list of choices. You can add drop-down lists to risk objects and exposures. You can also add drop-down lists to clauses (coverages, conditions, and exclusions), to create option and package clause terms.

In Advanced Product Designer, you can:

- Define new drop-down lists in visualized products.
- Add, remove, and modify drop-down list items.
- Use drop-down lists defined elsewhere in the current product. Select **Owning Attribute of shared code lists** on the **Drop-Down Definition** popup.
- In **Developer** mode, access drop-down lists defined in PolicyCenter.

Import from mind map

You can model drop-down lists in a product mind map and import them into Advanced Product Designer. In Advanced Product Designer, you can create and modify drop-down lists.

When you import a mind map with drop-down lists on risk objects, the list items become text in the visualized product regardless of whether they were modeled as Decimal, Number, or Money. This includes drop-down lists on the product line.

See also

- “Drop-down lists in mind map” on page 35

Clause terms

You can use drop-down lists to model clause terms in the visualized product.

When you import a mind map with clause terms, Decimal, Number, and Money values can be preserved.

See also

- “Clause terms in visualized product” on page 57

Drop-Down Definition

Drop-down lists are fields with **Type** set to **Drop-down list**. To define and modify the list definition, click **Definition** to display the **Drop-Down Definition** popup. In **Designer** mode, this popup has the following fields:

Field	Description
Owning Attribute of shared code lists	If you are defining the drop-down list here, set to <none> . To use a drop-down list defined elsewhere in the product, select one of the available lists. Displays all drop-down lists defined in the current product. The lists are identified by Label and Typelist (*PD) separated by a colon (:). If a list does not have Typelist (*PD) defined, it does not appear as a choice.
Add and Remove	Add or remove values from the list.

Drop-down lists in developer mode

When you generate the product, drop-down lists defined in the product become *typelists* in PolicyCenter.

Developer mode gives you added abilities with drop-down lists. For each drop-down list, you must specify the typelist name in the **Typelist (*PD)** field. In addition, you can specify that a drop-down list uses a typelist already defined in PolicyCenter.

PolicyCenter includes general purpose typelists such as Country, Jurisdiction, and Currency. PolicyCenter also includes typelists defined in generated products.

IMPORTANT **Developer** mode gives you the ability to make system-wide changes to typelists. Be careful when making changes. Your changes affect visualized and generated products and PolicyCenter features that use these lists.

Drop-Down Definition

In **Developer** mode, the **Drop-Down Definition** popup for a drop-down list includes **Typelist (*PD)** and **Typelist Rebuild State** fields.

Field	Description
Owning Attribute of shared code lists	If you are defining the drop-down list here, set to <none>. To use a drop-down list defined elsewhere in the product, select one of the available lists. Displays all drop-down lists defined in the current product. The lists are identified by Label and Typelist (*PD) separated by a colon (:). If a list does not have Typelist (*PD) defined, it does not appear as a choice.
Typelist (*PD)	The short name of the typelist that will be generated into the product. This field appears if Owning Attribute of shared code lists is <none>. To use a typelist already defined in PolicyCenter, enter the name of the typelist. The Sync button appears. Press Sync to synchronize and retrieve the typelist values.
Typelist Rebuild State	Rebuild – The typelist is rebuilt when you generate the product. Rebuild is the default if you make no selection. Do Not Rebuild – Changes that you make to the typelist are not generated into the product. If this product owns or maintains the typelist, then you want to rebuild changes into the typelist. If the typelist is owned or maintained by another product or by the application, then you do not want to rebuild. This field is visible when Owning Attribute of shared code lists is <none>.
Add and Remove	Add or remove values from the typelist.

Whether to rebuild typelists

In **Developer** mode, you can specify whether or not changes to drop-down lists are saved to the corresponding typelist in the generated product. You specify this in **Typelist Rebuild State** on the **Drop-Down Definition** popup.

In Advanced Product Designer, you can add or delete items from a typelist. If you choose to rebuild the typelist, when you generate the product, the typelist is rebuilt and reflects the changes you made. Therefore, your changes affect all visualized and generated products and PolicyCenter features that use this typelist.

For each typelist, determine where it will be *owned* and *maintained*, and have a single owner. Then set the rebuild state in accordance with your decision. Rebuild is the default if you make no selection.

Rebuild

If this product owns or maintains the typelist, then you want to rebuild the typelist, and update the system with your changes. Set **Typelist Rebuild State** to **Rebuild**.

A drop-down list *owns* and *maintains* the typelist if it defines the drop-down list. For example, a cell phone insurance product defines a Brand drop-down list. You want the product to own the drop-down list and be responsible for its contents. Therefore, you want to rebuild the typelist when the product is generated. You can set **Typelist Rebuild State** to **Rebuild**, or not specify a value since the default is to rebuild.

A drop-down list within a product *maintains* the typelist if the product is responsible for making changes to the typelist. For example, you create a drop-down list and sync it to a typelist in PolicyCenter. For business reasons, you decide that the product maintains this list. In this case, you want to rebuild the typelist when you generate the product.

Do not rebuild

If the typelist is *owned* or *maintained* by another product or by the application, do not rebuild. Set **Typelist Rebuild State** to **Do Not Rebuild**.

For example, you create a drop-down list and sync it to the Jurisdiction typelist. This typelist is used by various products and features in PolicyCenter. Therefore, you have decided that system administrators maintain this typelist. Drop-down lists within products can sync to this typelist, but you do not want them to make any changes. Therefore, you set **Typelist Rebuild State** to **Do Not Rebuild**.

Linking drop-down lists that reference other lists

In a mind map, drop-down lists can reference drop-down lists defined in the current product or reference drop-down lists that are typelists in PolicyCenter.

After importing a mind map, drop-down lists that get their values from other lists must be linked to those lists.

Set owning attribute for drop-down list

In a mind map, a drop-down list references a list that is defined by another attribute on the product. In the visualized product, set the owning attribute.

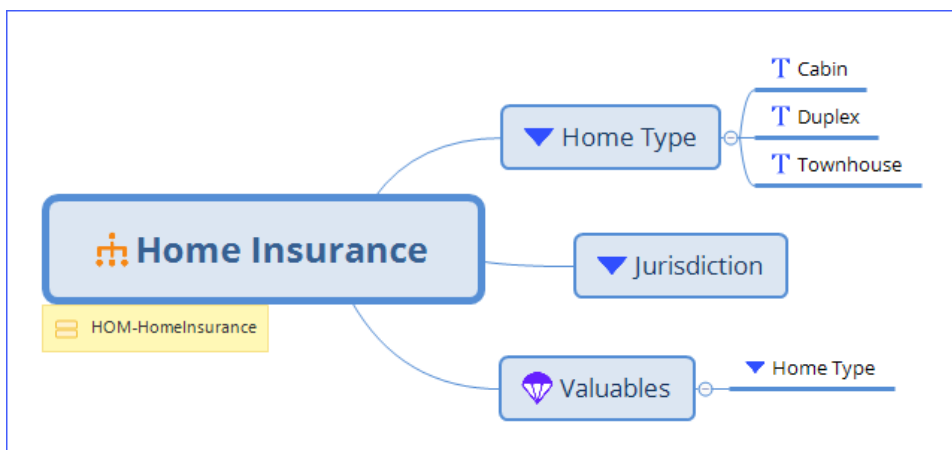
Before you begin

Set **Product Design Mode** to **Developer**.

About this task

If a list in the mind map references another drop-down list defined in the current product, the list is created in the visualized product, but has no list items. In the visualized product, the list has the same name as in the mind map. To establish this relationship to the other *owning* drop-down list, you must set the owning attribute in the **Drop-Down Definition** screen.

For example, in the following mind map, the Home Type drop-down list is defined on the Home Insurance product line. The Valuables risk object includes a drop-down list that references this list.



Procedure

1. In **Product Definition**, go to the **Fields** tab that contains the drop-down list.
In this example, go to the **Fields** tab on the Valuables risk object.

2. In the drop-down list, click **Definition** to display the **Drop-Down Definition** popup.
3. In **Owning Attribute of shared code lists**, select the name of the owning list.
Select **Home Type : HomeType**.
The list items appear below.

Link drop-down list to PolicyCenter typelist

In the mind map, you created a drop-down list that gets its values from a PolicyCenter typelist. In the visualized product, link the drop-down list to the typelist.

Before you begin

Set **Product Design Mode** to **Developer**.

About this task

If a drop-down list in the mind map references a typelist in PolicyCenter, the typelist name is prepended with the product line abbreviation in the visualized product. For example, you have a mind map where the product line label is HOM-HomeInsurance (<Abbreviation>-<Identifier>). This product includes a Jurisdiction drop-down list. In the visualized product, the typelist name becomes HOMJurisdiction. To link this list to the typelist, you must update the drop-down list.

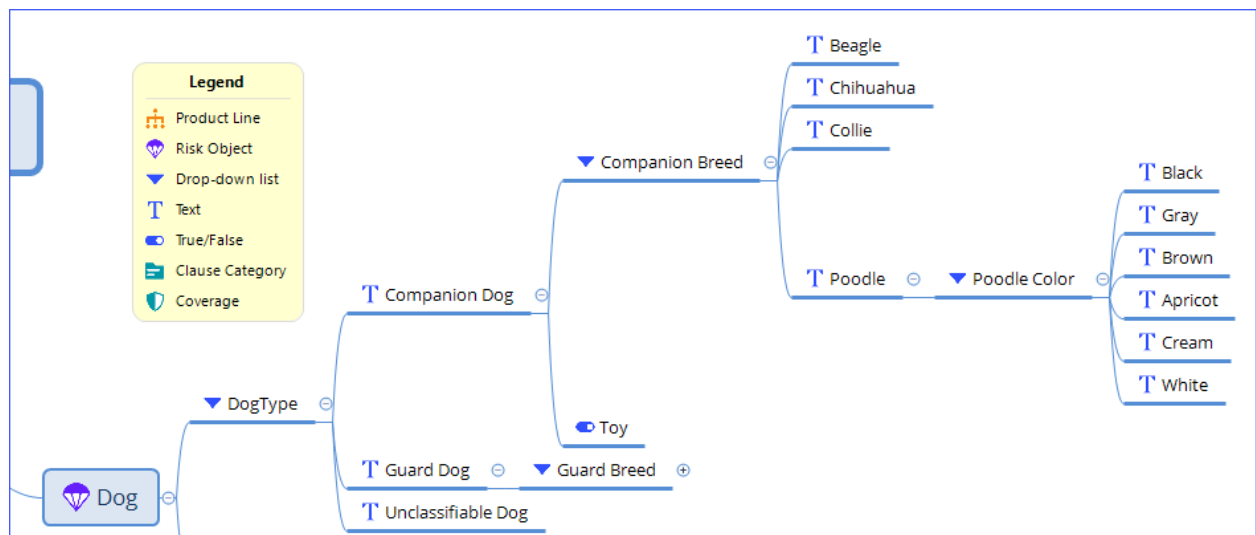
Procedure

1. In **Product Definition** for the product, go to the **Fields** tab that contains the drop-down list.
2. In the drop-down list, click **Definition** to display the **Drop-Down Definition** popup.
In this example, the drop-down list is Jurisdiction.
3. In **Typelist (*PD)**, remove the abbreviation from the name.
For example, remove HOM from HOMJurisdiction.
If the name of the typelist matches one in PolicyCenter, the **Sync** button appears.
4. Click **Sync** to retrieve the typelist values.

Modeling dependent relationships in drop-down lists

Mind map

You can model dependent relationships in mind maps by attaching attributes to drop-down list items. For example, in a Pet Insurance product, the Dog risk object has a Type drop-down list where you can specify whether the dog is a Companion Dog, Guard Dog, or Unclassifiable. If you select Companion Dog, you can further specify the breed and whether the dog is a toy. The breed is a drop-down list, and Poodle includes a nested Poodle Color drop-down list. Companion Dog also has a Boolean Toy attribute. You cannot attach attributes to the Toy attribute.



A drop-down list can have Text, Number, Money, and Decimal attributes. Beneath any of these list items, you can attach data type attributes, such as Text, Drop-down list, and Number, among others. If you attach a drop-down list, you can continue attaching attributes on the list items. This relationship is preserved when you import the product to PolicyCenter.

In schedules, dependent attribute relationships in drop-down lists are not supported. They are dropped when you import the mind map.

Visualized product

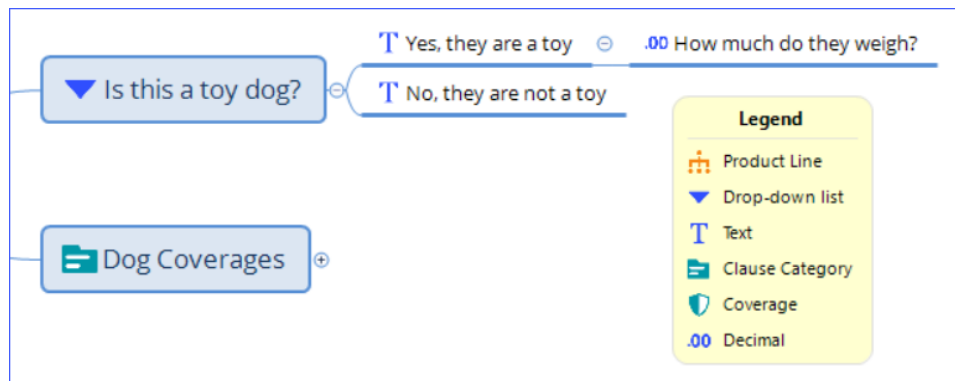
When you import the mind map into PolicyCenter, go to the **Fields** tab for the risk object to see the drop-down lists and attached attributes. For these attributes, **Field Usage** is always **Unavailable** with conditional exceptions that make the attribute available when the parent list item is selected.

For Pet Insurance, go to the **Fields** tab for the **Dog** risk object. The top-level **Type** drop-down list imports with a **Field Usage** of **User Entered**. The **Companion Breed** and **Toy** attributes have **Field Usage** of **Unavailable**. For **Companion Breed**, click **Unavailable** to see that this drop-down list has a conditional expression that makes it available when **Type** is **Companion Dog**.

Use drop-down list for follow-up to true/false questions in visualized product

You can model follow-up questions to a True/False (binary choice) question in mind maps by using a Drop-down list. This is because True/False topics cannot have subtopics. They are deleted when you import the mind map, and you cannot model this relationship in Advanced Product Designer.

The following example has a binary choice question, *Is this a toy dog?*, that is modeled as a drop-down list. The *Yes, they are a toy* choice has a follow-up question *How much do they weigh?*



Yes, they are a toy, *No, they are not a toy* are text. *How much do they weigh?* is decimal.

In the visualized product, *Yes, they are a toy* and *No, they are not a toy* become drop-down list entries, which are answers to the *Is this a toy dog?* attribute.

When you import the mind map into PolicyCenter, go to the **Fields** tab to see the drop-down list and attached attribute.

On the **Fields** tab, the top-level **Is this a toy dog?** drop-down list imports with a **Field Usage** of **User Entered**. The **How much do they weigh?** attribute has a **Field Usage** of **Unavailable**. Click **Unavailable** to see that this drop-down list has a conditional expression that makes it available when the answer to **Is this a toy dog?** is **Yes, they are a toy**.

See also

- “Use drop-down list for follow-up to True/False questions in mind map” on page 36

Involved party in visualized product

An involved party is a contact (person or company) who has an interest on the policy. In the product model, you can attach involved parties to the policy line, risk objects, and exposures.

Note: If you attach involved parties to objects others than policy line, risk objects, or exposures, they are imported as children of the nearest ancestor that is a policy line, risk object, or exposure.

In a submission

Adding an involved party in a submission is just like adding any other contact to the policy. From the drop-down list, select a contact **From Address Book**, **New Person**, **New Company**, or **Other Contacts**. Contact details include name, phone number, address, and other information. In the **Contacts** tab of a bound policy, these contacts have the Involved Party role.

Product model properties

You can use product model properties to specify how to handle product attributes. The product attribute can be a field, clause, clause term, or drop-down list entry. You can use properties to specify the following types of information for a product attribute:

- Default values
- Minimum and maximum values
- Usage properties to specify how a product attribute is handled or obtained
- Tags to mark a product attribute for special processing or handling

With a few exceptions, you can see the effect of product model properties in a policy transaction in a visualized product.

You can load product model properties into generated products when you load editions. See “Loading editions into generated product” on page 109.

Tags

Add a tag to a product attribute to mark it for special processing or handling in the generated product or external applications.

Note: Tags are product model properties.

You can add the following tags:

- **Rate** – This attribute applies to rating
- **Submission** – This attribute applies to submissions
- **Risk Score** – This attribute applies to the risk score for a policy

You can specify the circumstances under which the product attribute is tagged. For example, a coverage applies to rating by default, but does not apply to rating if the state is California.

Configuration

In the generated product, you can use Guidewire Studio and other development tools to do the special processing that the tag requires.

You can add your own tags by updating the APDTagType typelist.

Usage and availability in visualized product

Specify whether a product attribute is derived, hidden, unavailable, or user entered by default. You can also specify conditional exceptions that change usage and availability for special circumstances.

In the visualized product, you can specify usage and availability information for product attributes. Except as noted, you can see the effects in a submission.

Note: In conceptualization, you may have added notes about usage and availability in the mind map.

You can use these notes to manually add usage and availability to the visualized product.

Usage and availability properties are accessed from the **Field Usage** and **Clause Availability** columns of a product part. These properties cannot be imported from the mind map.

In various parts of the product model, you can define properties that control the usage and availability of parts of the product model. You can specify default usage or availability, and you can specify conditional exceptions. For example, a coverage is available by default. The coverage has a conditional exception specifying that it is not available if the applicant has been refused insurance in the past.

Fields

For fields, you can specify that the field is:

- **Derived** – Value is derived programmatically, and is read-only in the user interface. In the generated product, create code in Guidewire Studio that derives the value.
- **Hidden** – Value is stored in the database but is not displayed to the user.
- **Unavailable** – The field is not available and cannot be added to the policy.
- **User Entered** – Value is entered by the user.
- **User Entered: Required for Bind** – Value is entered by the user and is required to bind the policy. Has no effect in a visualized submission.
- **User Entered: Required for Issue** – Value is entered by the user and is required for issuance. Has no effect in a visualized submission.
- **User Entered: Required for Quote** – Value is entered by the user and is required at quote. Has no effect in a visualized submission.

Clauses

For clauses, you can specify that the clause is:

- **Optional** – Clause is optional and can be added to the policy.
- **Required** – Clause is required and is automatically added to the policy. It cannot be removed.
- **Suggested** – Clause is added automatically to the policy but can be removed.
- **Unavailable** – The clause is not available and cannot be added to the policy.

Conditional exceptions on fields and clauses

In conditional exceptions for both fields and clauses, the attributes you can access are determined by the product model. The drop-down list includes all attributes on siblings and parents, excepting attributes for locations and involved parties.

By default, all products have attributes for Product, Base Jurisdiction, Underwriting Company, and Contract Currency.

If a field or clause has multiple conditional exceptions, the conditional exceptions are evaluated in priority order, and stops when a conditional exception is true. If no conditional exceptions match, the default value is used.

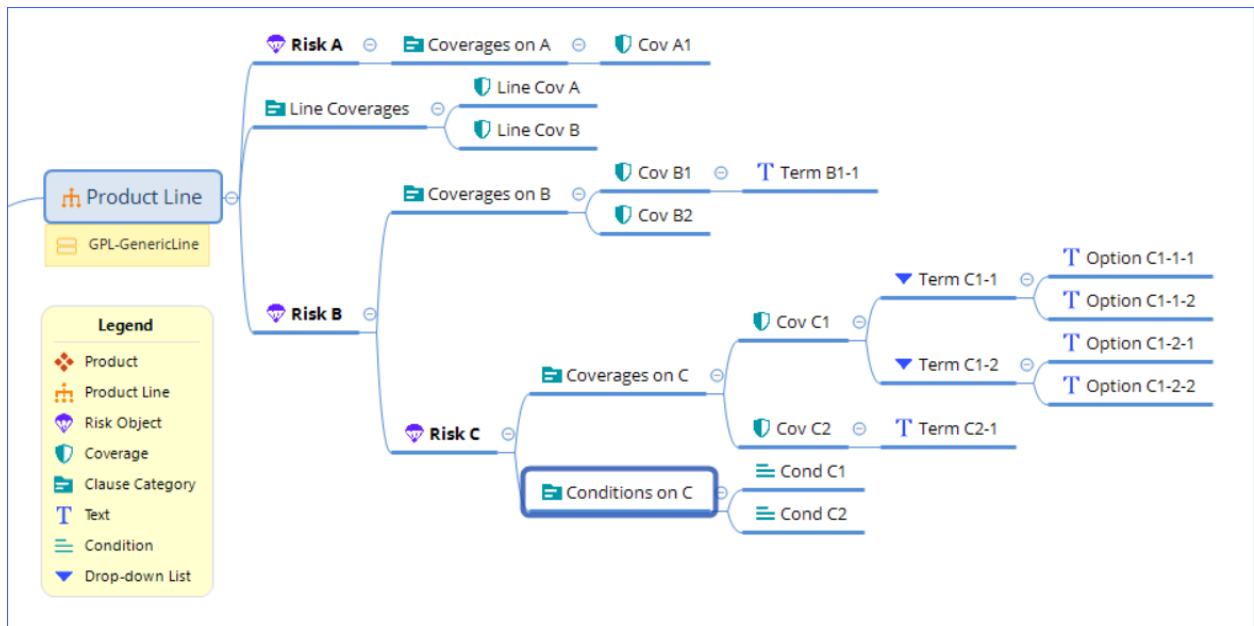
If a conditional exception has multiple rows (conditions), all conditions must be true. On the **Conditional Exceptions** table, the conditions in each exception are separated by the word **AND**.

Conditional exceptions on clauses

The conditional exception can access clauses according to where it is:

- On clauses – Choose from clauses on siblings and parents
- On clause terms – Choose from clauses and clause terms on sibling clauses and parents
- On clause term typelists, package and option terms – Choose from other clause terms on the clause

The following mind map shows a generic product structure with risk objects and clauses.



A conditional exception on the Cov C1 coverage can access the following clauses:

- Product Line: Line Cov A
- Product Line: Line Cov B
- Risk B: Cov B1
- Risk B: Cov B2
- Risk C: Cond C1
- Risk C: Cond C2
- Risk C: Cov C2

A conditional exception on Term C1-1 coverage term can access the following clauses and clause terms:

- Cov B1: Term B1-1
- Cov C1: Term C1-2
- Cov C2: Term C2-1
- Product Line: Line Cov A
- Product Line: Line Cov B
- Risk B: Cov B1
- Risk B: Cov B2
- Risk C: Cond C1
- Risk C: Cond C2
- Risk C: Cov C2

On Risk B and C, coverages and coverage terms cannot access Cov A1 because Risk A is not a parent.

A conditional exception on Option C1-1-1 option can access the following clause term:

- Cov C1: Term C1-2

Note: You cannot define conditional exceptions on schedule terms.

Specifying additional availability in Product Designer

If you have requirements for usage and availability that cannot be specified using Advanced Product Designer, you can use the availability features that Product Designer offers. For example, you can specify availability by policy transaction (job) type in Product Designer but not in Advanced Product Designer. In Product Designer you can specify availability in:

Availability scripts

Gosu scripts that can capture complex logic.

Grandfathering

Continue to offer a coverage or other pattern to existing customers, even if the pattern is not otherwise available.

Offerings

Enable or disable product model patterns for different sales or customer situations.

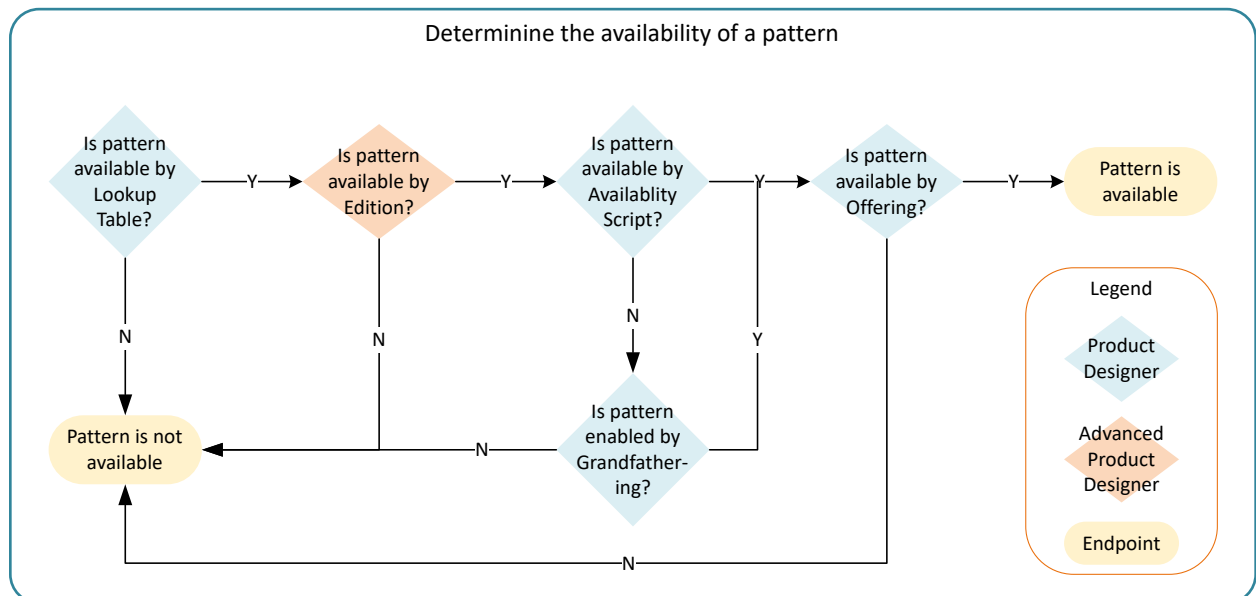
Availability lookup tables

Specify whether or not a given pattern is available based on date, jurisdiction, industry code, transaction type, underwriting company, among others.

Note: When possible, use the availability features provided in Advanced Product Designer. Only use Product Designer for availability requirements that Advanced Product Designer does not provide.

Determining the availability of a pattern

If you use a combination of availability features from Advanced Product Designer and Product Designer, availability is determined as follows:



Default, minimum, and maximum values for product model properties

You can specify default values for:

- Fields
- Exposure fields
- Clause terms
- Drop-down lists

You can specify minimum and maximum values for:

- Numeric fields and exposure fields
 - Clause terms of numeric and option type
- The default, minimum and maximum values are optional.

You cannot specify minimum and maximum values for:

- Drop-down lists on fields and exposure fields
- Clause terms of type package or typelist

Subclauses with numeric and option clause terms

When specifying minimum and maximum values for numeric and option clause terms, the value can be calculated if the current clause has child clauses (subclauses).

Restrictions on clauses and clause terms referenced in calculations

Clauses and clause terms that are referenced in calculated values of minimum and maximum values have the following restrictions:

- **Removal**
 - You cannot remove a clause if any of its terms, including terms on subclauses, is referenced in a calculated value.
 - You cannot remove a clause term that is referenced in a calculated value.
- **Change type**
 - You cannot change the type of a clause term if it is referenced in a calculated value.
 - You cannot change the type of a parent clause term if its minimum or maximum value is calculated.
- **Unassign clause**
 - You cannot unassign a clause from its parent clause if it or any of its subclauses is referenced in a calculated value.

Add default and minimum and maximum values

Before you begin

- Set **Product Design Mode** to **Designer** or above.

About this task

You can specify values on the **Product Definition** screen. Alternatively, you can enter values on **Policy Contract** screen in a submission or other policy transaction.

You can set **Default Value**, **Minimum Value**, and **Maximum Value**. You can specify **Conditional Exceptions** to handle more complex situations.

In this example, you set the default value of a field to 100. However, in California, the default is 200. The minimum value for the field is 50, but 75 in California.

The steps start a submission using the Ray Newton account defined in the sample data.

Note: Minimum and maximum values are product model properties.

Procedure

1. Select a product at the top of the **APD Managed** tab.

The selected product appears in the **Details** tab.

2. Click **Edit Product**.

PolicyCenter displays the **Product Definition** screen.

Product Definition screen

3. Select the **Fields** tab.
4. Click **Add** to add a field.
5. Set the **Label** to **Number of Objects** and **Type** to **Number**.

The **Default Value**, **Minimum Value**, and **Maximum Value** columns display **<none>** if a value has not been defined. The column is blank if a value cannot be specified.

Specify default, minimum, or maximum

6. In this row, click **<none>** in the **Default Value**, **Minimum Value**, or **Maximum Value** column.

A **Default Property**, **Minimum Property**, or **Maximum Property** popup appears.

7. Select **<none>** in the **Minimum Value** column.

The **Minimum Property** popup appears.

8. In **Minimum Possible Value**, specify a default value. Enter 100.

Add conditional exception

9. To add a conditional exception, click **Add Condition**.

The **Default Property** for popup appears.

10. Click **Add** and select **Policy:Base Jurisdiction** from the **Element** drop-down list.
11. Select or specify a **Matching Value** such as California.
12. In **Resulting Value**, enter a default value for this conditional exception, such as 200.
13. Click **OK** to save the conditional exception.
14. Add a conditional exception that sets the minimum value to 50, and to 75 for California.

Test in a submission

15. To test your default properties, switch **Product Design Mode** to **User**.
16. Start a submission for this product and navigate to the **Policy Contract** in a submission.
17. Go to the **Risk Details** screen.
18. On the **Fields** tab, enter a value of 60 in the **Number of objects** field, and click **Next**.
Because the base jurisdiction of Ray Newton's policy is California, the screen displays an error message because 60 is below the minimum of 75 for California.

Change default value in a submission

19. Switch **Product Design Mode** to **Designer** or higher, then navigate away from, then return to the **Risk Details** screen.
In the **Number of Objects** field, **Value** is set to 200, the default for California. You can change the value. The **Default Value** column displays **200**, the default for California.
20. In the **Default Value** column, click **200**.
The **Default Property** for popup appears and you can modify the default property.

Add minimum or maximum value to numeric or option clause term

Before you begin

- Set **Product Design Mode** to **Designer** or above.

About this task

On parent clauses, terms of type number, decimal, or monetary amount, the minimum and maximum values can be **Fixed** or **Calculated**. Although **Fixed** and **Calculated** appear on clauses without children, you cannot specify any **Argument(s)**. The steps for option clause terms are the same except that the value is always **Calculated**.

For option clause terms, **Conditional Exceptions** are not available.

Procedure

1. Navigate to a parent clause with numeric terms.
2. Click **Terms**.
3. On one of the terms, click **<none>** in the **Minimum Value** column.
The **Minimum Property** popup appears.
4. The **Value Type** can be either **Fixed** or **Calculated**.
5. If you select **Fixed**, enter a **Minimum Possible Value**.
6. If this clause has numeric clause terms and you select **Calculated**, select a **Function**.

Maximum The value is the maximum value of the selected **Argument(s)**.

Minimum The value is the minimum value of the selected **Argument(s)**.

Sum The value is the sum of the selected **Argument(s)**.

The **Argument(s)** field displays subclause terms of the same type underneath the parent clause.

7. In the **Argument(s)** field, select the clause terms that will be used in the calculation.

To select multiple arguments, press **Ctrl** and make your selections.

8. Optionally, add **Conditional Exceptions**.

In **Conditional Exceptions** on numeric clause terms, the value is always fixed.

You cannot add **Conditional Exceptions** to option clause terms.

Add field usage

Use field usage to specify whether a field is derived, hidden, unavailable, or user entered by default. You can also specify conditional exceptions to this property.

Before you begin

- Set **Product Design Mode** to **Designer** or above.

About this task

You can add field usage properties on the **Product Definition** screen. Alternatively, you can add properties in a submission or policy transaction that displays the **Risk Details** screen.

Field usage can include *conditional exceptions* that change usage for special circumstances. For example, the default for a **Test Field** field is user entered (**Field Usage** column is **User Entered**). A conditional exception changes the **Test Field**'s usage to unavailable if the base jurisdiction is California.

Note: Field usage is a product model property.

This example creates a submission using the Ray Newton policy in the sample data.

Procedure

1. Select a product at the top of the **APD Managed** tab.
The selected product appears in the **Details** tab.
2. Click **Edit Product**.
PolicyCenter displays the **Product Definition** screen.
3. Select the **Fields** tab.
4. Click **Add** to add a text field named **Test Field**.
5. In **Test Field**, select **User Entered** in the **Field Usage** column.
6. In **Default Usage**, select a default value from the drop-down list.
7. In **Conditional Exceptions**, click **Add Condition** to add a conditional exception.
8. Set the value of the conditional exception in **Usage Resulting Value**. You can set it to **Unavailable**.
9. Click **Add** to add a condition.
10. For example, in **Element** select **Policy:Base Jurisdiction**, specify the **Operator** as **=**, and set the **Matching Value** to **California**.

Depending upon the **Attribute** type, the **Operator** can be:

=	Equal to
!=	Not equal to
<	Less than

>	Greater than
<=	Less than or equal to
>=	Greater than or equal to

Test

11. To test your default properties, switch **Product Design Mode** to **Use**.
12. From the Ray Newton account, start a submission using this product and navigate to the **Policy Contract**.
13. Go to the **Risk Details** screen and select **Risk Objects**.
14. Select **Add→Add Group**.
Because you are in the Ray Newton policy which is in California, the **Test Field** field does not appear on **Risk Details**.

Changing field usage in Policy Contract

15. Switch **Product Design Mode** to **Design** or higher, then return to the **Risk Details** in the **Policy Contract**.
In the **Test Field** field, **Default Value** is empty and you cannot enter a value. The **Field Usage** column displays **Unavailable** because **Test Field** is unavailable in California.
16. In the **Field Usage** column for **Test Field**, click **Unavailable**.
The **Usage Property for** popup appears and you can modify the property.

Exposures in visualized product

An exposure represents a potential for accident or loss due to a risk, such as lawsuits or hazardous materials.

Exposures tab

In the **Product Definition** screen, exposures appears on the **Exposures** tab of a risk object. To see this tab, **Multiple Exposures for Insureds' Risk** on the risk object or product line must be selected. If you import a mind map with exposures, **Multiple Exposures for Insureds' Risk** is selected.

Field	Description
Label	Label that appears in the user interface.
Fields	Click to view or define Exposure Fields .
Type	Specify the type of exposure: • Liability • Property • Party • Other
Role	Appears if Type is Party . Choose: • Driver • Named Insured
Risk Location	Specify the location of the risk: • Risk is at parent's location – Risk is at the same location as the parent risk object. • Risk includes location reference – The exposure is a location. If selected, a Location field is added to the exposure.
Rating Basis	The basis for rating. Choose: • Pro-rata for term • Scaled within period • Mixed rating

Field	Description
Menu Label (*PD)	The menu label for exposures in the generated product. After you generate the product, the product uses this label; it does not appear in the visualized product.
Description (*PD)	Description of the field. This field is required for product generation and displays if Product Design Mode is Developer .
Short Name (*PD)	This field is required for product generation and displays if Product Design Mode is Developer . Unique name that identifies the field in the generated product. This name must be unique in the product line.

Splitting exposures in visualized products

In visualized products, you can define exposures as splittable into multiple rating periods. Exposures are splittable if one or more of their fields can be split.

Note: You cannot define exposures as splittable in the mind map.

Exposure Fields popup

In the **Exposure Fields** popup, fields have the following columns:

Column	Description
Label	Label that appears in the user interface.
Type	The type of data, such as text, number, monetary amount, or date-time. Involved party creates a contact field on the policy. Contact details include name, phone number, address, and other information. When viewing the Contacts for the policy, these contacts have the Involved Party role. The Involved Party  marker in a mind map becomes an involved party contact.
Basis Key	Select Yes if this field is a basis for other fields.
Ordering	The order of the items in the user interface, from first to last.
Name (*PD)	Name of the exposure field.
Description (*PD)	Description of the field. This field is required for product generation and displays if Product Design Mode is Developer .
Split By Rating Periods	This field can have split rating periods. Split rating periods create separate rating periods for dates selected by the user.
Scalable	Whether to use basis-scalable rating. When selected, the basis is scaled automatically before and after the split date. You can also manually scale the basis by editing the field values. This field is enabled if you select Split By Rating Periods , and Type is Number , Decimal (14,2) , or Monetary amount .

Define exposure as split by rating period

About this task

In the product definition, define an exposure as split by rating period.

Procedure

1. In the **Product Definition** screen, go to the **Exposures** tab.
2. On the exposure, select **Fields** to display the **Exposure Fields** popup.
3. If this field is used to calculate the basis, select **Basis Key**.
4. Select **Split By Rating Periods** on the field.
5. If the field is a number, decimal, or monetary amount, select **Scalable** if the field is scalable.

Split exposures in a policy transaction

About this task

In a policy transaction, you split the exposure and can modify independently the values of the split fields. Any number of splits may be performed on exposures. Different exposures may have different numbers of splits or different split dates.

Procedure

1. In a policy transactions for a policy line that has exposures that can be split, advance to **Risk Details** and select the **Exposures** tab.
2. Add the exposure that can be split.
For exposures that can be split, **Effective Date** and **Expiration Date** appear in the list view and the **Split** button appears above the list.
3. Select one or more of these exposures to enable the **Split** button.
4. Click **Split**.
The **Split Risks** popup appears.
5. Choose the **Date** on which to split the risks and click **Split**.

An error appears if the date is outside the policy period range or if a split already exists on the given date.

After a successful split, there are two versions of the exposure.

Both versions initially have the same data, except for scalable fields which are prorated based on the split date. Basis or number of employees are examples of scalable fields. You can edit splittable fields independently in both versions. Fields that cannot be split are editable only in the first version. Upon save, changes made in the first version propagate to the second version. Location and class code are examples of fields that cannot be split.

You can only select the first version of an exposure and **Remove** or **Split** it. Removing an exposure removes all versions. There is no way to remove a single version. There is no way to undo the split to an exposure.

Editions for product lines

Editions define product model properties and subclause relationships for product lines. Each product line has a **Base** edition that defines product model properties and subclause relationships for the base product line. You can use editions to introduce changes to the product after a product has gone into production. Typically, products start out with only the Base edition. Create additional editions if there is a filing or other business reason to change the product as of a certain effective date.

- The Base edition defines product model properties for the product line.
- Editions have an effective date.
- Editions have no expiration date.
- Later editions inherit product model properties from previous editions, including the Base edition.
- Later editions override changes introduced in prior editions.

With editions, you can:

- Change an insurance product line after it has gone into production
- Load product model properties into a generated product.

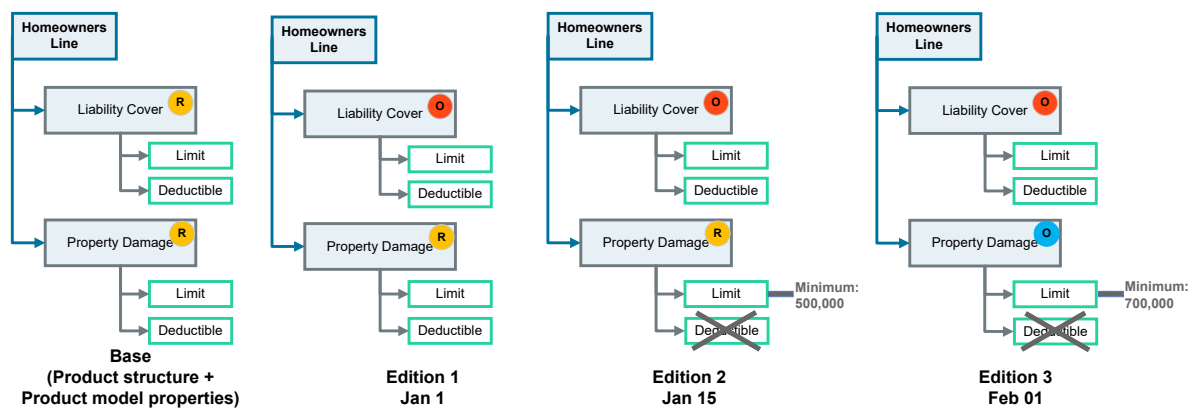
In a policy transaction, either with installed or visualized product, PolicyCenter selects the edition based on the **Effective Date** on the **Policy Info** screen.

See also

- “Loading editions into generated product” on page 109

Example of editions

This example shows a Homeowners product line with Liability Cover and Property Damage coverages. Both coverages have Limit and Deductible fields. The editions change product model properties that set the availability of the coverages and their fields.



- In the Base edition, Liability Cover and Property Damage coverages are required.
- Edition 1 sets Liability Cover to optional.
- Edition 2 makes Property Damage Deductible unavailable and sets the Limit minimum value to 500,000. Liability Cover remains optional.
- Edition 3 makes Property Damage optional and increases the Limit minimum value to 700,000. Liability Cover remains optional, and Property Damage Deductible remains unavailable.

Adding editions out-of-sequence

If you create an edition that has an effective date before existing editions, that edition is out-of-sequence to editions with later effective dates. As with any edition, changes in an out-of-sequence edition carry forward until superseded by a later edition.

Structural changes apply to all editions

When you are working in the base edition or subsequent edition, you can introduce structural changes to the product line.

These kinds of changes include but are not limited to:

- Adding a new risk object, clause, attribute, or other structural element.
- Adding a new clause, such as a coverage, and making it available starting on September 1st for all policies.
- Changing the label of an attribute or a clause. For example, changing the wording of a question attribute.
- Changing the name of a risk object or clause.

Adding or removing subclause relationships, without adding or removing clauses or clause terms, is not a structural change.

Only change the name, label, or other wording if the intent remains the same. Downstream systems may rely on the original meaning. For example, if you need a different attribute, then make the original attribute unavailable, and add a new one.

Even though you make these changes while editing a particular edition, these structural changes apply to all editions.

Note: Inform the administrator who will load the editions that you made structural changes to the product. For structural changes, the administrator must regenerate the product and redeploy.

Editions tab

You can view and define editions on the **Editions** tab on the **Product Definition** screen.

Field	Description
Edition Code	The edition. Displays Base for the edition on the base product line.
Effective Date	The date that changes in the edition become effective. A change remains in effect until a change in a later edition supersedes it. The Base edition has no effective date.
Create Edition	Create a new edition.
Use This Edition	Select the check box for an edition, and click Use This Edition . The selected edition becomes the current edition. Use This Edition is disabled if you select the current edition, more than one edition, or no editions.
Delete	Select the check box for an edition, and click Delete .
Editions	This table displays all editions defined for the product line.
Changes	This column displays the number of changes to product model properties in this edition. In the Base edition, the number of changes is always blank.

Edition property popup

When you view a product model property, the Property popup contains the following fields related to editions:

Field	Description
Delete Rule Edition	Delete the rule for the current edition. This button appears if there is a rule defined on this property in the current edition.
Rule Effective Date	Relative to the current edition, the effective date of the latest edition that edited the rule. If the latest edition is the Base edition, the field displays <base rule> . If the latest edition is the current edition, the field displays the effective date of the current edition, and Current Edition Effective Date is hidden.
Current Edition	Displays the name of the current edition. Hidden if the current edition is the Base edition.
Current Edition Effective Date	The effective date of the current edition. Only appears if the effective date is different than the rule effective date.

Edit

To make changes for the current edition, click **Edit For Current Edition**.

Create an edition

Create an edition in a visualized product.

Before you begin

Set **Product Design Mode** to **Designer** or above.

About this task

By default, all product lines can define product model properties for the base product line in the edition named **Base**. When you use the Base edition, changes made to product model properties, such as availability criteria or minimum or maximum values, are made in the base product line. For example, you have a user-entered limit that is a coverage term. You set the default minimum to 1,000 and maximum to 100,000 in the base edition.

If you add an edition, you can make changes to product model properties for that edition.

Procedure

1. Click the **Editions** tab.
2. Click **Create Edition** to create a new edition.
3. In the **Add new Edition for** popup, enter **Edition Name**, **Description**, and **Effective Date**.
Effective date must be unique. You cannot have two editions with the same effective date.
4. Click **Update and Edit** to add the edition and return to editing the edition.
By default, the newly created edition is the current edition and is highlighted in the table. The **Product Definition** screens display the product model properties for this edition.

Change product model properties in an edition

Before you begin

- Set **Product Design Mode** to **Designer** or above.
- “Create an edition” on page 81.

Procedure

1. On the **Product Definition** screen for a product line, click the **Editions** tab.
The tab displays the number of editions. If you have only the Base edition, the tab displays **Editions (0)**.
In the list of editions, the current edition is highlighted. When you navigate to the tab in the **Product Definition** screen, they display the product model properties for the current edition.
2. To make another edition the current edition, select the check box, and click **Use This Edition**.
3. Make a change to a product model property for the current edition. For example, as of this edition, you want to increase the minimum limit to 5,000.
 - a. Go to the **Fields** tab.
 - b. Select **Default Value** for a field.
On the **Default Property** popup, the box in the upper right contains information about the edition that you are viewing. **Current Edition** displays the name of the current edition. This field is hidden when working on the Base edition. See “Edition property popup” on page 81 for more information.
 - c. To make a change, click **Edit For Current Edition**.
You can now add or change the default value, and add or change conditional exceptions.
 - d. Click **OK** to save your changes.
The changes you made are part of the current edition.

Delete an edition

Before you begin

Set **Product Design Mode** to **Designer** or above.

Procedure

1. Go to the **Product Definition** screen for a product, and click **Edit Product**.
2. Select the **Editions** tab.
In the list of editions, the current edition is highlighted. The **Product Definition** screens display the product model properties for the current edition.
3. Select the check box for the edition you want to delete.
The **Delete** button is enabled. You can delete any edition except for the Base edition.
4. Click **Delete**.

Pricing for rating in visualized product

The price of a policy is the sum of all premiums, taxes and other charges. In PolicyCenter, *costs* are the individual components that determine the price. In simple terms, the amount of a cost is calculated by applying a rate to a basis. As you develop your product, the rate is ultimately calculated by the rating engine. You can enter the rate in a policy transaction and see how the rate affects the policy premium. After the product is generated, you must recreate the rates and the basis in the rate books that calculate the policy premium.

In Advanced Product Designer, you can create three types of pricing:

- Exposure to risk is defined by the coverage. The premium is calculated by applying a rate to an attribute of the coverage, such as sum insured.
- Exposure to risk is defined by something other than the coverage. For example, liability coverage is provided up to a limit of \$50 million, but the exposure to risk is the size of the business measured by classes of turnover. A cost is added and a rate supplied for each exposure of revenue.
- The premium is calculated from a base premium amount rather than some numeric value. In this case, there is no basis and the rate is simply the premium (the basis is one).

Advanced Product Designer can also visualize costs that are calculated from other costs. This may be related premiums. For example, business interruption premiums that are calculated from underlying material damage premiums, or more commonly, taxes and charges that are applied to premiums.

Recreate pricing information in visualized product

When you import a mind map, Pricing markers and their relationships are dropped. You can use the pricing relationships from the mind map as a guide to adding pricing information to your visualized product.

You can define pricing details in the **Pricing** screen in the **Product Definition** screen or in a submission or other policy transaction. Define the parameters and rates to be considered during the rating process for the associated lines and risk objects.

Rating integration

Product generation creates a rating framework for your product. Based on the costs added to each risk object or coverage in the **Pricing** screen during visualization, corresponding CostData objects and Cost entities are created by product generation. You must recreate the basis.

For example, a location has three coverage costs and one cost on the location (**Other Costs**) on the **Pricing** tab of the **Location Risk Object Definition** popup.

Location Risk Object Definition popup		
Pricing tab		
Coverage Costs		
Name	Cost Type	Debt Type
Equipment Breakdown Protection		

Location Risk Object Definition popup		
Premium	Standard premium	Premium, inc. Tax
VAT	Tax or surcharge	Premium, inc. Tax
Fire Department Services		
GST	Tax or surcharge	Premium, inc. Tax
Other Costs		
Name	Cost Type	
IPT	Tax or surcharge	Premium, inc. Tax

Product generation creates three costs for the Location coverage costs (Premium, VAT, and GST). It also creates an `XXLocationCovCostData` class and an `XXLocationCovCost` entity for the risk object (location).

One cost is created for the Location risk object cost (IPT), with an `XXLocationCostData` class and an `XXLocationCost` entity for this cost.

A `RatingEngine` class is also generated. You must implement the `rateSlice` method in this class.

Add pricing to a risk object

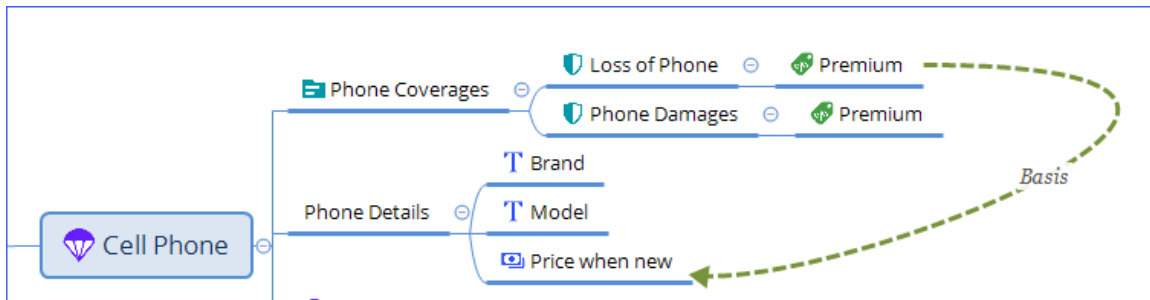
Specify pricing for a risk object in a product line.

Before you begin

Set **Product Design Mode** to **Designer** or above.

About this task

This task defines the premium for a coverage. In the mind map, the premium has a Pricing marker. The premium is based upon a percentage of the price when new. The following mind map shows the pricing relationship.



Procedure

1. Start a submission for a visualized product.
2. On the **Risk Details** screen, select the **Fields** tab.
3. Select **Add**→**Add risk_object**. In this example, *risk_object* is **Cell Phone**.
4. Enter values on the **Fields** tab.
5. Go to the **Coverages**. In this example, go to the **Phone Coverages** tab.
6. Select the coverages that you wish to price. In this example, select **Loss of Phone**.
7. When you are done selecting coverages, click **OK**.
8. Navigate to the **Pricing** screen and to the risk object.
The cell phone risk object you just added appears under **Risk Objects**. Click **Cell Phone** link.
9. Your coverage, **Loss of Phone**, is listed in the **Coverage Costs** table.

10. In **Cost Type**, select **Add Cost** from the drop-down list.
11. Define this cost. For this example, create basis pricing by entering the following information:

Code	premium
Basis Attribute	Cell Phone: Price when new
Rate	.014
Per	Per Unit

Pricing screen in submission in User mode

When you are in **User mode**, you can view the **Pricing** screen in a submission.

Set **Product Design Mode** to **User**.

In **User mode**, the **Pricing** screen displays the following:

- **Risk Objects** – List of all risk objects on the policy
- **Other Costs** – Other costs added to the policy in **Design** mode or higher.

In this mode, the **Other Costs** table has the following columns:

- **Name**
- **Basis Attribute** – The attribute from which the basis is obtained.
- **Basis** – The basis amount for this instance of the risk object.
- **Rate** – Although rating is only minimally defined, you can enter a rate to see how the rate affects the premium when you quote the policy.
- **Per** – Per 1000, 100, or per unit.

Coverage costs on risk objects have an additional **Exposure Pricing** column.

Adding costs in Pricing

Set **Product Design Mode** to **Designer** or above.

On the **Pricing** screen or tab, you can add coverage and other costs that can be used to price the policy. In the **Product Definition** screen, **Pricing** is a tab on the product screen and risk object screens. In a submission, **Pricing** is a step in the submission. This screen lists the risk objects on the policy. Click a risk object link to view the **Risk Object Pricing** popup.

The **Product Definition** screen provides a convenient way to define pricing within the product.

In a submission, you can define or modify pricing and generate a policy premium. Costs in a submission have a **Rate** column where you can enter rates for generating the policy premium.

In a submission, all risk objects on the policy appear under a **Risk Objects** heading. Select a risk object to specify costs for that risk object type in the **Risk Object Pricing** popup. These costs apply to this instance of the risk object. New instances of the risk object include all costs on the risk object. The product line is a type of risk object, and you can add costs to the product line on the **Pricing** screen.

Cost Type can be standard premium, non-standard premium, or tax or surcharge.

Recommended way to change costs in a saved product

You have a saved product, and need to change the code of a cost. For example, the cost is Premium but needs to be Income Premium. To make this change, edit the product, delete the cost (Premium), and add a new cost with the new code (Income Premium).

If you simply change the code of a cost in a saved product, the code is changed in all submissions, including bound submissions. You may not desire to have bound submissions affected by this change.

This recommendation does not apply to a product being edited for the first time.

Cost codes

Cost codes define the types of costs that you can add to a policy. Examples of cost codes include premium and value added tax. You define cost codes in the **Cost Definition Codes** screen.

Costs and how they appear on Policy Premium tab

The cost code of a cost affects how it appears in a submission as a charge on the **Policy Premium** tab of the **Quote** screen. This screen helps you define how you want billing information to be presented. You can specify that a cost is **Itemized for Billing** and appears as a separate line item in the bill. You can also specify that a cost has **Separate Collection** and is collected separately by the billing system.

Without **Itemized for Billing** or **Separate Collection**, the **Policy Premium** tab often displays **Premium, inc. Tax** billing collection charge that is the sum of the **Standard premium** and **Tax or surcharge** itemized charges:

Policy Premium Charge Type	Description	Settlement Amount
Premium, inc. Tax		\$242.20
	Standard premium	\$242.00
	Tax or surcharge	\$0.20

Charges for collection by the billing system are left-aligned in the **Charge Type** column and display the **Debt Type** of the charge. These charges are typically **Premium, inc. Tax**, which rolls up costs with a rate amount type of **Standard premium** or **Tax or surcharge**.

Itemized charges are right-aligned in the **Charge Type** column. These charges are rolled up into a charge for collection. These charges are typically **Standard premium** or **Tax or surcharge**.

Itemized for Billing

Itemized for Billing results in additional right-aligned itemized charge rows on the **Policy Premium** tab in a submission. You can specify **Itemized for Billing** on risk objects, coverages, and costs.

The following example shows itemization in the **Policy Premium** tab:

- Cell Phone: Samsung – Itemized charge on the cell phone risk object
- Cell Phone: Samsung | Loss of Phone | GST – Itemized charge on the GST cost on the cell phone's Loss of Phone coverage

Policy Premium Charge Type	Description	Settlement Amount
Premium, inc. Tax		\$121.10
	Standard premium	\$11.00
	Standard premium Cell Phone: Samsung	\$110.00
	Tax or surcharge Cell Phone: Samsung Loss of Phone GST	\$0.10

Separate Collection

Separate Collection results in additional left-aligned collection charges on the **Policy Premium** tab in a submission. These charges are billed separately.

The following example shows separate collection in the **Policy Premium** tab for the cell phone risk objects:

Charge Type	Description	Settlement Amount
Premium, inc. Tax		\$22.00

Charge Type	Description	Settlement Amount
Premium, inc. Tax	Cell Phone: Apple	\$110.10
	Standard premium	\$110.00
	Tax or surcharge	\$0.11
Premium, inc. Tax	Cell Phone: Samsung	\$110.10
	Standard premium	\$110.00
	Tax or surcharge	\$0.11

Itemized for Billing

Use **Itemized for Billing** to specify that a charge is listed as multiple line items on the invoices. This line item is right-aligned in the **Policy Premium** tab on the **Quote** screen in a submission.

On the **Pricing** screens, you can set **Itemized for Billing** on:

- Risk objects
- Coverages
- Coverage costs
- Other costs

In the **Policy Premium** tab, charges with the same risk type are rolled up into a right-aligned line item. You may need to increase the column width to see the alignment. If the line item charge is not itemized, the **Description** is blank. If **Itemized for Billing** is selected, the line item **Description** contains the name of the risk object, coverage, or cost being itemized.

The following examples show how itemized for billing changes the **Policy Premium** tab using a policy with a Cell Phone risk object with a Loss of Phone coverage. The submission has an Apple and a Samsung phone. Both cell phones have the same rates. The **Basis** of both phones is 1000.

Note: The **Rate** column appears in the **Pricing** screen in a submission or other policy transaction.

The list of **Risk Objects** does not appear on the **Product Definition**→**Pricing** tab.

Charges sent to billing system

Charges on the same risk type that are itemized for billing are grouped into a single charge, with the itemization in the charge breakdown details. This charge is sent to BillingCenter or the associated billing system.

No itemizing

If **Itemized for Billing** is No for all costs, then the **Policy Premium** tab on the **Quote** screen lists the three different charge types on the **Pricing** screens.

Policy Premium		
Charge Type	Description	Settlement Amount
Premium, inc. Tax		\$244.00
	Standard premium	\$220.00
	Tax or surcharge	\$24.00

The **Charge Type** information corresponds to the cost types. There is one left-aligned **Premium, inc. Tax** charge for collection by billing. This is the Debt Type. This charge is the sum of the right-aligned **Standard premium** plus **Tax or surcharge** charges that are **Premium, inc. Tax** that appear under the charge for collection.

- **Standard premium** – Sum of all Standard premium costs on the risk objects and on the policy.
- **Tax or surcharge** – Sum of all Tax or surcharge costs on the risk objects and on the policy.

The **Description** column is empty because there is neither itemization nor separate collection.

If nothing is itemized, the **Pricing** screen has the following information related to **Itemized for Billing**:

Pricing					
Risk Objects					
Cell Phone: Apple					
Cell Phone: Samsung					
Other Costs					
Name	Cost Type	Itemized for Billing	Basis Attribute	Rate	Per
GST	Tax or surcharge	No	<prior costs>	0.1	Per Unit

Note: <prior costs> is based on Premium.

For each cell phone risk object, the **Risk Object Pricing** screen has the following information related to **Itemized for Billing**:

Risk Object Pricing						
Cell Phone: <i>Name</i>						
Itemized for Billing	No					
Coverage Costs						
Name	Cost Type	Itemized for Billing	Basis Attribute	Basis	Rate	Per
Loss of Phone		No				
Premium	Standard premium	No	Cell Phone: Price when new	1000	0.1	Per Unit
GST	Tax or surcharge	No	<prior costs>		0.01	Per Unit
Other Costs						
Premium	Standard premium	No	Price when new	1000	0.01	Per Unit

Itemizing risk objects

If you itemize a risk object, costs of the same charge type are rolled up into a single line item for each instance of the risk object on the **Policy Premium** tab. The itemized charges on the risk object are identified in **Description**. Costs of the same charge type are rolled up into a single line item for each instance of the risk object. Therefore, the standard premium for the Apple cell phone is \$110.00, the sum of these standard premium amounts:

- Loss of Phone standard premium at 0.1%: \$100
- Other Costs standard premium at 0.01%: \$10

Policy Premium		
Charge Type	Description	Settlement Amount
Premium, inc. Tax		\$244.00
	Tax or surcharge	\$22.00
	Standard premium Cell Phone: Apple	\$110.00
	Tax or surcharge Cell Phone: Apple	\$1.00

Policy Premium Charge Type	Description	Settlement Amount
Standard premium	Cell Phone: Samsung	\$110.00
Tax or surcharge	Cell Phone: Samsung	\$1.00

In the **Risk Object Pricing** screen, the cell phone risk objects are **Itemized for Billing**:

Risk Object Pricing	
Cell Phone: Samsung	
Itemized for Billing	Yes

Itemizing coverages

If you itemize a coverage, costs of the same charge type are rolled up into a single line item for each instance of the coverage on the **Policy Premium** tab. In this example, Cell Phone risk object and the Loss of Phone coverage are itemized for billing.

On the **Policy Premium** tab, itemized charges on the coverage are identified in the **Description**. For each instance of the coverage, the costs of the same charge type are rolled up into a single line item. The Cell Phone: Samsung | Loss of Phone coverage has two itemized charges:

- Standard premium charge type
- Tax and surcharge charge type

Costs are also itemized on the cell phone risk object, therefore any remaining costs of the same type on the Cell Phone risk object are rolled up into a single line item. The Cell Phone: Samsung has an itemized standard premium charge.

Policy Premium Charge Type	Description	Settlement Amount
Premium, inc. Tax		\$122.00
Tax or surcharge		\$11.00
Standard premium	Cell Phone: Samsung	\$10.00
Standard premium	Cell Phone: Samsung Loss of Phone	\$100.00
Tax or surcharge	Cell Phone: Samsung Loss of Phone	\$1.00

The **Risk Object Pricing** screen for the cell phone has the following information related to **Itemized for Billing**:

Risk Object Pricing						
Cell Phone: Samsung						
Itemized for Billing	Yes					
Coverage Costs						
Name	Cost Type	Itemized for Billing	Basis Attribute	Basis	Rate	Per
Loss of Phone		Yes				

Itemizing individual costs

If you itemize a risk object and itemize individual costs on the risk object, both the risk object and the itemized costs appear on separate line items in the **Policy Premium** tab. In this example, the cell phone risk object and GST cost on Loss of Phone coverage are itemized. Loss of Phone coverage is not itemized.

On the **Policy Premium** tab, the GST appear as a separate tax or surcharge line item with the description **Cell Phone: Samsung | Loss of Phone | GST**. Any other tax or surcharge items are rolled up into the **Cell Phone: Samsung** line item.

Policy Premium		
Charge Type	Description	Settlement Amount
Premium, inc. Tax		\$122.00
Tax or surcharge		\$11.00
Standard premium	Cell Phone: Samsung	\$110.00
Tax or surcharge	Cell Phone: Samsung Loss of Phone GST	\$1.00

In the **Risk Object Pricing** screen for the cell phone, **Itemized for Billing** is set to Yes for the cell phone risk object and the GST cost on Loss of Phone coverage. Loss of Phone coverage is not itemized.

Risk Object Pricing						
Cell Phone: Samsung						
Itemized for Billing	Yes					
Coverage Costs						
Name	Cost Type	Itemized for Billing	Basis Attribute	Basis	Rate	Per
GST	Tax or surcharge	Yes	Cell Phone: Price when new	1000	0.01	Per Unit

Separate Collection

Use **Separate Collection** for charges that are collected separately by the billing system. For example, bill each location on a policy separately. In the examples presented here, a family has a cell phone policy, and each family member has a cell phone. They have agreed that each family member is responsible for paying the charges for their cell phone. In their policy, **Separate Collection** is Yes on the Cell Phone risk object.

On the **Policy Premium** tab of the **Quote** screen in a submission, all charges for separate collection by billing are left-aligned in the **Charge Type** column. You may need to increase the column width to see the alignment. The column displays the **Debt Type** of the charge. The debt type is usually **Premium, inc. Tax**, but can also be **Premium** or another value. See “Debt Type” on page 97.

On the **Pricing** screen, you can set **Separate Collection** on:

- Risk objects
- Coverages
- Coverage costs
- Other costs

The following examples use a submission with a Cell Phone risk object with a Loss of Phone coverage. The submission has an Apple and a Samsung phone. Both cell phones have the same rates. The **Basis** of both phones is 1000.

Note: The **Rate** column appears in the **Pricing** screen in a submission or other policy transaction.

Separate collection on risk object

Payment for each cell phone on the policy is collected separately by the billing system.

If you have separate collection on a risk object, a separate left-aligned line item for the **Premium, inc. Tax** for each instance of that risk object appears on the **Policy Premium** tab. **Premium, inc. Tax** is a **Debt Type**. Below this line item, the individual charges for the risk object are right-aligned.

For each instance of the risk object, the costs of the same charge type are rolled up as specified in **Itemized for Billing**. In this example, **Itemized for Billing** is always No.

Policy Premium Charge Type	Description	Settlement Amount
Premium, inc. Tax	Cell Phone: Apple	\$111.00
	Standard premium	\$110.00
	Tax or surcharge	\$1.00
Premium, inc. Tax	Cell Phone: Samsung	\$111.00
	Standard premium	\$110.00
	Tax or surcharge	\$1.00

The **Pricing** screen has the following information related to **Separate Collection**. **Separate Collection** is No on this screen.

Pricing							
Risk Objects							
Cell Phone: Samsung							
Cell Phone: Apple							
Other Costs							
Name	Cost Type	Itemized for Billing	Debt Type	Separate Collection	Basis Attribute	Rate	Per

The **Risk Object Pricing** screen for cell phones has the following information related to **Separate Collection**, which is Yes for the Cell Phone risk object:

Risk Object Pricing								
Cell Phone: <i>Name</i>								
Itemized for Billing	No		Separate Collection	Yes				
Coverage Costs								
Name	Cost Type	Itemized for Billing	Debt Type	Separate Collection	Basis Attribute	Basis	Rate	Per
Loss of Phone		No		No				
Premium	Standard premium	No	Premium, inc. Tax	No	Cell Phone: Price when new	1000	0.1	Per Unit

Risk Object Pricing								
GST	Tax or surcharge	No	Premium, inc. Tax	No	<prior costs>		0.01	Per Unit
Other Costs								
Premium	Standard premium	No	Premium, inc. Tax	No	Price when new	1000	0.01	Per Unit

Separate collection on costs

You can specify that a cost is collected separately by the billing system. If a cost is marked for **Separate Collection**, then the charges for that cost appear left-aligned as a separate item charge for collection on the **Policy Premium** tab. In this example, only the premium on Loss of Phone coverage is collected separately. Cell Phone charges are not collected separately. On the **Policy Premium** tab, the premium charges for each phone are left-aligned. All other charges are summed into the \$22.00 Premium, inc. Tax.

Policy Premium			Settlement Amount	
Charge Type	Description			
Premium, inc. Tax			\$22.00	
	Standard premium		\$20.00	
	Tax or surcharge		\$2.00	
Premium, inc. Tax	Cell Phone: Apple Loss of Phone Premium		\$100.00	
Premium, inc. Tax	Cell Phone: Samsung Loss of Phone Premium		\$100.00	

The **Risk Object Pricing** popup has **Separate Collection** set on the Premium cost on Loss of Phone:

Risk Object Pricing								
Cell Phone: Name								
Itemized for Billing	No		Separate Collection	No				
Coverage Costs								
Name	Cost Type	Itemized for Billing	Debt Type	Separate Collection	Basis Attribute	Basis	Rate	Per
Loss of Phone		No		No				
Premium	Standard premium	No	Premium, inc. Tax	Yes	Cell Phone: Price when new	1000	0.1	Per Unit

Multiple debt types

In addition to Premium, inc. Tax, a policy may have other debt types. Each debt type is a separate collection charge for the billing system. In this example that has a single cell phone, the cell phone has a non-standard premium that is collected separately.

On the **Policy Premium** tab, the Premium and Premium, inc. Tax charges are left-aligned.

Policy Premium Charge Type	Description	Settlement Amount
Premium	Cell Phone: Samsung Non-standard Premium	\$10.00
Premium, inc. Tax		\$101.00
Standard premium		\$100.00
Tax or surcharge		\$1.00

The **Risk Object Pricing** popup has **Separate Collection** on the Non-standard Premium cost on the phone:

Risk Object Pricing Cell Phone: Samsung								
Itemized for Billing		No		Separate Collection	No			
Other Costs								
Non- standard Premium	Non- standard Premium	No	Premium	Yes	Price when new	1000	0.01	Per Unit

Basis attribute

In a cost, the cost basis is obtained from the basis attribute.

The **Basis Attribute** can be set to:

- Attributes in scope of the parent
- Prior costs

To select an attribute in scope, select from the **Basis Attribute** drop-down list. This list displays all Number, Decimal, Monetary Amount attributes that are within scope of the parent. If you choose an attribute on an exposure in a submission, the **Exposure Pricing** button appears.

To use prior costs, from the picker , select **Use Prior Costs**.

Basis attribute scope

When defining the basis attribute for a cost, the **Basis Attribute** drop-down list displays all number, decimal, monetary amount attributes that are within scope of the current risk object or coverage. The scope depends upon whether the cost is on the risk object (**Other Costs**) or on a coverage (**Coverage Costs**).

Costs on the risk object

For costs on the risk object (**Other Costs**), the scope of the basis attribute includes all number, decimal, and monetary amount attributes in:

1. Fields on the risk object
2. Fields on all parent risk objects

Note: The order of the list is not significant, but the numbers are used in the examples below.

Costs on coverages

For costs on coverages (**Coverage Costs**), the scope of the basis attribute includes all number, decimal, and monetary amount attributes in:

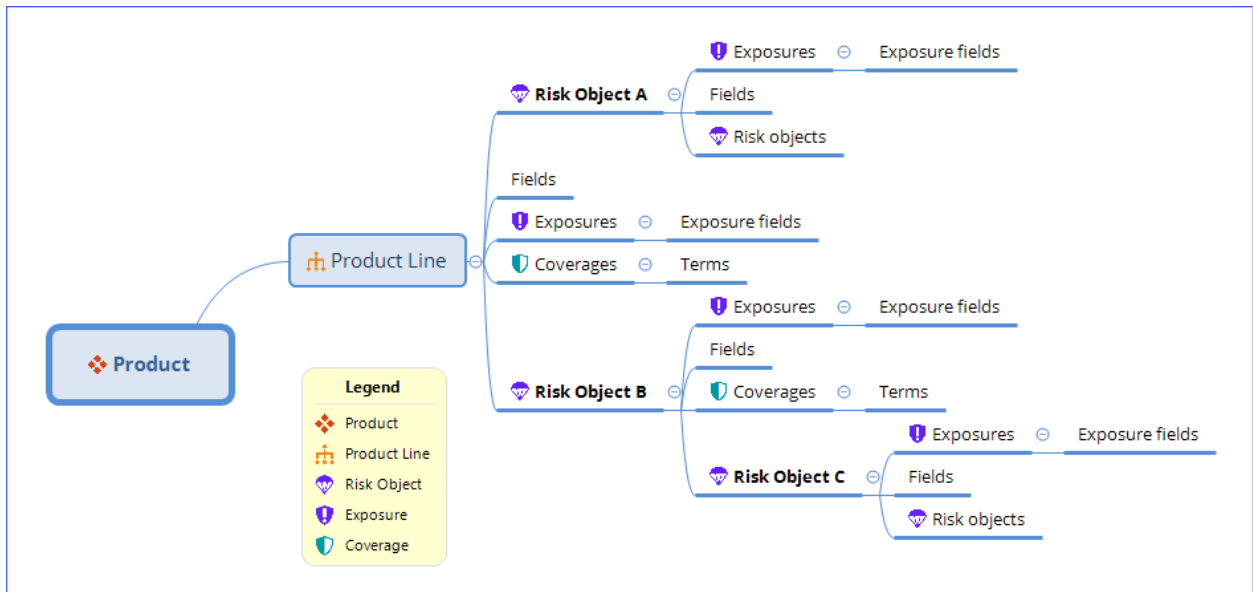
1. Fields on the parent risk object
2. Exposure fields on the parent risk object

3. Terms attached to this coverage, including option and package coverage terms
4. Fields from all risk objects above the parent risk object
5. Fields and exposure fields on risk objects that:
 - Do not have coverages, and
 - Are attached to the risk object that defines the current coverage

Note: The order of list items is not significant. The examples use the order to show how scope is affected by each list item.

Example

The following mind map shows a hierarchy of risk objects. Risk objects A and C are risk objects without coverages. Risk object B is a risk object with coverages. The Product Line is also a risk object.



Note: This mind map shows a generic hierarchy of risk objects in a product line. The map does not adhere to the rules for defining a product, and cannot be imported into PolicyCenter as a visualized product.

Risk object A

In **Other Costs** for risk object A, you can set the basis attribute to number, decimal, and monetary amount attributes on:

1. Fields on risk object A
2. Fields on the product line

Risk object C

In **Other Costs** for risk object C, you can set the basis attribute to number, decimal, and monetary amount attributes on:

1. Fields on risk object C
2. Fields on risk object B
3. Fields on the product line

Coverages on product line

In **Coverage Costs** on coverages on the product line, you can set the basis attribute to number, decimal, and monetary amount attributes on:

1. Fields on the product line
2. Exposure fields on the product line
3. Coverage terms, including option and package, on this coverage
4. (Does not apply in this example)
5. Fields and exposure fields from risk objects A and C. Because risk object B has coverages, it is out of scope.

Coverages on risk object B

In **Coverage Costs** on coverages on risk object B, you can set the basis attribute to number, decimal, and monetary amount attributes on:

1. Fields on risk object B
2. Exposure fields on risk object B
3. Coverage terms, including option and package, on this coverage
4. Fields from the product line
5. Fields and exposure fields from risk object C

Exposure pricing in visualized product

With exposure pricing, exposure to risk provides the basis for rating. For example, the cost of Phone Damages coverage is based on household members of various age ranges.

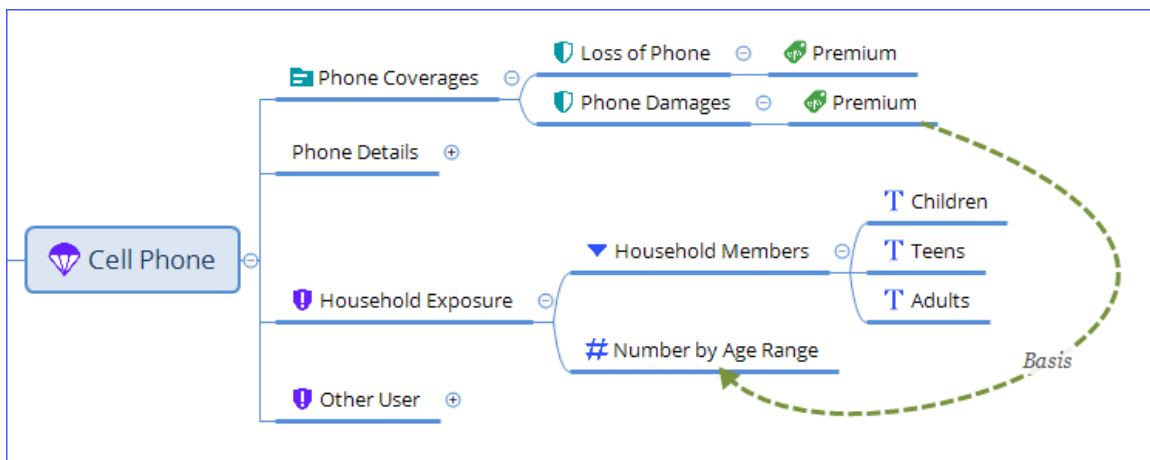
Add exposure pricing in visualized product

About this task

You can define exposure pricing on the **Exposure** tab in **Product Definition** and a submission or other policy transaction. These steps reference the mind map in “Exposure pricing in visualized product” on page 95.

This example uses the following mind map with an exposure and exposure fields. The Cell Phone risk object has a Household Exposure with Household Members drop-down list. Household members can be Children, Teens, or Adults. The exposure also has a Number by Age Range number topic where you can specify the number of household members in a particular age range.

The Phone Damages coverage has a Premium pricing topic that will be based upon Number by Age Range. When you import the mind map, pricing topics and their relationships are dropped. You recreate this relationship in the visualized product.



Procedure

Product Definition: define exposures

1. In **Product Definition**, go to the **Exposures** tab on the risk object.

2. In the exposure, select **Fields** to display the **Exposure Fields** popup.
3. To display a field for pricing that is a basis for other fields, select **Basis Key**.
For example, your choice in Household Members is the basis for Number by Age Range. Therefore, select **Basis Key** on the Household Members drop-down list.

Submission: define pricing

4. Create a submission on the product and add instances of the exposure and the coverage whose pricing is based on the exposure. For example:
 - a. Add a Cell Phone risk object with Loss of Phone coverage.
 - b. Select the **Exposures** tab, click **Add Household Exposure** for **Children**, **Teens**, and **Adults**, and specify the **Number by Age Range** for the current submission.
5. Advance to the **Pricing** screen or **Risk Object Pricing** popup for the risk object.
6. Add a cost and set **Basis Attribute** to an attribute on an exposure. In this example, add a Premium cost and set the basis attribute to Household Exposure : Number by Age Range.
Because you selected an attribute on an exposure, the **Exposure Pricing** button is enabled.
7. Click **Exposure Pricing**.
8. To test generating the premium, enter a **Rate** for each exposure.
9. Add the coverage for which this exposure is the basis. In this example, add **Phone Damages** coverage.

Prior costs

You can set the basis attribute of a cost to use prior costs. To use prior costs, select the picker  and choose **Use Prior Costs**.

Prior costs are the costs that are calculated before the current cost. Costs are calculated in the pricing order.

The **Pricing** screens display costs in increasing pricing order, starting with the first cost on the first coverage, and progressing until the final cost in **Other Costs**.

To change the pricing order on costs, select **Up** or **Down** from the picker . To change the pricing order of coverages, select **Up** or **Down** from the **Actions** drop-down list.

In the following example of pricing a cell phone risk object, the Loss of Phone Premium coverage cost is first in the pricing order, followed by Loss of Phone Tax. Tax in **Other Costs** is the highest in the pricing order.

Cell Phone	
Coverage Costs	
Code	Name
Loss of Phone	Loss of Phone Premium
	Loss of Phone Tax
Phone Damages	Damage Premium
	Damage Tax
Other Costs	Premium
	Tax

Coverage costs

For coverage costs, prior costs include:

1. All costs on the current coverage with a lower pricing order
2. All costs on coverages with a lower pricing order than the current coverage

Costs on risk object

For costs on the risk object (**Other Costs**), prior costs include:

1. All costs with a lower pricing order on this risk object
2. Costs for all coverages on this risk object
3. All costs, including coverage costs, on child risk objects

Cost codes

On the **Cost Definition Codes** popup, the **Code** column includes sample cost codes defined by Advanced Product Designer:

Code	Description
gst	Goods and services tax
incomePremium	Income replacement premium
ipt	Insurance premium tax
premium	General premium
stampDuty	Stamp duty
surcharge	Surcharge
tax	Tax
vat	Value-added tax

If you are in **Designer** mode or above, you can add or remove cost codes.

Rate Amount Type

In the **Cost Definition Codes** popup, the **Rate Amount Type** distinguishes between premium and non-premium costs. The following table includes each type and describes how it appears in the **Policy Premium** tab of **Quote** screen:

Rate Amount Type	Label in Policy Premium tab of Quote
Standard premium	Standard Premium
Non-standard premium	Premium
Tax or surcharge	Tax or surcharge

Debt Type

In the **Cost Definition Codes** popup, the **Debt Type** specifies the charge pattern for billing. Costs with the same charge pattern can be sent as one charge to the billing system. You can choose from the following:

Debt Type	Description
Premium, inc. Tax	Premium, including tax
Surcharge	An additional tax or charge
Installment Fee	Fee for processing premium or installments on the policy
Premium	Non-standard premium

Proration

In **Proration**, you can choose from the following:

Proration	Description
Day-based pro rata amount	Prorate the cost by number of days that the coverage is on the policy
Flat	The cost is the same regardless of number of days that the coverage is on the policy

Overrideable

Specify whether costs using this code can be overridden. In a policy transaction, you can do overrides on the **Rating Overrides** screen.

Configuring cost codes

When you select **Regenerate System Table XML**, the cost codes are stored in the `costcode_ext.xml` system table. You can access this file in Studio.

Modify cost code definitions

Use these instructions to modify, add, or remove cost codes.

Before you begin

Set **Product Design Mode** to **Designer** or above.

About this task

When editing a visualized product through the **Product Management** screens, if you add, delete, or change a cost code on the **Cost Definition Codes** popup, click **Regenerate System Table XML** to save these changes to the file system. These changes apply to all existing and new submissions in both visualized and generated products.

To work in **Product Management** without affecting generated products and the system, click **OK** to save your changes, but do not click **Regenerate System Table XML**. If you click **OK** only, your changes are available to new submissions of visualized products, but are not available to existing submissions.

In a submission, if you add or change a code but click **OK** only, then your changes are available to all existing and new submissions for visualized products.

Procedure

1. Navigate to the **Pricing** screen.
 - In a submission for a visualized product, go to the **Pricing** step.
 - In **Product Definition**, select the **Pricing** tab.
2. In **Other Costs**, click **Add** to add a cost.
3. In the **Code** column, select **Edit Cost Codes** from the picker .

The **Cost Definition Codes** popup appears.
4. Modify, add, or remove cost codes.
5. To save your changes to the file system, click **Regenerate System Table XML**.
6. Click **OK** to save your changes.

Multi-line products

Add product lines to a multiline product

You can add new or existing product lines to a multiline product in PolicyCenter.

Before you begin

- Set **Product Design Mode** to **Designer** or above.

Procedure

1. Select a product at the top of the **APD Managed** tab.
The selected product appears in the **Details** tab.
2. Click **Edit Product**.
PolicyCenter displays the **Product Definition** screen.
3. Select **Yes** in **Multiline**.
4. In **Product Lines**, from the add **Add** drop-down list select **Create New Product Line** or **Add Existing Line to Product**.
5. If you are creating a new product line, enter the information for the new line.

Remove or delete product line from multi-line product

You can remove a line from a visualized multi-line product. You can also delete the line completely.

Before you begin

- Set **Product Design Mode** to **Developer**.

About this task

If you remove a line from a visualized product, the line may still exist in other visualized products. If you delete a line, then no visualized products can include it.

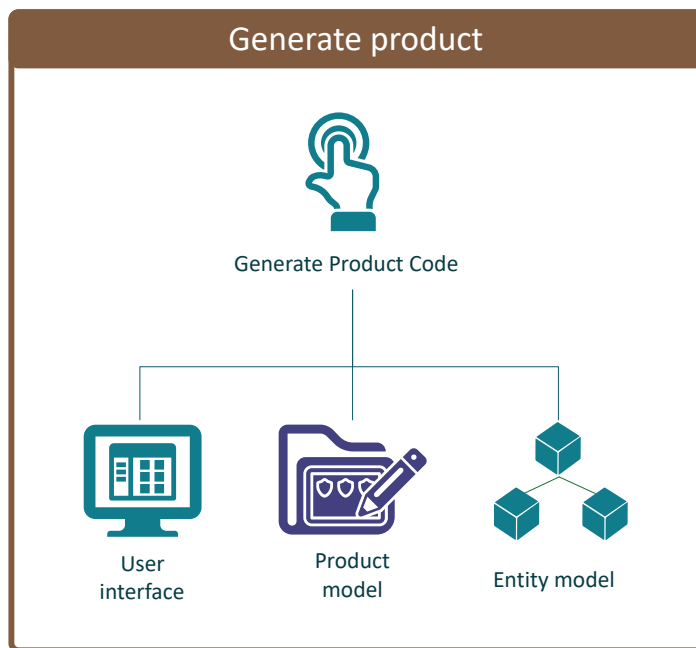
Procedure

1. In the **Product Definition** screen of a multi-line visualized product, Select the line that you wish to remove or delete.
2. Click **Remove Line From Product** to remove a line or **Delete Product Line** to delete a line.
If you remove a line, you can add it back by selecting it from the **Add→Add Existing Line To Product** menu.

Finalizing and generating the product

About this task

When you have refined and extended the visualized product, you are ready to finalize the product. Finalization is the phase where the visualized product is used to generate the artifacts needed for PolicyCenter including user interface elements, product model, and entities. The result is a generated product.



The result of finalization is a generated product that has a check in the **Installed** column on the **APD Managed** tab of the **New Submissions** screen in PolicyCenter.

Save frequently during finalization

Use **Export Template** on your product prior to the final build.

Note: If you drop your database before the final build is complete, you will lose your work on the visualized product.

Parts of visualized product not in generated product

- The rate in pricing
- Event data

Editions must be loaded

Editions containing the product model properties must be loaded into the generated product. See “Load and activate editions in generated product” on page 111.

Develop features not provided in Advanced Product Designer

For features not provided for in Advanced Product Designer, continue development using tools provided by PolicyCenter and described in the PolicyCenter documentation. Complete development by adding underwriting rules and rating, event data, integrating with other applications, and making other changes to meet your needs.

Continue product design

You can continue to enhance the product in Advanced Product Designer as described in “Continuing product design” on page 14.

Procedure

1. Review all metadata codes and descriptions to ensure that they follow Guidewire suggested protocols, or protocols agreed upon by your the project. See “Metadata” on page 102.
2. Review all content. See “Review product content” on page 105.
3. In the PolicyCenter installation where you will generate the product code, set the package name for generated product line Gosu files in `APDProductLinesGosuPackage` script parameter. See “Define Gosu package for product line files” on page 17.
4. Validation reviews the product definition for completeness and correctness. Validation can be done in any PolicyCenter instance where business users create products.
 - a. Business user starts validation of the product definition. See “Validate product in Designer mode” on page 106.
 - b. Developer completes validation of the product definition. See “Validate product in Developer mode” on page 106.
5. (Optional) When the product is valid, save it as a template and check it into version control.
6. (Optional) If it is not already there, move the product to the PolicyCenter instance where it will be generated and deployed. See “Move product” on page 107.
7. Generate, build, and deploy the product. See “Generate and deploy the product” on page 107.

You generate the product in the PolicyCenter instance where you continue development. Product generation accesses and saves files in the PolicyCenter installation. The product may need to be regenerated after making changes.

Metadata

You must add metadata for all elements in the product model tree in fields labeled **(*PD)**. These fields are not required to visualize the product, but are required for product generation.

The metadata includes business descriptions and codes, such as:

- General product details
- Line details and risk objects
 - Fields
 - Coverages, conditions, and exclusions
 - Drop-down lists associated with fields, coverages, conditions, and exclusions

The product model is not valid until all metadata has been entered.

How to specify metadata

You can specify metadata for the product model by:

- Entering (*PD) values in the visualized product screens directly.
- Defining short names in the mind map.

Adding Short Name (*PD)

If **Product Design Mode** is **Developer**, the **Product Definition** screens have **Short Name (*PD)** fields. These fields must be filled in before generating the product. For drop-down lists, you specify the short name in the **Typelist (*PD)** field.

When designing a product, create short names that follow best-practice naming conventions agreed upon by the organization or team.

Mind map label for short name

For all topics with markers (except Product  and Product Line ) in the XMind mind map, you can specify the short name in a label. The label becomes the **Short Name (*PD)** in the visualized product.

Short name requirements

Short names must:

- Start with an uppercase letter
- Include only letters, digits, \$ or _
- Clauses and clause categories – Be unique within the product line
- Clause terms – Be unique within the parent clause
- Drop-down list entries – Can start with a lowercase letter or digit

Short Name (*PD) fields have character limits that are enforced in the user interface. If you import a mind map or template where short names exceed the character limits, the short names are truncated. **Short Name (*PD)** is visible if **Product Design Mode** is **Developer**.

The character limits for short names are:

Fields	Limit	Description
Clauses	45	
Clause categories	45	
Clause terms	30	
Risk objects	22	
Exposures	22	
Attributes or fields	30	
Drop-down list values	45	
Product line	22	On the Product Definition screen, PolicyLine Subtype , excluding prefix (*PD)

Words to avoid

The short name cannot be one of the following keywords. The check is case-insensitive.

- Java keywords
- Gosu keywords – See *Gosu Reference Guide*
- The following SQL keywords:
 - use
 - type

The name can include these words. For example, abstract is a Java and Gosu keyword, so the short name cannot be ABSTRACT, but can be abstractPhrase.

Words to avoid for exposures and coverables

For exposures and coverables, field short names cannot match certain property names on PolicyCenter entities. Coverables are risk object that have coverages, including product lines. Using one of these words results in a warning, but does not block you from completing code generation. Guidewire recommends that you change the name so that no warning is issued.

On exposures and coverables, including the product line, do not use these words as field short names:

- | | |
|-----------------|------------------|
| • Changed | • ExpirationDate |
| • Class | • New |
| • DisplayName | • TYPE |
| • EffectiveDate | |

On coverables, including the product line, do not use this word as field short names:

- State

On the product line, do not use these words as field short names:

- AdditionalInsureds
- BaseState
- Branch

Note: These are the words to avoid that you are most likely to use. This is not a complete list.

Add metadata

Before you begin

- Set **Product Design Mode** to **Developer**.

About this task

Note:

Do not add prefixes to codes that identify the line or risk object. The build process will do this for you. These codes end with **excluding prefix (*PD)**.

Procedure

1. Select a product at the top of the **APD Managed** tab.
The selected product appears in the **Details** tab.
2. Click **Edit Product**.
PolicyCenter displays the **Product Definition** screen.
3. Starting with the **Details** tab, enter values for all fields appended with **(*PD)**.

Fields

4. On the **Fields** tab, enter **Menu Label for Generated UI (*PD)**.
5. On the **Fields** tab, click **Definition** for a drop-down list.

The **Drop-Down Definition** popup appears.

6. Enter **Description (*PD)**.

Coverages, Exclusions, and Conditions

7. On the **Coverages, Exclusions, or Conditions** tab, enter **Description (*PD)** for each row in the table.

8. For each row in the table, click **Terms** to update **Description (*PD)** and **Short Name (*PD)** for each term.

9. If terms have drop-down lists, update **Description (*PD)** and **Short Name (*PD)** for each list entry.

10. At the top of the **Coverages, Exclusions, or Conditions** tab, click **Manage Clause Categories** to update **Description (*PD)**.

Risk objects

11. Click the **Children** tab to set metadata on risk objects.

A table contains a link to each risk object.

12. For each risk object, click the link to display the **Risk Object Definition** popup.

13. On the **Details** tab, enter metadata.

Description (*PD)	Describe the risk in business terms
Menu Label for Generated UI (*PD)	

14. Click the **Fields** tab to enter metadata for fields on the risk object. Use the guidance provided above for completing the Fields metadata.

15. Continue this process for all tabs and all parts of the product model.

Review product content

Before generating the product, review all product content.

Procedure

Review all screens and fields for the product and make sure that the information is correct and descriptive.

For example, if you have a **Children's UI Label** set to something such as **Risk Objects**, consider changing the name to something more appropriate for the product. Changing **Risk Objects** to **Vehicles** in an auto product would be more descriptive.

Validated fields

The following fields are validated during product generation:

Field	Description
Attributes	Name and Description are required. Typelist name must be unique; two typelists cannot have the same name.
Drop-down list entries	For entries in a drop-down list, Short Name and Description fields on the Drop-Down Definition popup are required. In coverage terms with Option and Package drop-down type, Value is required.
Clauses	For product model clauses, such as coverages, conditions, and exclusions, Short Name and Description are required.
Clause categories	Name and Description must be defined.
Risk objects	For risk objects, including the product line, Description , Menu Label for Generated UI (*PD) , and Type of Risk Object or Product Line Type are required.

Field	Description
Exposures	Description (*PD), Menu Label (*PD), and Type are required.
Short names	Short Name (*PD) and Typelist (*PD) must conform to the naming requirements for short names.

In the generated product:

- The **Code** field corresponds to a `CodeIdentifier` property
- The **Menu Label for Generated UI (*PD)** corresponds to a `MenuLabel` property
- The **Product Line Type** or **Type of Risk Object** corresponds to a `TypeName` property

Validate product in Designer mode

A business user working in **Designer** mode can validate the product prior to having the product generated by a user with **Developer** mode. In **Designer** mode, the business user can fix some but not all validation errors.

Before you begin

Set **Product Design Mode** to **Designer**.

Procedure

1. Click **Validation Status** to find validation errors.
2. Fix errors, and click **Validation Status** until you have fixed as many errors you can in **Designer** mode.

Next steps

- “Validate product in Developer mode” on page 106

Validate product in Developer mode

In **Developer** mode, the developer can validate the product and fix all validation errors.

Before you begin

- Set **Product Design Mode** to **Developer**.
- “Validate product in Designer mode” on page 106

Procedure

1. In PolicyCenter, go to the **Administration→Product Management** screen.
2. Select a product at the top of the **APD Managed** tab.
The selected product appears in the **Details** tab.
3. Click **Edit Product**.
PolicyCenter displays the **Product Definition** screen.
4. On the **Product Definition** screen, select **Generate Product Code** and select **Base Product**.
PolicyCenter validates the product model and displays the **Review Product Elements** screen showing the product hierarchy. If there are problems with the product definition, the **Complete Generation** button does not appear in the upper right corner of the screen. Problems are marked with a red X inside a circle in the second column of the table. A red X indicates that there is a error in the current element or one of its children. The element with the error displays an error message. Click > to display the error.
5. If there are problems, click **Cancel Generation**, and fix the problems in the **Product Definition** screens, then **Generate Product Code**.

You cannot proceed until all problems are resolved. There are no problems in the product definition when a check mark appears in the second column of each row in the **Review Product Elements** table. A **Complete Generation** button appears next to **Cancel Generation**.

6. (Recommended) Export your product to template by clicking **Edit** and selecting **Export Template** before generating the product code. You can use this template to make further product changes using the **Product Definition** screens.
7. (Recommended) Check the template into version control. With version control, you can compare versions, and restore previous versions, if necessary.

Note: If you drop the database before the final build is complete, you will lose the product changes you have made.

Next steps

- “Move product” on page 107

Move product

If the product is not in the PolicyCenter instance where it will be generated, deployed, and tested, move it to that instance.

Before you begin

- Set **Product Design Mode** to **Designer** or above.
- “Validate product in Developer mode” on page 106

About this task

Procedure

1. In PolicyCenter, go to the **Administration**→**Product Management** screen.
2. Select the product on the **APD Managed** tab.
3. On the **Details** tab, select **Export Template**.
The product template is saved on the computer where PolicyCenter is installed.
4. In the PolicyCenter instance where you will generate and deploy the product, go to the **Administration**→**Product Management** screen.
5. In the **APD Managed** tab, select **Import From Template** and import the template.

Next steps

- “Generate and deploy the product” on page 107

Generate and deploy the product

Before you begin

- Set **Product Design Mode** to **Developer**.
- Set the package name for the product line Gosu files in `APDProductLinesGosuPackage` script parameter. See “Define Gosu package for product line files” on page 17.
- “Move product” on page 107

About this task

You generate the product in the PolicyCenter instance where you will continue product development. Product generation accesses and modifies files in the PolicyCenter configuration, therefore generation must be done in a local development environment and not a hosted instance of PolicyCenter.

After generation, check files into version control, recompile PolicyCenter in Guidewire Studio, and deploy the product.

Procedure

1. In PolicyCenter, select **Generate Product Code**, and select **All**.

At this point, you have fixed all validation errors.

2. Click **Complete Generation**.

The generated product files are saved into the PolicyCenter directory on the computer where PolicyCenter is running.

3. After the product code is generated, stop the PolicyCenter application.

4. Check your changes into version control.

5. In Studio, recompile PolicyCenter.

6. Restart PolicyCenter.

- Now you can select the product from the **Installed Products** tab on the **New Submissions** screen. You can make modifications to the product in Product Designer.
- If you do not drop the database, the product still appears on the **Visualized Products** tab and on the **Product Definition** screen.

Generate Product Code

From the **Generate Product Code** menu, you can select:

All

Generate everything.

Base Product

Generate PolicyCenter product artifacts only. These are the data model, product model, PCF files, and supporting Gosu code.

Extensions

Generate additional registered extensions only.

After generating product code

In the generated product, you can enhance the product using the development tools provided by PolicyCenter, such as Guidewire Studio. You can create underwriting rules, rating, and forms management. You can integrate with other applications, and make other changes to meet your needs. You incorporate changes made to the visualized product by regenerating it and merging changes.

Loading editions into generated product

Editions contain the product model properties and subclause (parent-child clause) relationships. After generating the product, you can load editions into the product running in PolicyCenter.

- You create editions in the visualized product line.
- The product template contains the editions.
- The editions are stored in the PolicyCenter database after loading the template for the generated product.
- Subclause relationships are in the base edition.

Note: Generating the product creates the artifacts needed for PolicyCenter including PCF files, the product model, and entity model. But generating the product does not include editions.

After loading editions, you activate the editions to make them active in the generated product. The rules take effect immediately after activation. Rules in the new editions may cause validation issues to appear or disappear from open policy transactions. Validation issues can even appear or disappear from quoted policy transactions.

See also

- “Editions for product lines” on page 79

Editions are deployed into a running PolicyCenter instance

You load editions into a running PolicyCenter instance.

If you make product changes that are completely contained within editions, the changes are not to the generated product. Therefore, you can load the editions into the product without first regenerating and redeploying it.

Changes that are completely contained within editions include:

- Changes to product model properties
 - Field usage
 - Clause availability and existence
 - Default, minimum, or maximum values
- Changes to subclause relationships that do not include structural changes
 - For example, you add or remove subclause relationships, but you do not add or remove any clauses or clause terms.

Structural changes require regenerating the product

Modifying editions can introduce structural changes to the product line.

These kinds of changes include but are not limited to:

- Adding a new risk object, clause, attribute, or other structural element.
- Adding a new clause, such as a coverage, and making it available starting on September 1st for all policies.
- Changing the label of an attribute or a clause. For example, changing the wording of a question attribute.
- Changing the name of a risk object or clause.

Adding or removing subclause relationships, without adding or removing clauses or clause terms, is not a structural change.

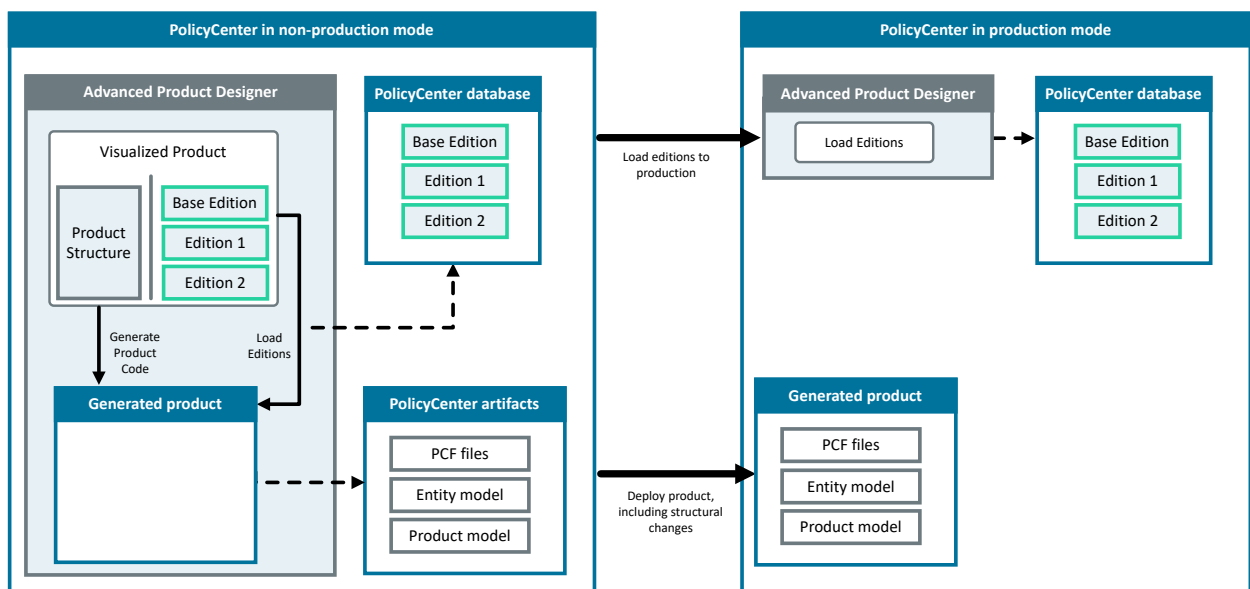
Before loading editions with structural changes, you must regenerate and redeploy the product in PolicyCenter.

Editions high-level workflow

These high-level steps show how editions move from a non-production PolicyCenter development environment to PolicyCenter in production mode.

About this task

Editions are created on the product line. You create editions in the visualized product in a non-production environment. Editions, including the **Base** edition that defines the base product line, contain the product model properties and subclause relationships.



Procedure

1. In a non-production environment of PolicyCenter, develop the product structure and editions.
The editions group product model properties by effective date. The Base edition contains product model properties and subclause (parent-child clause) relationships without an effective date.
2. If necessary, generate the product code in the non-production environment.
Generating the product code is necessary the very first time a product is created. After that, you need to generate when you make structural changes. For more information, see “Structural changes apply to all editions” on page 80.
PolicyCenter artifacts, such as PCF files, the entity model, and the product model, are created. Use Guidewire development tools, such as Studio, to work with these files.
Editions are not included in the generated product.
3. If you generated the product code in the previous step, recompile and restart PolicyCenter.
4. For testing, load editions into the generated product in the non-production environment.
 - a. Export the template from the visualized product.
 - b. Load the editions from the template into the generated product.
The editions are stored in the PolicyCenter database.
5. In a production environment of PolicyCenter, deploy the product code.
This step is necessary if you made structural changes to the product.
This step requires recompiling PolicyCenter.
6. Load editions to production PolicyCenter.
 - a. Export the template containing editions from non-production PolicyCenter.
 - b. Load the editions from the template into production PolicyCenter.
The editions are stored in the PolicyCenter database.

Load and activate editions in generated product

Before you begin

- Set **Product Design Mode** to **Designer** or above.

About this task

Editions are loaded from the product template into a generated product. Loading editions is a two step process:

1. Preload editions – Editions are loaded, but are not yet used in policy transactions.
2. Activate editions – Editions are active and are used in policy transactions.
 - In a production server, you cannot activate preloaded editions if there are issues requiring attention in the product template.
 - In a development or test server, you can choose to activate preloaded editions even if the product template contains issues requiring attention. You can control this with the `APDBlockInvalidEditionsInDev` script parameter.

After loading editions, you activate the editions to make them active in the generated product. The rules take effect immediately after activation. Rules in the new editions may cause validation issues to appear or disappear from open policy transactions. Validation issues can even appear or disappear from quoted policy transactions.

Procedure

1. Generate the product code if:
 - If this is the very first time the product is created
 - If you made structural changes in an edition
 Generating product code require that you set **Product Design Mode** to **Developer**.

2. If you generated the product code in the previous step, recompile and restart PolicyCenter.
3. Export the template for the visualized product.
 - a. In PolicyCenter, go to the **Administration**→**Product Management** screen.
 - b. Select the product on the **APD Managed** tab.
 - c. On the **Details** tab, select **Export Template**.
The product template is saved to your computer.
4. Log in to PolicyCenter where you want to load editions for this product.
This can be the same PolicyCenter from which you exported the template, or another PolicyCenter instance that has the generated product.
5. In the **Installed Editions** tab, load the template.
 - a. In PolicyCenter, go to the **Administration**→**Product Management** screen.
 - b. Select the product on the **APD Managed** tab.
 - c. Select the **Installed Editions** tab.
In a non-production server, this tab is visible only for generated products.
 - d. Click **Preload Editions** to preload the editions in the template.
Preloading editions removes any previously existing preloaded editions. New editions are not merged with the existing ones.
 - e. Click **Activate**.
When you activate editions, previously active editions are removed.
In a non-production server, the behavior of the **Activate** button is based on the `APDBlockInvalidEditionsInDev` script parameter.
 - When `false`, the **Activate** button is enabled, even when there are preloaded editions containing issues. A list of invalid rules is displayed in the **Installed Editions** tab. This is the default value.
 - When `true`, the **Activate** button is disabled, and you cannot activate editions with issues. Fix the issues and re-import the product template.
In a production server, the **Activate** button is disabled if there are issues in the product template. Fix the issues and re-import the product template.
The editions are active and are used in policy transactions.

Activating editions in development or test mode

When you are working on a server in development or test mode, you can activate editions even if the editions contain issues that require attention. By default, PolicyCenter is more lenient when you are working in these modes, and does not require that you fix all issues until you are ready. PolicyCenter displays messages when it encounters these issues. These messages appear on **Installed Editions** tab displays in **Requires Attention - Preloaded** and **Requires Attention - Activated** sections at the bottom of the screen.

You can run policy transactions, such as submissions, with either the visualized or installed versions of the product. When the policy transaction encounters an issue, it outputs a message to the log file. There is no message in the user interface.

About issues requiring attention

Issues can occur when there are multiple people working on a product, and the editions become out of sync with the generated product.

For example, you add a new field to the visualized product, referencing it in a product model property. This causes an issue when you preload the edition because the new field does not exist in the generated (installed) product. These are the steps to produce this kind of an issue:

1. Generate a visualized product.
2. Add a new field to a visualized product. For example, add a text field that asks, “Do you have more than two dogs?”

3. Create a condition on a clause that references the field. For example, make Flea and Tick coverage not available if you answer **Yes** to the question.
4. Export the visualized product as a template.
5. On the **Installed Editions** tab, click **Preload Editions** and select the template.
6. Even though an issue was detected, you can activate preloaded editions.

If you add a new field to the visualized product but do not reference it in a product model property, an issue is not raised when you preload editions.

Disabling the Activate button

You may choose to work in a more restrictive mode, where the **Activate** button is disabled if there are issues requiring attention in preloaded editions. This may be useful when you have completed developing editions and want to resolve all issues.

To disable the **Activate** button, go to **Administration**→**Utilities**→**Script Parameters**. Change the `APDBlockInvalidEditionsInDev` parameter to `true`. In the base configuration, this parameter is `false`. The **Activate** button is disabled if preloaded editions contain issues.

If there is already an invalid activated edition before you change the parameter from `false` to `true`, PolicyCenter throws an exception when it executes an existing invalid rule.

In production mode, this parameter is ignored. In this mode you cannot activate preloaded editions with issues.

In production mode, structural changes in a product can cause existing, formerly valid, properties in an already-activated edition to become invalid. PolicyCenter throws an exception when it executes an invalid rule.

Changing visualized product after code generation

After completing product code generation, if you make deletions in the visualized product then regenerate the product, the deletions are not removed from the generated product. You must delete them manually in Product Designer or Guidewire Studio. These deletions could be the result of reimporting the mind map or template, or of deleting something from the product in Advanced Product Designer.

Regenerating the product

After generating your product, you can continue product development in Advanced Product Designer while simultaneously making changes to the product in Guidewire Studio and Product Designer.

If a feature is provided in Advanced Product Designer, then always use Advanced Product Designer to develop the feature. Do not use Product Designer or Studio to add product features that Advanced Product Designer provides. See “When possible, make changes in Advanced Product Designer” on page 14.

To update the generated product, you need to regenerate it. When you click **Generate Product Code**, Advanced Product Designer displays a warning that lists the modified files that will be overwritten if you proceed. Before completing the generation, make a copy of these files so that you can merge your changes after regenerating.

For each type of file, the following table shows whether you are warned before regenerating.

Type of files	Warn ?	Description
“System files” on page 115		No warning if file is modified.
“Resources” on page 115		No warning if file is modified. The files <code>yyycostdef.xml</code> , <code>XXXDiffTree.xml</code> , and <code>YYDiffTree.xml</code> are exceptions.
“Product model” on page 116		No warning if file is modified.
“Policy line patterns” on page 117		No warning if file is modified.

Type of files	Warn ?	Description
"Risk object entities" on page 119	•	If you can, make all changes to the object in Advanced Product Designer. When you regenerate the product, you can accept the regenerated version of the object, and do not need to merge it with the older version. If that is not possible, you can make changes in an extension to the object. You have to add the extension manually in the file system. Although you cannot add the extension in Studio, you can edit it in Studio after creating it.
"Clause entities" on page 120	•	
"Schedule entities" on page 120	•	
"Typelists" on page 121	•	For each typelist, maintain it either in a particular Advanced Product Designer product or in Studio. Do not modify the typelist in more than one place.
"Display names" on page 121	•	Make all changes to display strings in Advanced Product Designer so that there is no need to merge changes when you regenerate.
"PCF files" on page 121	•	Save a copy before generating, and merge changes.
"Gosu classes" on page 122	•	When you generate the product, Advanced Product Designer creates a base class and an implementation class in the <i>PolicyCenter_installation/modules/configuration/gsrc</i> directory. The base class is created in the <i>gw.lob</i> package and has the suffix <i>Base</i>. The implementation class is created in the package defined by the <i>APDProductLinesGosuPackage</i> script parameter. The default is <i>ext.lob</i>. To avoid triggering warnings when you regenerate the product, make your changes in the implementation class, and do not modify the base class. You will receive warnings for changes to base classes, but no warnings for changes to implementation classes. When you regenerate, the implementation class is not affected; it is only generated if it does not exist. For enhancement classes, <i>YYYRiskEnhancement.gs</i> and <i>YYYExposureEnhancement.gs</i>, make your changes directly in the enhancement and merge changes.

Note: Some files are generated only once. You can make changes, and your changes are not overwritten when you regenerate. These exceptions are noted in the topics describing the file types.

Merging changes in generated product

When merging changes to files, do a three-way merge between:

- File as originally generated by Advanced Product Designer
- Modifications that you made to the file
- File as generated this time by Advanced Product Designer

Generated product files

Generating a product creates files in the *PolicyCenter_installation/modules/configuration/config* directory of PolicyCenter.

The files are listed below. The following variables are used in the file names:

- *XXX* and *xxx* – The three character product abbreviation in upper and lower case
- *YYY* and *yyy* – The three character product line abbreviation in upper and lower case
- *ProdId* – Product identifier
- *LineId* – Line identifier
- *Line* – Short name of the product line
- *Clause* – Short name of the clause
- *Typelist* – Short name of the typelist

- *Risk* – Short name of the risk object

System files

Information about the product line appears at the end of the following files. Paths are relative to *PolicyCenter_installation/modules/configuration/config*.

When regenerating the product, you are not warned about modified files.

Subdirectory	File name	Description
locale	display.properties	Display labels for screens. Generates labels for all languages defined by the product. In Studio, if you add a display label that is not generated, the display key remains intact after regenerating the product.
	productmodel.display.properties	Display labels for user interface fields for all languages defined by the product.
	typelist.properties	For all languages defined by the product: - TypeKey.InstalledPolicyLine.XXX – Type key for policy line. - TypeKeyDescription.InstalledPolicyLine.XXX – Type key for description.
lookuptables	lookuptables.xml	Lookup table for all product model patterns.

Resources

Product generation adds files to the resources directory under *PolicyCenter_installation/modules/configuration/config*.

When regenerating the product, you are not warned about modified files. The files *yyycostdef.xml*, *XXXDiffTree.xml*, and *YYDiffTree.xml* are exceptions.

Subdirectory	Directory or File name	Description
	systables.xml	Cost definition for the product line: <pre><FileDefinition Name="XXXcostdef.xml" Priority="0"> <Entity Type="XXXCostDefinition"/> </FileDefinition></pre>
diff	XXXDiffTree.xml	Before regenerating the product, you will receive a warning about modified files that will be overwritten if you proceed.
productmodel	policylinepatterns/ <i>LineId</i>	
	products/ <i>ProdId</i>	
systables	costcode_ext.xml	

Subdirectory	Directory or File name	Description
	yyycostdef.xml	Before regenerating the product, you will receive a warning about modified files that will be overwritten if you proceed.
	underwriting_companies.xml	For each jurisdiction: <pre><LicensedState public- id="ProdId:Jurisdiction"></pre>

Product model

Product model files are in the resources/productmodel/products directory under *PolicyCenter_installation/modules/configuration/config*.

When regenerating the product, you are not warned about modified files.

File name	Description
<i>ProdId/ProdId.xml</i>	These files are generated once. You can make changes, and your changes will not be overwritten when you regenerate.
<i>ProdId/ProdId-lookups.xml</i>	

Update availability in Product Designer

If possible update availability in Advanced Product Designer. If Advanced Product Designer does not support a particular type of availability, you can make updates directly in Product Designer.

The first column of the table describes the update you make in Product Designer. The second column describes what happens in Studio when you commit the change in Product Designer. The third column describes what happens when you regenerate the product in Advanced Product Designer.

Product Designer	Studio	After regenerating product
Availability on Product	Availability criteria is updated in <i>ProdId/ProdId-lookups.xml</i> . Availability criteria for a jurisdiction is updated in <i>ProdId-lookups.xml</i> in the <i>ProdId/jurisdictions/XX/coveragepatterns</i> directory, where <i>XX</i> is the jurisdiction.	Updates in all <i>ProdId-lookups.xml</i> files remain intact.

Update offerings in Product Designer

Advanced Product Designer does not support offerings. You can specify offerings for your product in Product Designer.

Product Designer	Studio	After regenerating product
Add offering	<i>XXXOffering.xml</i> and <i>XXXOffering-lookups.xml</i> created in the <i>ProdId/offerings</i> directory, where <i>XXX</i> is the name of the offering.	<i>XXXOffering.xml</i> and <i>XXXOffering-lookups.xml</i> files remain intact. Overwritten – Offering disappears from <i>productmodel.display.properties</i> without warning.

Update modifiers in Product Designer

Advanced Product Designer does not support modifiers. You can specify modifiers for your product in Product Designer.

The first column of the table describes the update you make in Product Designer. The second column describes what happens in Studio when you commit the change in Product Designer. The third column describes what happens when you regenerate the product in Advanced Product Designer.

Product Designer	Studio	After regenerating product
Add modifier	<i>ProdId.xml</i> and <i>ProdId-lookups.xml</i> created in the <i>ProdId</i> directory.	Overwritten – Modifier disappears from both files without warning. Overwritten – Modifier disappears from <i>productmodel.display.properties</i> without warning.

Update question sets in Product Designer

Advanced Product Designer does not support question sets. You can specify question sets for your product in Product Designer.

The first column of the table describes the update you make in Product Designer. The second column describes what happens in Studio when you commit the change in Product Designer. The third column describes what happens when you regenerate the product in Advanced Product Designer.

Product Designer	Studio	After regenerating product
Add question set with question	Question set added to <i>ProdId/ProdId.xml</i> . <i>QuestionSetName.xml</i> and <i>QuestionSetName-lookups.xml</i> created in <i>productmodel/questionsets</i> .	Files remain intact.

Policy line patterns

Policy line patterns are in *resources/productmodel/policylinepatterns/Line* under *PolicyCenter_installation/modules/configuration/config*. The policy line corresponds to the product line in Advanced Product Designer.

When regenerating the product, you are not warned about modified files.

File name	Description
<i>Line.xml</i>	These files are generated once. You can make changes, and your changes will not be overwritten when you regenerate.
<i>Line.properties</i>	
<i>Line-lookups.xml</i>	

Update categories in Product Designer

Guidewire recommends that you update categories in Advanced Product Designer not Product Designer. In Advanced Product Designer, these are *clause categories*. This topic describes the consequences of updating categories in Product Designer.

The first column of the table describes the update you make in Product Designer. The second column describes what happens in Studio when you commit the change in Product Designer. The third column describes what happens when you regenerate the product in Advanced Product Designer.

Product Designer	Studio	After regenerating product
Add new category	Category is added to <i>Line.xml</i> in the <i>policylinepatterns</i> directory. Display string for category is added to <i>productmodel.display.properties</i> .	Overwritten – Category disappears from both files without warning.
Modify category	Category is modified in <i>Line.xml</i> in the <i>policylinepatterns</i> directory. Display string for category is modified in <i>productmodel.display.properties</i> .	Overwritten – Your modifications to both files disappear without warning.
Remove category	Category is removed from <i>Line.xml</i> in the <i>policylinepatterns</i> directory. Display string for category is removed from <i>productmodel.display.properties</i> .	Overwritten – The removed category reappears in both files without warning.

Update clauses in Product Designer

Guidewire recommends that you update clauses in Advanced Product Designer not Product Designer. This topic describes the consequences of updating clauses in Product Designer.

Clause patterns are in *resources/productmodel/policylinepatterns/Line/coveragepatterns*. This directory includes patterns for coverages, conditions, and exclusions.

File name
<i>YYYClause.xml</i>
<i>YYYClause-lookups.xml</i>

The first column of the table describes the update you make in Product Designer. The second column describes what happens in Studio when you commit the change in Product Designer. The third column describes what happens when you regenerate the product in Advanced Product Designer.

Product Designer	Studio	After regenerating product
Add clause	<i>YYYClause.xml</i> and <i>YYYClause-lookups.xml</i> files are added in the <i>coveragepatterns</i> directory. Display string for clause is added to <i>productmodel.display.properties</i> .	<i>YYYClause.xml</i> and <i>YYYClause-lookups.xml</i> files remain intact. Overwritten – Clause disappears from <i>productmodel.display.properties</i> without warning.
Remove clause	<i>YYYClause.xml</i> and <i>YYYClause-lookups.xml</i> files are removed from the <i>coveragepatterns</i> directory. Display string for clause is removed from <i>productmodel.display.properties</i> .	Overwritten – <i>YYYClause.xml</i> and <i>YYYClause-lookups.xml</i> files reappear after generate without warning. Overwritten – Clause reappears in <i>productmodel.display.properties</i> without warning.
Add term to existing clause	Term is added to <i>YYYClause.xml</i> and <i>YYYClause-lookups.xml</i> files in the <i>coveragepatterns</i> directory. Display string for term is added to <i>productmodel.display.properties</i> .	Overwritten – Term disappears from <i>YYYClause.xml</i> without warning. Term remains in <i>YYYClause-lookups.xml</i> . Overwritten – Clause disappears from <i>productmodel.display.properties</i> without warning.
Remove term from clause	Term is removed from <i>YYYClause.xml</i> and <i>YYYClause-lookups.xml</i> files in the <i>coveragepatterns</i> directory.	Overwritten – Term reappears in <i>YYYClause.xml</i> and <i>YYYClause-lookups.xml</i> without warning.

Product Designer	Studio	After regenerating product
	Display string for term is removed from <code>productmodel.display.properties</code> .	Overwritten – Term reappears in <code>productmodel.display.properties</code> without warning.

Update availability in Product Designer

If possible, define availability for the product in Advanced Product Designer. If you have requirements for usage and availability that cannot be specified using Advanced Product Designer, you can use the availability features that Product Designer offers. For more information, see “Specifying additional availability in Product Designer” on page 72.

Availability scripts, lookup tables, and grandfathering are not supported in Advanced Product Designer. You can enter these for your product line in Product Designer.

The first column of the table describes the update you make in Product Designer. The second column describes what happens in Studio when you commit the change in Product Designer. The third column describes what happens when you regenerate the product in Advanced Product Designer.

Product Designer	Studio	After regenerating product
Lookup table (availability criteria that you define at the top of the Availability screen).	Availability criteria is updated in <code>YYYClause-lookups.xml</code> in the <code>coveragepatterns</code> directory. Availability criteria for a jurisdiction is updated in <code>YYYClause-lookups.xml</code> in the <code>Line/jurisdictions/XX/coveragepatterns</code> directory, where <code>XX</code> is the jurisdiction.	Updates in all <code>YYYClause-lookups.xml</code> files remain intact.
Availability script	Script is updated in <code>YYYClause.xml</code> in the <code>coveragepatterns</code> directory.	For clauses, availability script updates remain in <code>YYYClause.xml</code> . Overwritten – Availability script is overwritten in clause terms. You are warned before the file is overwritten.
Grandfathering (randomly choose if covered object at line or risk)	Grandfather criteria is updated in <code>YYYClause-grandfathering.xml</code> in the <code>Line/jurisdictions/XX/coveragepatterns</code> directory, where <code>XX</code> is the jurisdiction.	Updates in <code>YYYClause-grandfathering.xml</code> remains intact.

Risk object entities

Entities for risk objects are in `extensions/entity` directory under `PolicyCenter_installation/modules/configuration/config`.

Before regenerating the product, you will receive a warning about modified files that will be overwritten if you proceed.

If you can, make all changes to the object in Advanced Product Designer. When you regenerate the product, you can accept the regenerated version of the object, and do not need to merge it with the older version. If that is not possible, you can make changes in an extension to the object. You have to add the extension manually in the file system. Although you cannot add the extension in Studio, you can edit it in Studio after creating it.

File name	Description
<code>YYCost.eti</code>	For each product line, there is a cost on the product line. These files are generated once. You can make changes, and your changes will not be overwritten when you regenerate.

File name	Description
YYCostDefinition.eti	
YYTransaction.eti	These files are generated once. You can make changes, and your changes will not be overwritten when you regenerate.
YYRisk.eti	An entity for each risk object, including the product line. These files are generated once. You can make changes, and your changes will not be overwritten when you regenerate.
YYRiskCost.eti	A cost entity for each risk object.
YYExposure.eti	An entity for each exposure.
YYLineId.eti	An entity for the product line.

Clause entities

Entities for clauses are in the extensions/entity directory under *PolicyCenter_installation/modules/configuration/config*. Clause types are coverages, conditions, and exclusions.

Before regenerating the product, you will receive a warning about modified files that will be overwritten if you proceed.

If you can, make all changes to the object in Advanced Product Designer. When you regenerate the product, you can accept the regenerated version of the object, and do not need to merge it with the older version. If that is not possible, you can make changes in an extension to the object. You have to add the extension manually in the file system. Although you cannot add the extension in Studio, you can edit it in Studio after creating it.

File name	Description
YYRiskCov.eti	An entity for each clause category on the risk object.
YYRiskCond.eti	
YYRiskExcl.eti	
YYRiskCovCost.eti	A cost entity for each clause type.
YYRiskCovLookup.eti	These files are generated once. You can make changes, and your changes will not be overwritten when you regenerate.
YYRiskCondLookup.eti	
YYRiskExclLookup.eti	
YYRiskOptLookup.eti	
YYRiskPackLookup.eti	
YYRiskTermLookup.eti	

Schedule entities

Entities for schedules are in the extensions/entity directory under *PolicyCenter_installation/modules/configuration/config*.

Before regenerating the product, you will receive a warning about modified files that will be overwritten if you proceed.

If you can, make all changes to the object in Advanced Product Designer. When you regenerate the product, you can accept the regenerated version of the object, and do not need to merge it with the older version. If that is not possible, you can make changes in an extension to the object. You have to add the extension manually in the file system. Although you cannot add the extension in Studio, you can edit it in Studio after creating it.

File name
YYYRiskSchCovItemCov.eti
YYYRiskScheduleCov.eti
YYYRiskScheduleCovItem.eti
YYYScheduledItem.eti

Display names

Language display strings for each entity are in the `displaynames` directory under *PolicyCenter_installation/modules/configuration/config*. The file names are:

- *YYYEntity.Lang-code*

For example, `PETDogCov.en` or `PETDogCov.fr`.

Before regenerating the product, you will receive a warning about modified files that will be overwritten if you proceed.

Make all changes to display strings in Advanced Product Designer so that there is no need to merge changes when you regenerate.

In Studio, if you add a display name that is not generated, the display name remains intact after regenerating the product.

Costs and transactions

The following files are generated once. You can make changes, and your changes will not be overwritten when you regenerate.

- *YYYCost.Lang-code*
- *YYYTransactionLang-code*

Typelists

Typelists in the `extensions/typelist` directory under *PolicyCenter_installation/modules/configuration/config*.

Before regenerating the product, you will receive a warning about modified files that will be overwritten if you proceed.

For each typelist, maintain it either in a particular Advanced Product Designer product or in Studio. Do not modify the typelist in more than one place.

File name	Description
<i>Typelist.tti</i>	Typelist for each drop-down list defined by the product line.
<i>InstalledPolicyLine.YYY.ttx</i>	Adds this policy line to the <code>InstalledPolicyLine</code> typelist.

PCF files

Subdirectories for PCF files are in the `web/pcf/line`, `xxx` and `yyy` directories. Paths are relative to *PolicyCenter_installation/modules/configuration/config*.

Before regenerating the product, you will receive a warning about modified files that will be overwritten if you proceed.

PCF files that you add are not overwritten when you regenerate the product.

For the product, the directory contains the following file:

Subdirectory	File name
policyfile	PolicyMenuItemSet.ProdId.pcf

For the product line, the directory contains the following files:

Subdirectory	File name
job	LineWizardStepSet.ProdId.pcf
policy	YYYLineIdPanelSet.pcf
	YYYLineIdScreen.pcf
policyfile	PolicyFile_YYYLineId.pcf
	LineIdLinks.pcf

For each risk object, the directory contains the following files:

Subdirectory	File name
policy	YYYRiskListPanelSet.pcf
	YYYRiskPanelSet.pcf
	YYYRiskPopup.pcf
	YYYRiskScreen.pcf
policyfile	PolicyFile_YYYRisk.pcf
	PolicyFile_YYYRiskScreen.pcf

For each exposure, the directory contains the following files:

Subdirectory	File name
policy	YYYExposureListPanelSet.pcf

Gosu classes

When you generate the product, Advanced Product Designer creates a base class and an implementation class in the *PolicyCenter_installation/modules/configuration/gsrc* directory. The base class is created in the *gw.lob* package and has the suffix *Base*. The implementation class is created in the package defined by the *APDProductLinesGosuPackage* script parameter. The default is *ext.lob*.

To avoid triggering warnings when you regenerate the product, make your changes in the implementation class, and do not modify the base class. You will receive warnings for changes to base classes, but no warnings for changes to implementation classes. When you regenerate, the implementation class is not affected; it is only generated if it does not exist.

For enhancement classes, *YYYRiskEnhancement.gs* and *YYYExposureEnhancement.gs*, make your changes directly in the enhancement and merge changes.

See also

- “Define Gosu package for product line files” on page 17

Displaying Risk Details in generated product

In the generated product, the **Risk Details** screen may display differently than in the visualized product.

In the generated product, the **Risk Details** screen layout uses the following rules for displaying risk objects, using the first match:

Condition	Display
Risk object has more than eight fields	Display fields in two columns.
Risk object has eight or fewer fields	
Only one type of child risk object	Display fields in two columns and list risk objects underneath the fields.
More than one type of child risk object or no risk objects, and exposure details are defined	Display a list of exposures underneath the fields. For example, display a list of exposures in a Workers' Compensation policy.
No exposures or child risk object	In first column, display fields. In second column, display a list of coverages.
No fields	Depending upon what exists, display: • Risk objects if there is just a single child risk object type, or • Exposures, or • Coverages

Risk objects and coverables

In the generated product, risk objects with coverages become **Coverable** entities. If a risk object has no coverages, then it is not a **Coverable**.

Upgrade generated APD products

After upgrading to a new PolicyCenter release, you must upgrade all APD products by regenerating them. This task is unnecessary if you have not yet generated the product.

Before you begin

1. Upgrade PolicyCenter.
2. Set **Product Design Mode** to **Developer**.

Procedure

1. In PolicyCenter, navigate to **Products**→**Product Management**, and select the product you wish to generate.
2. Generate the product code by selecting the product and clicking **Generate Product Code**.
3. Merge code changes in Bitbucket.

See

- “Regenerating the product” on page 113
- “Generated product files” on page 114

Benefits of APD for externally-managed products

If you have *externally-managed products*, there are several benefits that Advanced Product Designer provides. Externally-managed products are products not developed with Advanced Product Designer, such as Personal Auto in the PolicyCenter base configuration, or products you developed prior to Advanced Product Designer.

The benefits that Advanced Product Designer provides include:

- “View a visualized version of your product” on page 127
- “Generate EDGE APIs for externally-managed products” on page 128
-

View a visualized version of your product

About this task

Generate a visualized version of your product. With the visualized product, you can:

- View the product structure and attributes in Advanced Product Designer by importing the template
- Run policy transactions using the visualized product
- View the product as a mind map by converting it to XMind format

You continue to manage the externally-managed product as before using:

- Guidewire Rating Management
- Underwriting authority provided in PolicyCenter
- Product Designer
- Guidewire Studio

Procedure

1. Export the product model to an Advanced Product Designer template. See “Extract APD representation of externally managed product” on page 129.
2. Import the template into Advanced Product Designer. See “Create visualized product from template” on page 43.

3. If you wish to view the product as a mind map, generate a mind map. See “Generate mind map from product” on page 39.

Generate EDGE APIs for externally-managed products

About this task

Generate product-specific EDGE APIs for use with Digital applications.

Note: At each step in this process, manual work may be required to successfully complete the next step.

You continue to manage the externally-managed product as before using:

- Guidewire Rating Management
- Underwriting authority provided in PolicyCenter
- Product Designer
- Guidewire Studio

Procedure

1. Export the product model to an Advanced Product Designer template. See “Extract APD representation of externally managed product” on page 129.
2. Import the template into Advanced Product Designer. See “Create visualized product from template” on page 43.
3. Generate the EDGE APIs. See the *Digital Code Generation Guide* for more information.

Migrate externally-managed product to Advanced Product Designer

About this task

Migrate your product to Advanced Product Designer, and manage and update it as any other visualized product. With migration, you replace the original product with the one created using Advanced Product Designer.

If you wish to migrate, you can bring the product into Advanced Product Designer and optionally create the mind map.

Note: The APD product is a different product than the original externally-managed product. If you wish to retire the externally-managed product, you can migrate in-force policies of the externally-managed product over to the APD product upon renewals.

Procedure

1. Export the product model to an Advanced Product Designer template. See “Extract APD representation of externally managed product” on page 129.
2. Import the template into Advanced Product Designer. See “Create visualized product from template” on page 43.
3. If you wish to view the product as a mind map, generate a mind map. See “Generate mind map from product” on page 39.

4. Change the following product attributes so as not to interfere with the externally managed product during development:
 - Name
 - Description
 - Identifier
 - Abbreviation
5. Change the following product line attributes so as not to interfere with the externally managed product during development:
 - Name
 - Description
 - Line Prefix
 - Identifier
6. Review and modify all field names in the visualized product. Label and field names may be truncated and require your attention. See “Finalizing and generating the product” on page 101, paying special attention to “Metadata” on page 102, “Review product content” on page 105, and “Validated fields” on page 105.
7. Reimplement product functionality that is not supported by Advanced Product Designer and is not in the visualized product. You may be able to reuse code from the externally-managed product.

To retire the externally-managed product (optional)

8. After you complete development and testing, you can remove the original externally-managed product, update the product and product line attributes to match the original product, and install the Advanced Product Designer product in its place.
9. When the APD product is in production, migrate in-force policies written against the original product over to the APD product upon renewal.

Extract APD representation of externally managed product

You can export products that appear on the **Product Management**→**Externally Managed** tab from PolicyCenter to template format, then create a visualized version of the product.

Procedure

1. Navigate to the **Externally Managed** tab on the **Product Management** screen.
2. Select **Extract APD Representation**.
This saves the product as template.
3. Navigate to the **APD Managed** tab of the **Product Management** screen.
4. Select **Import From Template** and choose the template you just generated.
This creates an APD representation of the product.

Example mind maps

Advanced Product Designer includes mind maps and other resources that you can use as examples.

Guidewire markers for XMind

Map	Description
GWResourceBundle.xrb	Resource bundle file that contains Guidewire markers for tagging the product model. You can import these markers into XMind. See “Add Guidewire markers to XMind” on page 26.

Mind maps

The following mind maps of insurance products are provided as examples. The mind maps are in XMind format.

Map	Description
CommercialProperty.xmind	Commercial Property product
CommercialPropertyPackage.xmind	Commercial Package Policy product containing the following product lines: • Commercial Property • Crime • General Liability • Inland Marine
FocusedBusiness.xmind	Focused Business product for small businesses
GeneralLiability.xmind	General Liability product
Home_Generic_UK.mind	Homeowners product for the United Kingdom
MarineCargo.xmind	Marine Cargo product

Advanced Product Designer glossary

attribute

When writing a policy, you may require specific information or answers to question about parts of the policy. You can model this information using attributes. For example, in a small business policy, you need to know whether the insured has been refused insurance in the past. On a building, you may need to know the number of stories. Attributes can be attached to various parts of the product, such as the product line, risk objects, and exposures. Attributes can impact the pricing of a policy and drive underwriting decisions.

basis-scalable

A basis-scalable object means that the ratable basis for the object already considers time in some way. For example, it might capture the total payroll between two dates. Those costs are rated with a calculated adjusted rate multiplied by the basis in question, just like rate-scalable costs. Notable basis-scalable objects include workers' compensation exposures and some general liability exposures.

category

Groups of clauses (policy coverages, exclusions, and conditions). In PolicyCenter, you can group clauses into categories.

class code

Businesses that perform similar tasks are prone to similar risks. Class codes differentiate between the relative riskiness of different types of activities and are a way for an insurer to assess and rate risk. They also imply a unit of measurement for the risk (the basis) and a rating factor. Class codes are usually defined by industry organizations such as ISO or by regulators. Class codes only apply to commercial lines, and not all jurisdictions use class codes.

clause

A generic term for policy coverages, conditions, and exclusions.

clause term

A value that specifies the extent, degree, or an attribute of a clause. Clause terms are often limits or deductibles. A clause can have zero, one, or many clause terms.

clause term option

One of a list of values pertaining to a specific clause term.

condition

A contractual obligation defined in the insurance contract. Conditions neither provide nor exclude coverage. Conditions vary from simple and obligatory such as, “You will pay your bill,” to complex concepts such as retrospective rating in workers’ compensation policies.

coverable

See “risk object” on page 136.

coverage

Protection from a specific risk.

coverage category

Use categories to group similar coverages.

exclusion

Causes of loss that are explicitly not covered by the policy, so that the insurer has no exposure to claims in those areas.

exposure

An exposure represents a potential for accident or loss due to a risk, such as lawsuits or hazardous materials. Exposures quantify risk, typically by entering class codes and a basis amount. Insurance premiums are calculated by applying rates to the exposure basis such as payroll (as in worker’s compensation or general liability), receipts, sales, square footage, area, or man-hours (for general liability), per unit (as in automobile), or per \$1,000 of value (as in property insurance).

form

Forms are the documents (template contracts) defining standard types of coverage and endorsements. A policy typically contains a series of forms plus a dec sheet (which provides a summary of the benefits provided by the policy).

generated product

Visualized products become generated products after you generate product code in the finalization process. You can view generated products on the **Externally Managed** tab of the **Product Management** screen.

mind map

A visual representation of hierarchical information. The map can help you capture ideas quickly and efficiently, clarify thought processes, manage complex information, and promote team collaboration.

modifier

Modifiers are factors that affect rating and typically result in an increase or decrease in the premium for a policy. There are multiple types of modifiers, many of which are specific to a jurisdiction. Modifiers can be added to a product or a product line. Modifiers added to a product affect rating for all lines in that product. Modifiers added to a product line affect only that line.

offering

Offerings define product variations. You can use offerings to provide products that vary depending on target customers or other marketing criteria.

package

A product that combines multiple lines of business into a single policy for easy administration and as a convenience to the insured.

policy line

The PolicyCenter term for a product line. In PolicyCenter, the policy line defines coverages, forms, modifiers, and other information related to the policy.

product

A *product* is an insurance product, such as personal auto or commercial package, that an insurance company offers for sale to its customers. A product sale results in a policy. The corresponding PolicyCenter term is *product*.

product definition

In the P&C industry, this term usually refers to rates, rules, and forms. In PolicyCenter, this term also includes product and risk modeling, rating, underwriting rules, and forms inference, among others.

product line

A *product line*, sometimes referred to as *line*, is contained in a *product*. Multiline products contain one or more product lines. The corresponding PolicyCenter term is *policy line*. In PolicyCenter documentation, *policy line*, *line of business*, and *product* are sometimes used interchangeably, especially in products containing a single policy line.

product model properties

You can use product model properties to specify how to handle product attributes. The product attribute can be a field, clause, clause term, or drop-down list entry. You can use properties to specify the following types of information for a product attribute:

- Default values
- Minimum and maximum values
- Usage properties to specify how a product attribute is handled or obtained
- Tags to mark a product attribute for special processing or handling

question set

A question set is a collection of questions presented within PolicyCenter that gathers information about an applicant. You use the answers to these questions to evaluate the risk associated with a policy applicant. You can use question sets to:

- Determine whether an applicant qualifies for a given product
- Gather information relevant to the policy premium

rate-scalable

Most insurable objects are rate-scalable. For rate-scalable objects, the rating looks up the rate based on a variety of factors such as the type of vehicle, the coverage terms chosen, the date and state. Often this calculation uses data from a table. Rating modifies that rate based on a variety of other factors (such as vehicle cost or driver age) to make a final adjusted rate. That final adjusted rate is generally the cost for the entire rated term. The final cost is that amount prorated over the portion of the policy for which the cost is appropriate.

reference date

The reference date is compared against the start and end effective dates to determine the availability of a pattern.

risk object

A risk object captures information about the risks covered by a policy. The risk object can be a tangible object such as a car or a building. The risk object can be a location, which is special type of risk object. Or the risk object can be an intangible object such as a jurisdiction or the policy line. A risk object may or may not have attached coverages. In PolicyCenter, a risk object is referred to as a *coverable*.

schedule

Schedules are lists that contain detailed information about an insured's risk objects. Schedules can have coverage terms, and these types of schedules can affect rating.

surcharge

As a premium, it is a extra charge, typically when an insurer feels that standard rates are not sufficient for a single policyholder. The policyholder may be offered insurance only if they will accept a surcharge added to the premium. Some taxes are also referred to as surcharges. As a tax, it is usually a regulatory charge enacted into law for a particular purpose in a particular jurisdiction. In essence, it is a tax that is being collected for a particular time and for a certain defined purpose.

template

The XML file that Advanced Product Designer creates when you export a visualized product. You can also import this template back into PolicyCenter.

visualized product

You can view visualized products on the **APD Managed** tab of the **Product Management** screen. Visualized products are created using Advanced Product Designer screens in PolicyCenter or by importing an XMind mind map into PolicyCenter.

