Guidewire PolicyCenter™

New and Changed Guide

Release 10.0.0

GUIDEWIRE

Adapt and succeed™

Product Name: Guidewire PolicyCenter
Product Release: 10.0.0
Document Name: New and Changed Guide
Document Revision: 25-September-2018

Contents

Part 2
New and changed in 9.0 maintenance releases

Part 3
New and changed in 9.0.0

Part 4
New and changed in 8.0 maintenance releases

Part 6
Release Notes Archive

GUIDEWIRE

About PolicyCenter documentation

The following table lists the documents in PolicyCenter documentation:

Document	Purpose
InsuranceSuite Guide	If you are new to Guidewire InsuranceSuite applications, read the *InsuranceSuite Guide* for information on the architecture of Guidewire InsuranceSuite and application integrations. The intended readers are everyone who works with Guidewire applications.
Application Guide	If you are new to PolicyCenter or want to understand a feature, read the *Application Guide*. 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.
Database Upgrade Guide	Describes the overall PolicyCenter upgrade process, and describes how to upgrade your PolicyCenter database from a previous major version. The intended readers are system administrators and implementation engineers who must merge base application changes into existing PolicyCenter application extensions and integrations.
Configuration Upgrade Guide	Describes the overall PolicyCenter upgrade process, and describes how to upgrade your PolicyCenter configuration from a previous major version. The intended readers are system administrators and implementation engineers who must merge base application changes into existing PolicyCenter application extensions and integrations. The *Configuration Upgrade Guide* is published with the Upgrade Tools and is available from the Guidewire Community.
New and Changed Guide	Describes new features and changes from prior PolicyCenter versions. Intended readers are business users and system administrators who want an overview of new features and changes to features. Consult the "Release Notes Archive" part of this document for changes in prior maintenance releases.
Installation Guide	Describes how to install PolicyCenter. The intended readers are everyone who installs the application for development or for production.
System Administration Guide	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.
Configuration Guide	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.
PCF Reference Guide	Describes PolicyCenter PCF widgets and attributes. The intended readers are configuration engineers.
Data Dictionary	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.
Security Dictionary	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.
Globalization Guide	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.
Rules Guide	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.
Contact Management Guide	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.

Document	Purpose
Best Practices Guide	A reference of recommended design patterns for data model extensions, user interface, business rules, and Gosu programming. The intended readers are configuration engineers.
Integration Guide	Describes the integration architecture, concepts, and procedures for integrating PolicyCenter with external systems and extending application behavior with custom programming code. The intended readers are system architects and the integration programmers who write web services code or plugin code in Gosu or Java.
Java API Reference	Javadoc-style reference of PolicyCenter Java plugin interfaces, entity fields, and other utility classes. The intended readers are system architects and integration programmers.
Gosu Reference Guide	Describes the Gosu programming language. The intended readers are anyone who uses the Gosu language, including for rules and PCF configuration.
Gosu API Reference	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.
Glossary	Defines industry terminology and technical terms in Guidewire documentation. The intended readers are everyone who works with Guidewire applications.
Product Model Guide	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 *Product Designer Guide*.
Product Designer Guide	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.

Conventions in this document

Text style	Meaning	Examples
italic	Indicates a term that is being defined, added emphasis, and book titles. In monospace text, italics indicate a variable to be replaced.	A *destination* sends messages to an external system. Navigate to the PolicyCenter installation directory by running the following command: `cd installDir`

bold	Highlights important sections of code in examples.	`for (i=0, i<someArray.length(), i++) {`
		`   newArray[i] = someArray[i].getName()`
		`}`

narrow bold	The name of a user interface element, such as a button name, a menu item name, or a tab name.	Click **Submit**.
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.
monospace italic	Variable placeholder text within code examples, command examples, file paths, and URLs.	Run the `startServer server_name` command. Navigate to `http://server_name/index.html`.

Support

For assistance, visit the Guidewire Community.

Guidewire customers

`https://community.guidewire.com`

Guidewire partners

`https://partner.guidewire.com`

New and changed in 10.0.0

This part contains information about PolicyCenter 10.0.0.

chapter 1

New and changed in the application in 10.0

New in the application in 10.0.0

Homeowners line of business

Homeowners insurance provides protection against the financial consequences of losses related to owning and renting a home. A homeowners policy is a combination of property and liability coverages.

The homeowners line is new in this release.

See also

• *Application Guide*

Smart Communications for PolicyCenter

Smart Communications for PolicyCenter integrates with SmartCOMM and enables you to create and edit documents in the PolicyCenter user interface and in bulk using SmartCOMM templates. In the base configuration, the integration produces documents in PDF format. Through configuration, you can add support for other channels such as HTML and email. This integration requires that you have Smart Communications SmartCOMM product. The integration does not include any SmartCOMM templates. You must create and configure templates to work with the integration. You must purchase and license SmartCOMM separately. The integration uses a subset of SmartCOMM features.

See also

• *Application Guide*

New in quoting and rating performance

There are several improvements to quoting and rating performance in this release.

See also

- *Application Guide*

Asynchronous quoting

In certain lines of business, generating quotes for policies can take a noticeable amount of time. For example, this can occur on commercial policies with large numbers of coverables. Asynchronous quoting enables the quote to run in the background, so that the user can do work on other screens until the quote completes.

Two-step quoting

With two-step quoting, the first step validates the policy data and rates the policy, generating all cost and premium information, and bringing the policy period to Rated status. The second step completes post-rating tasks, such as generating forms, checking reinsurance, and raising underwriting issues. If the second step is successful, the policy period is then in Quoted status. With two-step quoting, you can delay or omit the second step entirely. For example, when working with multiple versions of a policy transaction, an underwriter does not need to generate forms, reinsurance, or underwriting issues for the policy until a specific version is chosen. Or in high volume quoting, only the actual price of the policy is relevant so it is unnecessary to do post-rating tasks. In the base configuration, two-step quoting is enabled for the commercial property line of business, but can easily be enabled for the remaining lines. Side-by-side quoting in personal auto uses two-step quoting. High volume quote requests use two-step quoting for all lines of business.

Parallel rating

Parallel rating can improve the performance of generating quotes on policies with large numbers of coverables by rating coverables in parallel using multiple threads. The implementation requires Guidewire Rating Management. The base configuration implementations of parallel rating for the commercial property line of business. Two types of parallel rating are provided:

- Parallel rating using entities
- Parallel rating using DTOs

Parallel product model synchronization

Parallel product model synchronization runs in parallel on multiple threads and improves the performance of synchronizing the product model. In the base configuration, product model synchronization occurs primarily during quoting and is available for all commercial products.

New in business rules in 10.0

New business rule variables

New in 10.0, Guidewire has added the ability to create rule variables that store a specific expression or value for later re-use in the rule condition and action builders. A variable expression can take any of the following forms:

- Formula
- Count
- Sum
- Lookup

See the *PolicyCenter Application Guide* for more information.

New Lookup expressions in business rules

New in PolicyCenter 10.0 is the ability to create Lookup expressions in business rules. A lookup expression is a named function (query) that takes one or more inputs and returns a single value from a database lookup table. A common use is to create a rule variable that uses a Lookup expression and then use that variable in the rule condition and action builders.

See the *PolicyCenter Application Guide* for more information.

Summary dashboards

The **Account Summary**, **Policy Summary**, **Account Holder Summary**, and **Desktop** screens now have summary dashboards that provide organized overviews of key data. These dashboards contain information from ClaimCenter and BillingCenter if you are integrated with these products. The dashboards also provide navigation to take actions or get more information.

See also

- *Application Guide*

New formatting support for input fields

As the user types in data for some types of input fields, PolicyCenter formats the data appropriately for the field. This feature supports currency, date, and time fields and fields with input masks.

In addition, there is macro support for currency fields that enables the user to enter a number and a letter for thousands, millions, and so on and have them converted to numbers. For example, in the base configuration, entering `1.45k` in a currency field automatically converts to `1,450.00`.

When you change data in an input field, the field is highlighted to indicate that it has been changed.

See also

- "As-you-type formatting support for input fields" in the *Application Guide*
- "Using the currency macro in currency fields" in the *Application Guide*
- "Highlight changed values" in the *Application Guide*

New interface settings

There are new settings that control the behavior and appearance of the application interface.

See also

- "Changing interface settings" in the *Application Guide*

Changed in the application in 10.0.0

Changes to high volume quote requests

This release includes the following changes for handling high volume quote requests:
- User interface for searching the quote store
- Additional methods in the API

Expanded BillingCenter integration

PolicyCenter integration with BillingCenter now provides improved control over the invoice stream and payment instrument.

From PolicyCenter, you can:
- With policy-level billing, you can override defaults for the invoice stream of each policy. These overrides are propagated to BillingCenter.
- Specify how to invoice excess premium in a policy change.
- Capture the payment instrument.

Multicurrency integrations now provide a more consistent experience in creating organizations and producer codes.

Multicurrency changes

BillingCenter now enables agency bill and commission plans to be defined in multiple currencies. PolicyCenter makes use of these improvements and can associate multicurrency plans with producers and organizations. In previous releases, you had to define a plan for each currency.

If a plan offers more than one currency, you can now select multiple currencies on these screens:

- **Organization→Agency Bill Plans**
- **Producer Code→Commission Plans**

The plugins that communicate with BillingCenter have been updated to handle these changes.

Payment screen changes

The **Payment** screen of a policy transaction has changed:

- New read-only **Billing Level** on the **Billing** summary indicates whether to bill the policy separately or on an account bill. The `IBillingSystemPlugin` plugin implementation retrieves the billing level from the billing system. If this is a new policy, the plugin sets a default value. When integrated with BillingCenter, for existing policies the billing level comes from BillingCenter.
- New **Override Change in Cost Billing** appears in policy change, issuance, and reinstatement policy transactions. The **Special Handling** field enables you to override how to bill the changes in cost. The overrides are propagated to BillingCenter. The default billing splits the cost change proportionally among the payments. **Bill Immediately** creates a new payment for the whole cost change amount. **Bill on Next Invoice** adds the payment to the next scheduled invoice.
- New **Hold for Final Audit** billing override option appears for policies that have the schedule final audit set to **Yes**.
- **Invoicing Overrides** replaces **Custom Billing Options**. You can override the default invoicing by specifying whether to send invoices on a specified date or relative to the due date. You can also override the payment instrument.
- New **Fix Invoices by** replaces **Due Date** in the **Invoicing** summary. The value comes from the `BillDateOrDueDateBilling` typelist on the payment plan (`PaymentPlanSummary`).
- In the **Payment Schedule** listing, you can only view the **Schedule** for the selected **Payment Plan**.
- If you specify an **Alternate Billing Account**, the policy gets the **Billing Level** from the alternate account.

Restoring Custom Billing Options

In this release, **Custom Billing Options** on the **Payment** screen has been deprecated and replaced by **Invoicing Overrides**. If you used **Custom Billing Options** in a previous release and wish to restore **Custom Billing Options** and the related functionality, set the `EnableCustomBilling` parameter to `true` in `config.xml`.

Inference tab for form patterns changes

In this release, form inference now includes inferring forms by:

- Policy type or coverage part
- Additional insured and additional interest
- Based on whether a selected coverage, condition, or exclusion is not used on the policy

See also

- *Application Guide*

Generic schedule changes

This release includes generic schedules that you can work with in Product Designer. In Product Designer, you can add new *scheduled clauses* to a line of business. Scheduled clauses are clauses with a generic schedule attached. You can adapt the generic structure to represent many different schedules. In the base configuration, the Homeowners line of business provides examples of scheduled clauses that use generic schedules.

See also

- *Product Model Guide*

Changes to PolicyCenter business rules

In PolicyCenter versions prior to 10.0.0, the rule list screen showed an **Enabled** column. Guidewire has removed this column in PolicyCenter 10.0.0. Instead, in 10.0.0, the **Run Status** column icon (and tooltip) indicate whether a rule is enabled or disabled.

Also new in 10.0.0 is the ability to directly enable or disable a business rule from the rule list screen or the rule details screen:

- From the rule list screen, by selecting **More→Enable Selected** or **More→Disable Selected**.
- From the rule detail screen, by placing the rule in edit mode and updating the **Enabled** check box.

Clicking the **Enable** / **Disable** button on a rule detail screen toggles the **Enabled** field on the selected version of the rule. The label of the button changes depending on whether the selected rule version is currently enabled or disabled.

Business rules and rolling upgrade

In 10.0.0, during a rolling upgrade, Guidewire disables the ability to deploy a business rule in a production environment. The intent is to prevent the deployment of a rule that contains data that is incompatible with the current environment, thereby causing the rule execution to fail.

During a rolling upgrade, PolicyCenter:

- Disables the **Deploy All** and **Deploy Selected** options on the rule list screen.
- Disables the **Deploy** button on the rule detail screen.
- Displays an alert message on the rule list and rule details screen that indicates rule deployment is not possible due to an ongoing upgrade.

See also

- *Application Guide*

chapter 2

New and changed in system administration in 10.0.0

New in system administration in 10.0.0

Intentional logging

New in PolicyCenter 10.0.0 is the ability to track specific business events in the application logs.

Business event logging

A *business event* is an event of special interest in the PolicyCenter application, such as a user logging into Guidewire PolicyCenter. New in 10.0 is the ability to use special event markers to capture information about certain significant events in the application and output that information in multiple ways. A special event marker is an additional attribute that prints to the log along with the log message. This additional information makes it relatively easy to find and extract the log record associated with a special event.

In Guidewire InsuranceSuite applications, *intentional logging* is the act of capturing information about specific business events or application metrics and outputting that information to the PolicyCenter application logs.

Profiler integration

Guidewire integrates intentional logging with Guidewire Profiler. Thus, it is possible to generate markers for intentional logging automatically while using Guidewire Profiler.

Traceability IDs

New in logging in 10.0.0 is the use of traceability IDs. A *Traceability ID* is a `String` UUID (Universally Unique Identifier) that Guidewire uses to track a single user transaction or operation across multiple logging statements. The logging statements can span calls across multiple applications in inter-application integration.

Configuring intentional logging

See "Intentional logging configuration" on page 39 for more information.

Skip producer code checks on accounts with no active policies

To skip producer code checks on accounts with no active policies, use the `bypassProducerCodeCheckOnAccountsWithNoActivePolicies` attribute in `security-config.xml`. When true, if all policies on the account are expired, any agent (regardless of agency) can view the account, and can start a new submission on that account. The account is treated as if there are no policies on it.

See also

- *System Administration Guide*

System property VerifyRuleImportDataModelCheckSum

New in PolicyCenter 10.0.0 is system property `VerifyRuleImportDataModelCheckSum`. Use this system property to enable or disable verification of the data model while importing business rules into a non-production system. This is especially useful as a data model change in the importing environment can prevent rule import. By disabling the data model verification, it becomes possible to import the rules.

System property `VerifyRuleImportDataModelCheckSum` can be one of the following Boolean values.

`true`	PolicyCenter enforces verification of the data model during rule import. The default is `true`.
`false`	PolicyCenter does not enforce verification of the data model during rule import.

You can set system property `VerifyRuleImportDataModelCheckSum` in the following ways:
- In PolicyCenter Studio, in the server JVM options.
- From a PolicyCenter command prompt, as you start the application server.

See also

- *System Administration Guide*

Changed in system administration in 10.0.0

Changes to PolicyCenter logging

In PolicyCenter 10.0, Guidewire has changed how to configure and use application logging.

Logging API slf4j

Guidewire now uses the `slf4j` API, in conjunction with Apache log4j-2 libraries and internal Guidewire libraries, to provide logging in the PolicyCenter application.

Logging configuration file log4j2.xml

New in 10.0, Guidewire uses the XML-formatted `log4j2.xml` logging configuration file to control the behavior of the logging activity in PolicyCenter. You access this file from the following location in Guidewire Studio:

configuration→config→logging

Configuration file `log4j2.xml` replaces the previous logging configuration file `logging.properties`. File `log4j2.xml` contains a number logger categories defined in the following public classes:

```
PCLoggerCategory
BizRulesLoggerCategory
```

Changes to Guidewire Profiler

In PolicyCenter 10.0.0, Guidewire has added the ability to profile Gosu servlets and REST API operations. To access Guidewire Profiler, log into Guidewire PolicyCenter using an administrative account and navigate to the following location:

Server Tools→Guidewire Profiler→Configuration

Changes to SQL Server DMV Snapshot screen

In PolicyCenter 10.0.0, Guidewire has changed the name and functionality of the (Server Tools) **SQL Server DMV Snapshot** screen. The new name of the screen is **SQL Server Performance Report**. This screen is only available if the database is SQL Server.

The updated SQL Server Performance Report screen

The renamed **SQL Server Performance Report** screen can now generate reports that include the following types of data, if specified:

Statistics data	Description
SQL Server Dynamic Management Views	SQL Server provides Dynamic Management Views as a tool to monitor, diagnose, and tune database performance. Each performance report contains DMV information. PolicyCenter automatically generates this type of data and provides the data in each report.
SQL Server Query Store	SQL Server Query Store provides a set of tables separate from the Dynamic Management Views. Query Store persists data to disk, whereas DMV data is in-memory only and lost after a server restart. Thus, it is possible to obtain Query Store database statistics across server restarts. You must enable SQL Server Query Store before you can generate statistics that use Query Store.
Database statistics	This is the same information as available from the Server Tools **Database Statistics** report.

The updated screen contains several options that you can select:

Option	Action
Include Query Store Analysis	Select to include Query Store statistics in the SQL Server performance report. If you select this option, then you must also select a time interval for the Query Store data as well.
Include database statistics	Select to include basic database statistics in the SQL Server performance report.

See the *System Administration Guide* for more information.

Changes to system_tools

To support the updated SQL Server performance reporting, Guidewire has modified the `system_tools` command slightly. The following command options relate to the SQL Server performance reports.

Command option	New or Changed	Description
`-sqlListIntervals`	New	Determine the IDs of the SQL Server performance report time intervals
`-mssqlPerfRpt`	Changed	Generate a performance report using the interval IDs
`-listPerfReports`	No change	Determine the ID of each generated report
`-getPerfReport`	No change	Download an existing report to a local directory using its ID

See the *System Administration Guide* for more information.

New and changed in configuration in 10.0.0

Changes to PolicyCenter configuration

Remove unused locations

In the base configuration, PolicyCenter removes unused locations from policies in a line of business. You can choose to implement your own code that removes unused locations for a line of business.

See also

- *Configuration Guide*

Configuration changes for quoting and rating performance

Several configuration changes were made for quoting and rating performance.

Incorporating your customizations to QuoteProcess class

Adding two-step quoting to the lines of business in the base configuration required many changes to the `QuoteProcess` class. In particular, many changes were made to the `requestQuote` method. This method is now called for requesting a rate in addition to requesting a quote. If you customized the `QuoteProcess` class, merge your changes with the new class during the upgrade merge process.

PolicyPeriod entity changes

For two-step quoting, added a `QuoteMaturityLevel` property to the `PolicyPeriod` entity.

In previous releases, `PolicyPeriod.ValidQuote` indicated that policy period had a valid quote. With this release, more settings are needed for the quote level. Therefore, the `ValidQuote` property has been replaced by a new `QuoteMaturityLevel` typelist with `unrated`, `rated`, and `quoted` typecodes. The `PolicyPeriod.ValidQuote` property is now virtual. `ValidQuote` is `true` when `QuoteMaturityLevel` is rated or quoted.

New typecodes added to PolicyPeriodStatus typelist

New typecodes have been added to the `PolicyPeriodStatus typelist.`

For two-step quoting, a new `Rated` typecode indicates that the policy has been rated. This status is in the `Expirable` and `Open` typelist filters.

For asynchronous quoting, a new `RateRequested` typecode indicates that this policy has been queued to rate asynchronously.

New checking set and blocking point for underwriting issues

Two-step quoting adds a new `BlocksRateRelease` blocking point to the `UWIssueBlockingPoint` typelist, and an associated `PreRateRelease` checking set to the `UWIssueCheckingSet` typelist.

Changes for parallel rating

Adding parallel rating to this release required changes to the following files, among others:

- *XX*`RatingEngine`
- *XX*`PolicyLineMethods`
- `AbstractParallelRatingEngine`

Guidewire Solr Extension updated to Solr 6.6

Guidewire updated the Guidewire Solr Extension version of the Apache Solr free-text search engine to 6.6.

Change to loading Rating Management components

In this release, `preload.txt` no longer loads Rating Management components on system startup. Now `RatebookPrimer.gs` in `gw.rating.startup` loads rate books and the included Rating Management components on system startup. Loading Rating Management components on system startup is enabled by default. You can modify `RatebookPrimer.gs` to turn off or customize this.

Changes to rules

New configuration options for business rules in 10.0.0

The following configurations are new for business rules in PolicyCenter 10.0

New business rule permissions

Guidewire has added the following business rule-related permissions for underwriting rules to file `security-config.xml`:

- View underwriting rule
- Edit underwriting rule
- Deploy underwriting rule
- Import underwriting rule
- Approve underwriting rule

Guidewire now manages the business rule permissions through file `security-config.xml`.

See the *System Administration Guide* for more information.

Business rule Lookup expression configuration

New in PolicyCenter 10.0 is the ability to create Lookup expressions in business rules. Guidewire disables the use of Lookup expression in PolicyCenter by default. However, the PolicyCenter `BizRulesPlugin` implementation class overrides this default, permitting the use of Lookup expressions in PolicyCenter business rules.

See the *Configuration Guide* for details.

Changes to Guidewire Studio

Changes to inspections in Guidewire Studio

Guidewire Studio includes multiple changes to the available code inspections, including new inspections and changes to which inspections are enabled by default. These changes could potentially trigger new build errors or warnings in your existing code. Review the enabled inspections listed in Guidewire Studio to familiarize yourself with them, and make any desired changes to the inspections that are enabled or disabled. In many cases, the base Guidewire applications have been updated to conform to the conventions indicated by the inspections.

See also

- *Configuration Guide*

Changes to configuration parameters

Product model CodeIdentifier uniqueness

The new `StrictProductModelIdUniquenessChecking` parameter specifies whether product model code identifier, `CodeIdentifier`, must be unique for all product model objects. Because the default setting is true, PolicyCenter throws an error if two product model patterns of any type share the same code identifier. For recommendations on how to handle this issue, see *Configuration Guide*.

Added configuration parameters

The following configuration parameters have been added in PolicyCenter 10.0.0. For more information, see the *Configuration Guide*.

- AllowStartupInDevelopmentModeWithDomainGraphErrors
- AsynchronousQuotingEnabled
- CanceledActivityRetentionWindow
- ConfigurationServiceEnabled
- CoverableCacheEnabled
- DocumentProductionBindingType
- DocumentProductionIntegrationEnabled
- EnableCPDTOParallelRating
- EnableCustomBilling
- HVQEnabled
- HVQServer
- MaximumXmlExportSize
- MaxHVQSearchResults
- MaxProductModelSyncThreadPoolSize
- MaxRatingThreadPoolSize
- MinNumberOfCoverablesToParallelizeProductModelSync
- ParallelizedProductModelSyncEnabled
- ParallelizedRatingEnabled
- ParallelProductModelSyncTimeoutPerCoverable
- ParallelRatingTimeoutPerCoverable
- RewriteMaxQuotes
- RewriteMaxSideBySideQuotes
- ScopedVariableSupportedForAllCapturedRequests
- SideBySideTwoStepQuotingEnabled
- StrictProductModelIdUniquenessChecking
- TwoStepQuotingEnabled
- UpLinkShortcut

Removed configuration parameters

The following configuration parameters have been removed in PolicyCenter 10.0.0:

- CheckSpellingOnChange
- CheckSpellingOnDemand
- DisableDomainGraphSupport
- MigrateToLargeIDsAndDatetime2
- RateTableManagementMemoryRowThreshold

Changes related to PCF files

Themes

The method for changing the appearance and styles of the application has changed.

See also

- "Configuring the visual style of the application" in the *Configuration Guide*
- "Change the visual theme" in the *Application Guide*

Icon files and types

The types of icon files and method for specifying them in PCF files has changed.

See also

- "Using icons" in the *Configuration Guide*

More control over list view layout

You have more control over the layout and behavior of a list view. Use the new properties `stretch`, `grow`, `wrap`, `width`, `minWidth`, and `maxWidth` to control how a list view and its columns appear on the page.

See also

- "Managing the layout of a list view" in the *Configuration Guide*
- "Best practices for the layout of a list view" in the *Configuration Guide*

Dashboards

You can create dashboards, which provide summary information, highlight interesting data, and provide links to more detailed views.

See also

- "Dashboards" in the *Configuration Guide*

Other PCF changes

The following are changes to the configuration of PCF files:

- The supported chart types have changed. `Radar`, `Scatter`, and `XYArea` charts are no longer supported. Added the new chart type `Donut`.
- You can now set the position of the legend in a chart using the `LegendPosition` element.

Other configuration changes

Change to exported user data

In 10.0.0, PolicyCenter handles the export of user data from the **Administration→Utilities→Export Data** screen differently between development and production systems.

Server mode	User export data
Development mode	The export file contains usernames and the associated passwords in hash code format.
Production mode	The export process strips both username and password information from the export file.

Because of this change, it is not possible to import a previously exported data file from a production environment back into PolicyCenter as it does not contain the required username/password data.

Configure environments by specifying multiple values in env attributes

You can now specify multiple values for an `env` attribute by using a comma-separated list.

The env attribute allows you to switch between configurations for any configuration element that supports the attribute. The configuration elements for which the env attribute allows you to switch between configurations impact the following XML and other configuration files:

- `batch-process-config.xml`
- `config.xml`
- `database-config.xml`
- `messaging-config.xml`
- `scheduler-config.xml`
- `servlets.xml`
- `suite-config.xml`
- `work-queue.xml`
- `*.gwp`
- `*.wsc`

A prominent use case of the env attribute is switching URLs based on which environment is active. For example, you can assign production, QA, and development environments to different URLs. A match between the server start environment and an env attribute value determines the URL at which PolicyCenter starts a server. If no match exists, a default env attribute applies to determine the URL.

The new feature allows you to specify multiple values for an env attribute by using a comma-separated list. Designating multiple values for an env attribute eliminates the need for configuration elements that, except for the attribute, are otherwise duplicates. Moreover, the multi- value *env* attribute reduces the number of configuration elements you must specify by one less the number of env attribute values.

Consider an example of using the multi-value feature of the env attribute. In the Plugins Registry editor, you can set the plugin deployment environment by setting the **Environment** property. The **Environment** property lists the deployment environment or environments in which the plugin can be active. You can have several possible deployment environments, such as two test environments and a development environment.

Suppose that you want your plugin to be active in only two of these environments, the two test environments. To effect this change, in the **Environment** property, specify the test environments by using the following comma-separated list: `test1, test2`. Responsive to the change, the plugin you are configuring will be active only when the server environment variable has a value identifying one of the test environments—`test1 or test2`.

See also

- *System Administration Guide*
- *Integration Guide*

New RetryIfInitialConditionsFail parameter in batch-process-config

Guidewire has added a new optional `<param>` element to file `batch-process-config.xml` to use in defining a value for the `RetryIfInitialConditionsFail` parameter. This parameter governs the behavior of a batch process depending on the return value of batch process class method `checkInitialConditions`. The name of the parameter is case-insensitive.

This parameter takes the following values.

true	If the batch process `checkInitialConditions` method returns `false`, the batch process does not terminate immediately. Instead, it continues to execute until its component lease expires. This is the batch process default behavior if the attribute is missing from `batch-process-config.xml`.
false	If the `checkInitialConditions` method returns `false`, the batch process terminates immediately.

It is also possible to set this parameter individually on a PolicyCenter batch process by including the optional `<param>` element on the `<batch-process>` element definition. If you set the `RetryIfInitialConditionsFail` parameter at the individual batch process level, it overrides any default value for this parameter set at the global level using the root-level `<param>` element.

See the *System Administration Guide* for more information.

Externalized server configuration

In Guidewire PolicyCenter 10.0.0, Guidewire provides the means to modify certain configuration files without the necessity to rebuild and redeploy a server WAR or EAR file. This change provides a mechanism that you can use to modify a well-defined portion of the PolicyCenter application configuration from outside the deployment WAR/EAR file.

The change involves inserting configuration placeholders into the selected configuration files. At run time, the server reads the substitution value from an external configuration file rather from the configuration file packaged with the WAR/EAR file. For example, it is possible to insert a placeholder into a PolicyCenter configuration file and then replace that value externally to modify URLs that access integration points.

Configuration files that support value substitution

The following configuration files support substitution:
- `config.xml`
- `database-config.xml`
- `suite-config.xml`
- `scheduler-config.xml`
- `messaging-config.xml`
- `solrserver-config.xml`
- `*.gwp` (plugin files)
- `inbound-integration-config.xml`

Plugin ExternalConfigurationProviderPlugin

The internal Guidewire Substitutor process uses plugin `ExternalConfigurationProviderPlugin` to determine the substitution for the configuration placeholders. In the base configuration implementation, this plugin implements public class `FileBasedConfigurationProviderWrapper`, which contains a single public method:

    lookupValue(*namespace, key*)

The *namespace* value represents the configuration file that contains the substitution, `config` for file `config.xml`, for example. The key value represents the substitution value to replace in the configuration file.

Logging substitution information

Guidewire provides information in the application log regarding all configuration substitutions and the initial loading of the `ExternalConfigurationProviderPlugin` plugin. The logging information is verbose enough to be able to track all steps and problems of any substitution operation.

Guidewire defines substitution-related log messages in file `log4j2.xml` under the `Server.PropertySubstitutor` logger.

New gwb build command

New in PolicyCenter 10.0.0 is a `gwb` build command that you can use to verify your externalized server configuration. Use the following syntax at a command prompt to execute the command:

    gwb verifyExtConfig

As with all `gwb` build commands, you must supply the password of an administrative user for the command to execute.

Intentional logging configuration

There are two types of configuration possibilities for PolicyCenter:
- Enable or disable intentional logging globally for the entire PolicyCenter application cluster
- Control the individual application elements that PolicyCenter logs during intentional logging

Guidewire abstracts the individual application elements that it logs in intentional logging as `ILElement` element.

Global configuration

Use `IntentionalLoggingEnabled`, a static Boolean property on `ILElement`, to enable or disable intentional logging globally, for example:

```
ILElement.IntentionalLoggingEnabled = true
ILElement.IntentionalLoggingEnabled = false
```

Individual element configuration

Use the Boolean `Enabled` property on specific `ILElement` elements to enable or disable individual elements in intentional logging, for example:

```
(PCProfilerTag.QUOTE_SYNC as ILElement).Enabled  = true
```

New logging classes

Guidewire provides the following new classes in PolicyCenter 10.0.0 to manage business event markers.

Class	Description
PLLoggingMarker	Read-only Guidewire class that defines the valid logging categories for business event logging.
LogMessageParams	Public class to use in constructing messages for the application log.

User credential masking

In PolicyCenter 10.0.0 it is now possible to mask the input of the user name and passoword in a command prompt tool. To do so, insert a dash in the command in place of the actual user name or password, for example:

```
system_tools -ping -user - -password -
```

Before PolicyCenter executes the tool script, it prompts you to enter the missing information. If you chose to mask the user name and password information:

- PolicyCenter does not echo the entered user name or password back to the screen.
- The operating system, either Windows or Linux, does not store the plain text information in the command prompt buffer and command history.

New and changed in Product Designer in 10.0

Changed in Product Designer in 10.0

Product Designer language

Product Designer 10.0 includes all files needed for languages supported by PolicyCenter. Language-specific content for the user interface and online help is provided for all languages in the base configuration.

Previously, a `README.html` file in `productdesigner/language-config` provided instructions for installing other languages. This information has been revised and moved to the Product Designer Guide.

See also

• *Product Designer Guide*

New and changed in globalization in 10.0.0

New in globalization in PolicyCenter 10.0

New process for enabling languages

All language translations that Guidewire supports are provided directly in Guidewire Studio localization files at the time PolicyCenter becomes generally available. To enable a language, such as a display language, you use Studio to indicate that a language is to be used in the product, and then restart the application server.

See also

- *Globalization Guide*

Changed in globalization in PolicyCenter 10.0

Language packs no longer used to install languages

PolicyCenter 10.0 ships with all files needed for the languages supported by Guidewire. You no longer need to install a language pack to enable languages in addition in U.S. English.

Note: To install locale changes, you continue to install locale packs.

See also

- *Globalization Guide*

Changes in effects of setting the DefaultApplicationLanguage

The primary display language defined in `DefaultApplicationLanguage` does not cause that language to become a fallback for any other language. Setting the primary display language does set the default preferred language for new users. A user always has a preferred language, and the default preferred language is the primary display language defined in `DefaultApplicationLanguage`.

A user can set their preferred language to any language that you have enabled for PolicyCenter.

See also

- *Globalization Guide*

PolicyCenter uses language fallback

PolicyCenter uses strict language fallback to resolve missing display and typecode keys for the user's preferred language. In the base configuration, U.S English is defined as the primary display language and is the base language for language fallback.

See also

- *Globalization Guide*

Localizations folder in Studio has new subnodes

In PolicyCenter 9.0, the **Localizations** folder in Guidewire Studio displayed files directly in the folder.

In PolicyCenter 10.0, there are nodes under **Localizations** that group files as follows:

Resource Bundle 'display'

Contains display key files for all the language translations provided in the base configuration.

Resource Bundle 'gosu.display'

Contains Gosu display key files.

Resource bundle 'typelist'

Contains property files for all the typecode language translations provided in the base configuration.

Resource bundle 'productmodel.display'

Contains display key files for all the product model language translations provided in the base configuration.

See also

- *Globalization Guide*

Base fallback display key properties file renamed display.properties

In the base configuration of PolicyCenter 9.0, the `en_US` display properties file was named `display_en_US.properties`. In the base configuration of PolicyCenter 10.0, this file is now named `display.properties` and is the base display properties file for language fallback.

See also

- *Globalization Guide*

Base fallback typelist language properties file renamed typelist.properties

In the base configuration of PolicyCenter 9.0, the `en_US` typelist properties file was named `typelist_en_US.properties`.

In the base configuration of PolicyCenter 10.0, the `en_US` typelist properties file is now named `typelist.properties` and is the base typelist properties file for language fallback.

See also

- *Globalization Guide*

Base fallback Gosu display properties file renamed gosu.display.properties

In the base configuration of PolicyCenter 9.0, the `en_US` Gosu properties file was named `gosu.display_en_US.properties`.

In the base configuration of PolicyCenter 10.0, this file is now named `gosu.display.properties`.

See also

- *Globalization Guide*

Base fallback product model display properties file renamed productmodel.display.properties

In the base configuration of PolicyCenter 9.0, the `en_US` product model properties file was named `productmodel.display_en_US.properties`.

In the base configuration of PolicyCenter 10.0, this file is now named `productmodel.display.properties` and is the base product model display properties file for language fallback.

See also

- *Globalization Guide*

chapter 6

New and changed in Gosu in 10.0.0

New in Gosu in 10.0.0

GUnit Test Framework

The GUnit Test Framework enables the writing of tests to exercise the application's Gosu configuration code. Tests are written in Gosu. You can execute these tests from the command line to include the tests in an automated continuous integration work flow. You can also execute these tests from Guidewire Studio.

See also

- *Gosu Reference Guide*

Support for static imports

Gosu now provides support for importing static features of Gosu and Java classes. The syntax for a static import uses a # symbol between the class name and the static features to import. After importing the static features, you can use them in your code without qualifying them with their path.

See also

- *Gosu Reference Guide*

New modifier reified for generic functions

In previous releases, the type parameter for a generic method was always reified, so type information was always available at run time. This parameter is now reified only if the `reified` keyword is a modifier on the method definition.

Gosu now requires the modifier `reified` on generic functions that have any of the following characteristics, where `T` is the token for the type parameter of a generic type:
- The method uses the `new` keyword to create an instance of `T`.
- The method uses the `new` keyword to create an instance of a generic type and includes the `<T>` qualifier on the type name.
- An expression in the method uses the generic type with the `typeis` or `typeof` operator.
- The method contains code that uses `as T` to cast a value to the generic type.
- The method overrides a reified method in a parent class or implements a reified method in an interface.
- The method calls another reified method and passes an argument of a generic type to that method.

The modifier `reified` is optional on a parameterized method that does not have any of these characteristics. Gosu forbids the modifier `reified` on methods that are not generic.

See also

- *Gosu Reference Guide*

Support for empty vararg arguments in calls to Java methods

In previous releases, if you called a Java method that had a `vararg` parameter, you had to provide an empty list if you did not want to provide a value. Now, you do not have to pass that empty list.

Example

Consider a Java method that has the following signature. The method has a single named parameter that is a `vararg`. The method accepts zero, one, or many `String` arguments.

```
public String makeOneString(String... args)
```

To pass values to this method, use code like the following lines.

```
var noArgsResult  = makeOneString()               // No arguments
var oneArgResult  = makeOneString({"one"})        // One argument value
var twoArgsResult = makeOneString({"one", "two"}) // Two argument values
```

See also

- *Gosu Reference Guide*

Support for class path in manifest in JAR file

To provide support for long class paths, Gosu now recognizes the `Class-Path` header in the manifest of a JAR file. The class path in this header adds to the existing class path that Gosu uses to find classes.

See also

- *Gosu Reference Guide*

Changed in Gosu in 10.0.0

Syntax has changed for numeric literal of short integer type

The Gosu syntax for short integer numeric literals has changed. Instead of using the `s` suffix, Gosu now requires the explicit specification `as short`. Using the `s` suffix for a short integer numeric literal is a syntax error. For example, you must now use the following assignment syntax:

```
var aShort = 4 as short
```

The following line is no longer valid:

```
var aShort = 4s
```

Stricter property access validation

Accessing properties in the following ways now causes compilation errors.
- Accessing a property on an internal class from outside the package
- Accessing a protected property on an abstract class from an anonymous instance

Removed classpath command

Gosu does not support the `classpath` command in Gosu program files. To provide a class path to a Gosu program, set the path before running the program.

See also

- *Gosu Reference Guide*

Getter and setter method syntax for properties is not supported

In previous releases, for Java compatibility, Gosu supported using `getXXX` and `setXXX` methods to get and set the value for a property `XXX` on a Gosu class. This syntax is not supported in release 10.0.0. You must use the dot notation and the property name to get and set property values, as shown in the following code examples.

The class `MyClass` has a property `MyProperty`.

```
class MyClass {
  private var  _myPropertyValue : String

  property get MyProperty() : String   {
    return  _myPropertyValue
  }
  property set MyProperty(str : String)   {
    _myPropertyValue = str
  }
}
```

The following code demonstrates valid and invalid syntax to access the property `MyProperty`.

```
var c = new MyClass()
c.MyProperty = "Yes"     // Sets to "Yes" by calling the set MyProperty property function
c.setMyProperty("Java") // *Compiler error*
print("Gosu Style " + c.MyProperty)      // Calls the get MyProperty property function
print("Java Style " + c.getMyProperty()) // *Compiler error*
```

New and changed in installation and upgrade of PolicyCenter 10.0.0

Changed in upgrade in 10.0.0

Removed support for SQL Server upgrade with 32-bit IDs

Before upgrading your SQL Server database to PolicyCenter 10.0.0, you must migrate your schema to use 64-bit IDs (`BitInt` datatype instead of `Int`).

Database upgrade will fail the if the source version schema is still using 32-bit IDs. If you have not yet migrated your SQL Server database to use 64-bit IDs, you must set the parameter `MigrateToLargeIDsAndDatetime2` to true and complete your migration to 64-bit IDs before upgrading to PolicyCenter 10.0.0.

New and changed in integration in 10.0.0

New in integration in 10.0.0

Guidewire InsuranceSuite Integration Views

Integration Views allows you to define a stable, versioned contract for publishing data to and retrieving data from the PolicyCenter database. In addition, the feature provides a simple, declarative way to map underlying PolicyCenter data into a serialized format that conforms to the contract.

The technical realization of this contract starts with two mandatory components and an optional component. The mandatory components include a mapping and a schema. The optional component is a filter. The mapping is a set of declarations that transform an object of a particular type first into an intermediate object that conforms to a specified schema. In the Integration Views context, the ultimate serialized form of this intermediate object is a JSON object or XML element. The schema defines the structure of the data involved in the custom processes. The schema also enables you to use off-the-shelf tools to generate objects and methods to work with the data once it is serialized. The optional filter specifies the parts of the schema to use as output or for presentation. The end result of an Integration View is a JSON object or XML element.

See also

• *Integration Guide*

Marshal and unmarshal Gosu objects to and from XML

The `marshal` and `unmarshal` methods of the `WsiExportableUtil` class can convert additional types of Gosu objects to and from XML.

The methods could always process Gosu DTOs (data transfer objects). The following additional types can also be processed.

- Simple types, such as `int`, `Integer`, `Double`, `String`, `boolean`, and `Boolean`
- `GWRemotable` types
- GX models
- XSD types
- Lists of marshallable types

Support for testing JMS inbound integration in non-production mode

PolicyCenter now supports running inbound integration in `dev` mode on the Jetty server that the base configuration of Guidewire Studio provides. You can use this environment to test your implementation of inbound integration while you develop your code. Use one of the unmanaged thread pool types, not a J2EE managed thread pool.

You cannot use the Jetty server in production mode. You must use one of the supported application servers: JBoss, WebLogic, or WebSphere.

Additional directory in plugins/Gosu for internal use only

The `PolicyCenter/modules/configuration/plugins/Gosu` directory now has a subdirectory called `gresources`. This new subdirectory is for internal use as a compilation output path. Never use, modify, or rely upon the contents of this directory.

Changed in integration in 10.0.0

Changes to servlet definitions in servlets.xml

The `configuration/config/servlet/servlets.xml` file now supports `env` and `server` attributes and static parameters that PolicyCenter uses to configure a servlet at start up.

The `env` attribute is an optional comma-separated list of names of environments for which the servlet definition applies. The `server` attribute optionally specifies the server for which the servlet definition applies. The value of this attribute is either a server role preceded by the `#` character or a server ID.

The `servlets.xml` file in the base configuration contains an example of the use of `<param>` subelements in the definition of the `InfoServlet` servlet.

See also

- *Integration Guide*

Changes to inbound integration configuration file

The XML definition structure for inbound integrations has changed:

XML configuration item	Previous releases	Changes
Thread pool definition	Generic `<threadpool>` element	Specific element: • `<j2ee-managed-threadpool>` for JMS integration • `<unmanaged-threadpool-cached>` to reuse an available thread that is not in use and to create a new thread if one is not available • `<unmanaged-threadpool-fixed>` to specify a fixed number of threads • `<unmanaged-threadpool-forkjoin>` to use the number of CPUs that the

XML configuration item	Previous releases	Changes
		JVM detects as the number of threads • `<unmanaged-threadpool-single>` to specify a single thread
Thread pool type	`<gwthreadpooltype>` subelement of `<threadpool>` element	No longer available because the new specific thread pool element types do not require this property
Thread pool environment	Single value for environment	Optional. env attribute accepts comma-separated list of values.
Inbound integration environment	Single value for environment	Optional. env attribute accepts comma-separated list of values.
For intentional logging, traceability ID creation point for all integration types	Not available	Required. `traceabilityidcreationpoint` subelement, a value specified by the constants in `gw.logging.TraceabilityIDCreationPoint`.
JNDI properties for JMS integration	Not available	Required. `<jndi-properties>` subelement.
Plugin handler for custom inbound integration	Required	Not supported.

See also

• *Integration Guide*

JMS inbound integration support for JBoss

PolicyCenter now supports running JMS inbound integration on a JBoss application server. Use the `<j2ee-managed-threadpool>` element to define the thread pool for this integration.

Changes to billing system integration

Several changes were made to billing system integration.

New typelists

• `InvoicingMethod` – Whether to use default billing, custom billing, or override the invoice stream
• `BillingLevel` – How to bill policies on the account
• `BillDateOrDueDateBilling` – Corresponds to typelist in BillingCenter

Changes to StandAloneBillingSystemPlugin

Modified and added more payment plans to the `StandAloneBillingSystemPlugin` plugin:

• Twice Per Month Demo – Statement dates are now reliably twice per month. By using **Invoicing Overrides**, you can set the statement dates. Previously, the statement dates were determined by dividing the range in days by the number of payments, and sometimes missed the expected payment dates.
• Quarterly Demo – New. Send statements quarterly.
• Monthly Demo – New. Send statements monthly.
• Every Other Week Demo – New. Send statements every other week.

New and changed in 9.0 maintenance releases

This part includes information about PolicyCenter 9.0 maintenance releases.

See also

- "New and changed in 9.0.0" on page 79

New and changed in PolicyCenter 9.0.5

For resolved and known issues in 9.0.5, see the Release Notes.

New in PolicyCenter 9.0.5

Oracle statistics preferences

New in 9.0.5 is the ability of PolicyCenter to use the Oracle AutoTask infrastructure to manage the task of collecting table statistics. To enable this functionality, do the following:

- Set the `useoraclestatspreferences` attribute to `true` on the `<databasestatistics>` element in file `database-config.xml`.
- Enable Oracle AutoTask for statistics collection in the Oracle database.

Enabling this functionality makes visible an **Oracle Statistics Preferences** tab in the Server Tools **Info→Database Statistics** screen.

Resetting useoraclestatspreferences

Any change that you make to the value of `useoraclestatspreferences` (`true` to `false` or `false` to `true`) takes affect only after an upgrade, either a full upgrade or a rolling (configuration) upgrade. However, if you reset this attribute from `true` to `false`, PolicyCenter throws an exception during the next upgrade and prevents the upgrade from continuing due to locked table statistics in the Oracle database. Review the details of the exception provided in the server log to determine which table statistics need to be unlocked.

See also

- *System Administration Guide*

New database autoupgrade attribute

In 9.0.5, Guidewire has added an `autoupgrade` attribute to the `<database>` element in file `database-config.xml`. This attribute takes one of the following values:

`full`	Takes precedence and initiates a full upgrade assuming all other necessary conditions are met.

manual	Requires that you set either the database upgrade type (in Server Tools **Upgrade and Versions** screen) or the date system property to initiate an upgrade.

See also

- *System Administration Guide*

Tags and tag values on Data Dictionary entity columns

The data dictionary now indicates that tags are set on fields that use them. It also indicates the values set for the tags.

For example, the data dictionary now indicates whether the `PersonalData` obfuscation tag is set on the various fields that use the tag as well as the tag value. One such value for the `PersonalData` tag is `ObfuscateDefault`.

New configuration parameter for pre-loading system tables

Guidewire has added the following configuration parameter in PolicyCenter 9.0.5:
`PreloadSystemTableToCacheForFasterSynchronization`.

Use this configuration parameter to enable or disable the pre-load of system tables at server start. To enable pre-loading, set this parameter to `true`. Pre-loading the system tables at server start allows for faster synchronization and faster server startup times. However, be aware that the use of system table pre-loading can require PolicyCenter to allocate additional heap space. The default is `true`.

New product model parameters

Clustering

Guidewire has added the following new configuration parameters in PolicyCenter 9.0.5 that relate to PolicyCenter multi-node clustering:

- `ClusteringProductModelUpdateMaxRetries`
- `ClusteringProductModelUpdateRetryTimeout`
- `ClusteringProductModelUpdateSleepTimeCeiling`

In a multi-node PolicyCenter cluster, you install product model changes on one server, then start that server to force the upgrade changes. After the first server completes the upgrade process, you can then start the other nodes in the server cluster simultaneously. Due to the number of concurrent server starts, it is possible for there to be contention for the shared update lock while updating the PolicyCenter product model on each node.

These three parameters work in conjunction with each other to manage how PolicyCenter manages lock contention in the product model upgrade process The following table illustrates the connections between these three configuration parameters.

Parameter	Default Value	Effect
`ClusteringProductModelUpdateMaxRetries`	25 retries	The server attempts to acquire the database lock and fails. It then tries another n-1 times to acquire the lock until it is either successful or reaches the maximum number of permitted retries (n), at which point the lock acquisition fails.
`ClusteringProductModelUpdateRetryTimeout`	5 seconds	The first retry attempt to acquire the lock occurs 5000 ms or 5 seconds after the first attempt. The second retry occurs 10 seconds (2 * 5) after the second attempt. The third retry occurs after 15 seconds (3 * 5). This pattern repeats until the 9th retry,

Parameter	Default Value	Effect
		after which the time between successive retries increases to 50 seconds (10 * 5).
`ClusteringProductModelUpdateSleepTimeCeiling`	50 seconds	After the calculated time between retries equals the value of the sleep time ceiling, PolicyCenter waits this amount of time between all succeeding retry attempts until the lock process either succeeds or fails.

See the configuration parameter information in the PolicyCenter *Configuration Guide* for more information

Availability lookup tables

the lookup table manager uses a cache to improve performance of product model availability lookups. added a `AvailabilityContextCacheEntryExpirationTime` configuration parameter that sets the expiration time.

See also

- *Configuration Guide*

Configure reference date for offerings availability

In the base configuration, offerings availability uses the written date as the reference date. You can change the reference date to effective date by modifying the `UsePolicyPeriodReferenceDateForOfferingAvailability` configuration parameter. If reference date is set to effective date, the date range is the start and end dates of the policy period associated with the policy transaction (job).

See also

- *Product Model Guide*

New Java properties to set default TLS version

New in PolicyCenter 9.0.5 is the ability to use Java system properties to set the default TLS version to use on outgoing secured connections. Use these property overrides to provide a comma-separated list of TLS protocol versions. PolicyCenter uses the first item on the list as the preferred protocol. If that protocol is not available, PolicyCenter tries the subsequent protocols on the list until the connection either succeeds or fails completely.

The following table lists the available property overrides.

Web service type	Property	Syntax
WS-I	`gw.webservices.tls.protocols`	`-D.gw.webservices.tls.protocols="a, b"`
RPC	`gw.tls.protocols`	`-Dgw.tls.protocols="a, b"`

In the table, `a` and `b` refer to TLS versions, for example:

```
<java> ... -D.gw.webservices.tls.protocols="TLSv1.2, TLSv1.1"
```

Notice the following for this example:
- The property definition indicates that TLS1.2 is the preferred protocol. However, if TLS1.2 is not available, PolicyCenter attempts to use TLS 1.1 instead.
- The property definition affects only client WS-I web service calls.

- *System Administration Guide*

Loading Rating Management components

You can now configure how and whether PolicyCenter loads Rating Management components on startup.

- *Configuration Guide*

Use new FormInferenceContext methods

Added `FormInferenceContext` methods that cache coverable jurisdictions (as returned by `Coverable#CoverableState` and `PolicyLine#CoveredStates`) during form processing. Updated the following form inference classes to use the new methods:

- `gw.forms.FormData`
- `gw.forms.generic.GenericClauseSelectionForm`
- `gw.forms.generic.GenericCoverableTypeKeyForm`
- `gw.forms.generic.GenericCovTermSelectionForm`

Update your form inference classes that make calls to `Coverable#CoverableState` or `PolicyLine#CoveredStates` to use the new methods.

Changed in PolicyCenter 9.0.5

Change to rate book import

When importing a rate book in XML format, that XML can now include a rate table definition with additional parameters. Previously, adding a parameter resulted in an import error. You can add parameters but you cannot remove parameters.

For example this can occur when you need to update a rate table definition in multiple instances. In one instance of PolicyCenter, you update the rate table definition by adding a new parameter, then export the containing rate book to XML. You then import that XML to second PolicyCenter instance.

Rating worksheets changes

In Rating Management, made a few changes to the **Rating Worksheet**screen. Added a **Download** button to **Rating Worksheet** screen. The **Expand All** button disappears if rating worksheet contains more than 10,000 rows. Use **Download** to view all rows of the rating worksheet in CSV or HTML5 format.

Updated ping utility

Guidewire has updated the `ping` utility, an unauthenticated web page that you can ping to access information about a PolicyCenter server. To access this web page, use the following URL:

```
http://server:port/pc/ping?v=2
```

In the URL, provide the server name (`server`) and port number (`port`) of the server that you want to ping.

The ping utility returns the following types of information, depending on various factors, including whether the server is a production server and whether the server start was successful.

Information returned	Description
Planned shutdown status	Shutdown status (for example, planned, ready) if the server is involved in a planned shutdown.
Server run level code	ASCII decimal code; for example, 45, which represents the '(' character

Information returned	Description
Server run level name	Server run level; for example, MULTIUSER.
Server run level ordinal	QuickStart run level; for example, 5, which corresponds to the MULTIUSER run level.
Server ID	Name of the server host.
Server up time	Time, in seconds, for which the server has been operational.
Server start up exception	Exception that caused the server to fail start up, if applicable. By default, the ping utility does not display this information on a production server.
Thread stack trace	Stack trace of the thread performing the transition from one server run level to another. By default, the ping utility does not display this information on a production server.

Database-related changes in PolicyCenter 9.0.5

Changes to the database statistics report

Guidewire has removed the **Update Statistics Statements for Indexes** table report from the Server Tools **Database Statistics** download report.

Changes to the server tools Consistency Checks screen

Guidewire has slightly modified the (Server Tools) **Info Pages→Consistency Checks** screen. Specifically, Guidewire has modified the labels on the **Change** popup used to set the number of workers to use in generating the consistency report.

New server tools Oracle Statistics Preferences screen

PolicyCenter 9.0.5 has added a new **Oracle Statistics Preferences** screen. You access the screen in the following PolicyCenter location:

Server Tools **Info Pages→Database Statistics** screen, **Oracle Statistics Preferences** tab

Use the functionality on this screen to manage statistics preferences for Oracle databases. See the Server Tools documentation in the Guidewire PolicyCenter System Administration Guide for more information.

Default for ora-parallel-dml attribute changed

Guidewire has changed the default value of attribute `ora_parallel-dml` (on the `<upgrade>` element in file `dabatase-config.xml`) to `enable_all`.

DomainGraphSafeLinks Parameter not supported

Guidewire no longer supports configuration parameter `DomainGraphSafeLinks`. Do not use this configuration parameter.

Deprecated TableImportAPI web service methods

Many of the tasks that the `TableImportAPI` web service performs take a long time to complete. If a task runs synchronously and requires a longer execution time than the network time-out value, the task is terminated before completion.

The `TableImportAPI` web service has deprecated the following methods that provide synchronous access to tasks. To perform the tasks, use the related method that starts a batch process and track the activity of that process.

Deprecated method: deleteExcludedRowsFromStagingTables

Replace calls to the deprecated `deleteExcludedRowsFromStagingTables` method with calls to the `deleteExcludedRowsFromStagingTablesAsBatchProcess` method. This method returns the identifier of the batch

process that is running the task. Use the `MaintenanceToolsAPI` web service method `batchProcessStatusByID` to check for completion of the batch process.

Deprecated method: encryptDataOnStagingTables

Replace calls to the deprecated `encryptDataOnStagingTables` method with calls to the `encryptDataOnStagingTablesAsBatchProcess` method. This method returns the identifier of the batch process that is running the task. Use the `MaintenanceToolsAPI` web service method `batchProcessStatusByID` to check for completion of the batch process.

Deprecated method: integrityCheckStagingTableContents

Replace calls to the deprecated `integrityCheckStagingTableContents` method with calls to the `integrityCheckStagingTableContentsAsBatchProcess` method. This method returns the identifier of the batch process that is running the task. Use the `MaintenanceToolsAPI` web service method `batchProcessStatusByID` to check for completion of the batch process.

Deprecated method: integrityCheckStagingTableContentsAndLoadSourceTables

Replace calls to the deprecated `integrityCheckStagingTableContentsAndLoadSourceTables` method with calls to the `integrityCheckStagingTableContentsAndLoadSourceTablesAsBatchProcess` method. This method returns the identifier of the batch process that is running the task. Use the `MaintenanceToolsAPI` web service method `batchProcessStatusByID` to check for completion of the batch process.

Deprecated method: populateExclusionTable

Replace calls to the deprecated `populateExclusionTable` method with calls to the `populateExclusionTableAsBatchProcess` method. This method returns the identifier of the batch process that is running the task. Use the `MaintenanceToolsAPI` web service method `batchProcessStatusByID` to check for completion of the batch process.

Deprecated method: updateStatisticsOnStagingTables

Replace calls to the deprecated `updateStatisticsOnStagingTables` method with calls to the `updateStatisticsOnStagingTablesAsBatchProcess` method. This method returns the identifier of the batch process that is running the task. Use the `MaintenanceToolsAPI` web service method `batchProcessStatusByID` to check for completion of the batch process.

Message locking for distributed message transactions

When processing a **distributed** message transaction, the transaction's message object is not locked. The locking behavior for distributed transactions can be configured by setting the new `LockDuringDistributedMessageRequestHandling` configuration parameter.

The message-locking behavior for **non-distributed** message transactions remains unchanged. For non-distributed transactions, the message object is always locked.

The locking of the message object affects whether the message's associated primary entity is also locked. The locking of a primary entity instance can be configured by setting the `LockPrimaryEntityDuringMessageHandling` configuration parameter. However, regardless of the parameter's setting, the primary entity instance is locked only if the transaction's message object is also locked. For example, a distributed message transaction that does not lock its message object will not lock the primary entity either, even if locking of the entity is enabled by the parameter.

Changes to Coverable arrays

In previous releases, coverage, condition, and exclusion arrays for coverables in the base configuration were not marked as `owner=true`. This caused situations in which different users adding coverages to the same coverable were not correctly identified by PolicyCenter as concurrent data change exceptions. To remedy this, all `Coverable` entities in the base configuration have been modified to make their `Coverage`, `Condition`, and `Exclusion` arrays `owner=true`.

Guidewire recommends that you examine your configuration and make similar changes to any `Coverable` entities you have defined. Be aware that setting `owner=true` has side effects, such as causing copies of the `Coverable` to be deep copies (including its child `Coverage`, `Condition`, and `Exclusion` entity instances) as opposed to shallow copies. Examine any custom code you have that copies coverables and make sure that this code does not also copy coverages, conditions or exclusions (since these will be copied when `owner=true`).

New and changed in PolicyCenter 9.0.4

See also

- For resolved and known issues in 9.0.4, see the Release Notes.

New in PolicyCenter 9.0.4

This topic includes:
- "Personal data destruction" on page 67
- "Distributed MessageReply plugins" on page 67

Personal data destruction

PolicyCenter supports destruction of some kinds of data. Destruction can mean either purging the data completely from the database or it can mean obfuscating data, making the original contents permanently unreadable.

Guidewire recognizes the need for insurers to be able to destroy personal information both on an on-demand basis or on a time-based basis. Destruction can be mandated by regulation or business practices, within the requirements of regulation, codes of conduct, or other business practices.

See also

- *Application Guide*
- *Configuration Guide*

Distributed MessageReply plugins

By default, `MessageReply` plugin code runs on the server assigned to the message destination. If the plugin class is defined with the `@Distributed` annotation, the plugin code can run on any server in the cluster.

```
@Distributed
public class MyMsgReplyPlugin implements MessageReply { ... }
```

Changed in PolicyCenter 9.0.4

This topic includes:

- "Retrieving a document" on page 68
- "Support for Bing Maps V8 REST interface" on page 69
- "Database element modification" on page 69
- "Histogram statistics database element modification" on page 69
- "Table statistics database element modification" on page 69

Retrieving a document

Several changes are introduced to the operation of retrieving a document.

See also

- *Integration Guide*

New URL_DIRECT response type

The `IDocumentContentSource` plugin methods that retrieve a document support the new response type of `URL_DIRECT`. A response of `URL_DIRECT` is the same as the existing `URL` response type, except the plugin caller will read the returned URL address and redirect the client browser to it. The client browser is then responsible for retrieving the document contents.

Referencing values of the ResponseType property

The supported values for the `DocumentContentsInfo` class `ResponseType` property are `DOCUMENT_CONTENTS`, `URL`, and `URL_DIRECT`. The syntax used when referencing these values has changed.

Original reference syntax: `gw.document.DocumentContentsInfo.VALUE`

New reference syntax: `gw.document.DocumentContentsInfo.ContentResponseType.VALUE`

For example, the `URL` value is referenced as `gw.document.DocumentContentsInfo.ContentResponseType.URL`.

Replace the TargetHiddenFrame property

The `DocumentContentsInfo` class `TargetHiddenFrame` property is replaced with the `ContentDispositionType` property. Supported values for the `ContentDispositionType` property are `DEFAULT`, `INLINE`, and `ATTACHMENT`.

Replaced `TargetHiddenFrame` value	Equivalent `ContentDispositionType` value
true	ATTACHMENT
false	INLINE

The syntax used when referencing a `ContentDispositionType` value is shown below.

Reference syntax: `gw.document.ContentDispositionType.VALUE`

For example, the `INLINE` value is referenced as `gw.document.ContentDispositionType.INLINE`.

With the removal of the `TargetHiddenFrame` property, the `IDocumentContentSource` plugin `target` parameter is deprecated. The replacement parameter is `contentDisposition`. The `target` and `contentDisposition` parameters are defined in the Studio plugin registry for the `IDocumentContentSource` plugin.

DocumentContentsInfo Methods

Changes occurred to some of the `DocumentContentsInfo` class methods.

The `setTargetHiddenFrame` method is deprecated. The replacement method is the new `setContentDispositionType` method.

```
setContentDispositionType(contentDispositionType : ContentDispositionType)
```

The data types changed for two arguments of the `getDocumentContents` method. The new method signature is shown below with the changed argument types in bold.

```
getDocumentContents(strDocUID : String,
                    contentResponseType : gw.document.DocumentContentsInfo.ContentResponseType,
                    contentURL : String,
                    documentURL : String,
                    documentsDir : String,
                    demoDocumentDir : String,
                    contentDispositionType : gw.document.ContentDispositionType,
                    includeContents : boolean)
```

In the original signature, the `ContentResponseType` argument was a `String` and the `ContentDispositionType` argument was of type `boolean`.

New DocumentsUtilBase method

The `DocumentsUtilBase` class includes the new method `getContentDispositionForDocument` to retrieve the `ContentDisposition` value for a document.

Support for Bing Maps V8 REST interface

Features that require geocoding use Bing Maps in the base configuration. As described at `https://blogs.bing.com/maps/June-2016/Bing-Maps-V8-Web-Control-Released`, Microsoft is retiring the SOAP service Bing Maps V7 on June 30, 2017. Its replacement, Bing Maps V8, requires that existing code that uses the Bing Maps SOAP web services be implemented as REST.

There is a new plugin implementation class, `BingMapsPluginRest`, that supports Bing Maps V8. This class is registered by default in `GeocodePlugin.gwp`. There are also new support classes for making requests and handling the returned information.

See also

- *Integration Guide*

Database element modification

Guidewire modified the `<database>` element in `database-config.xml` and added attribute `autoupgrade` to the element. This attribute takes one of the following values:

```
full
manual
```

Histogram statistics database element modification

Guidewire modified the `<histogramstatistics>` element in `database-config.xml` to remove the following attributes:

```
databasedegree
samplingpercentage
```

This change has no meaning or effect at the column (histogram) level.

Table statistics database element modification

Guidewire modified the `action` attribute on the `<tablestatistics>` element in `database-config.xml` to add a `force` option.

```
<tablestatistics name="string" ... action="update|delete|keep|force"/>
```

This option updates statistics for the table while running incremental statistics, regardless of the percentage threshold.

New and changed in PolicyCenter 9.0.3

For resolved and known issues in 9.0.3, see the Release Notes.

New in PolicyCenter 9.0.3

New option field on server tools Oracle AWR screen

In 9.0.3, Guidewire added the following option to the list of options to use in generating a set of performance reports from the Server Tools **Oracle AWR Information** screen:

Include native Oracle report

Selecting this option ensures that PolicyCenter generates an Oracle Standard AWR report automatically as you generate a Guidewire AWR report.

See also

• *System Administration Guide*

Support for labels in GX models

Labels on an XML GX model enable instances of the model to be created that include only the properties required for a particular context. For complete information, see the *Gosu Reference Guide*.

New work-queue.xml attribute

Element `<work-queue>` in `work-queue.xml` contains a new optional attribute named `blockWorkersWhenWriterActive`. If set to `true`, PolicyCenter blocks work queue workers from acquiring work items while the writer is writing new work items to the work queue. The default is `false`.

See the *System Administration Guide* for details.

Force call to Before_Send plugin in distributed messaging process

Support has been added to the distributed messaging process to force the `Before Send` plugin to be called even on retries where the message has already been bound. Forcing the plugin call is desired if an update to the message payload is necessary before retrying to send the message. By default, forcing the plugin call is disabled.

For more information, see the *Integration Guide*.

New gwbexportwsdl command

A new command, `gwb exportWsdl`, exports the WSDL for all WSI web services to a single JAR file.

For more information, see the *Installation Guide*.

New gwbgenschemajar command

A new command, `gwb genSchemaJar`, improves build times.

After an integration project is finished and stable, the XML and XSD schema files that are created during WSDL code generation will rarely change. Instead of having each build recreate these static files, the files can be created a single time and stored in a Schema JAR file. The JAR file is then made available to all developers for general access. Subsequent builds use the contents of the preprocessed JAR to reduce build times.

For more information, see the *Integration Guide*.

New business rules configuration parameter

Guidewire has added configuration parameter `BizRulesCacheStaleTimeMinutes`. Use to manage cache eviction times for the `RuleVersion` cache. See the *Configuration Guide* for more information.

Rolling upgrade supports changes to product model patterns

Guidewire has made the following changes in PolicyCenter 9.0.3 to support the rolling upgrade of very specific updates to product model patterns.

1. Guidewire modified the rolling upgrade process to support making changes to the following product model types in a rolling upgrade:
   - Coverage, Condition and Exclusion, Coverage Term, and Coverage Term Option Patterns
   - Offering and Offering Selection Patterns
   - Question Set and Question Patterns
   - Modifier Patterns
   - Lookup, Grandfathering, and ModifierMinMax
2. Guidewire added batch process `ProductModelPatternActivation`, which you run after completing a rolling upgrade to activate any new product model patterns added during the rolling upgrade.

Guidewire disables (makes inactive) any changes to these product model types until you fully complete the rolling upgrade on all servers. You must then run the `ProductModelPatternActivation` batch process to activate your changes.

See also

- *System Administration Guide*

Changed in PolicyCenter 9.0.3

Changes to upgraded underwriting rules

PolicyCenter 9.0 added a new business rules framework which provided a user interface for defining underwriting rules. You can view this new interface by navigating to the **Administration Business Settings→Business Rules→Underwriting Rules** screen.

When upgrading from PolicyCenter 7.0 and 8.0, legacy underwriting rules have a corresponding underwriting rule in the **Underwriting Rules** interface. The Gosu code that implements the rule continues to work, and accesses the corresponding rule in the **Underwriting Rules** interface. You can choose whether to keep the Gosu implementation, or reimplement the rule in the Underwriting Rules framework.

In PolicyCenter 9.0.0 through 9.0.2, these imported underwriting rules implemented in Gosu are marked as invalid in the **Error** column on the **Underwriting Rules** screen. Although the rules are marked as not enabled, the Gosu implementation continues to run.

As of PolicyCenter 9.0.3, these rules are marked as externally managed in the **State** column on the **Underwriting Rules** screen. On the underwriting **Rule** screen, the **Applies to**, **Rule Context**, **Rule Condition**, and **Underwriting Issue Details** sections do not appear. A rule, `UWRule` object, that is externally managed has its `ExternallyManaged` property set to `true`.

Externally managed rules have their rule condition and action defined outside business rules, often in Gosu code. Externally managed rules include, but are not limited to, upgraded underwriting rules implemented in Gosu.

Specify servers to run messaging operations

Messages are sent to a particular destination, usually an external system. If Distributed Request Processing is enabled for a particular message destination, then its before-send messaging operations can be configured to run on a specific server or group of servers. The server that runs a destination's before-send operations can be different from the server running its other messaging operations.

This capability enables greater flexibility in balancing the execution load associated with messaging operations.

For more information, see the *Integration Guide*.

Document metadata source plugin retrieveDocument method

Implementations of the Document Metadata Source plugin method `retrieveDocument` must initialize the internal properties of the returned `Document` object. The internal properties are initialized by calling the new method `DocumentsUtilBase.initOriginalValues`.

For more information, see the description of the `retrieveDocument` method in the *Integration Guide*.

Work info queue info report

Guidewire now sets the allowable maximum number of writer runs to show in the Work Queue Info report at 150. Previously, the allowable maximum number was 25.

Gosu changes

The following components have been removed from Gosu in PolicyCenter 9.0.3:

- Command-prompt options

  The `-i`, `-interactive`, and - options for the `gosu` command are no longer available. Gosu does not provide an interactive shell or accept a program from the standard input stream.

- Package `gw.lang.cli`

  The `gw.lang.cli` package is no longer available. To get the full list of command-prompt arguments as a list of `String` values, use the `RawArgs` property of the `gw.lang.Gosu` class.

- Class `gw.util.Shell`

- Support for definition of class path in `registry.xml`

  Use either of the following techniques to set the class path:

  ∘ Use the `classpath` command as the first line in a Gosu program file.

  ∘ Use the `-classpath` option to set the class path for a Gosu program that you run from the command prompt with the `gosu` command.

New in PolicyCenter 9.0.2

For resolved and known issues in 9.0.2, see the Release Notes.

New in PolicyCenter 9.0.2

Rule comparison

PolicyCenter now includes a rule comparison feature for underwriting rules, which enables the side-by-side comparison of existing rules and rules to be imported. The **Compare Rules** screen highlights changes and helps resolve rule conflicts.

See also

- *System Administration Guide*

Changed in PolicyCenter 9.0.1

For resolved and known issues in 9.0.1, see the Release Notes.

Changed in PolicyCenter 9.0.1

Distributed message payload transformations before sending

As in previous releases, a destination can optionally implement a plugin to perform payload transformations before the main send action in the `MessageTransport` plugin method called `send`.

In the previous release:

- Payload transformations before message send time was handled by a message request (`MessageRequest`) plugin implementation. This one plugin interface handled transforming the message payload in its `beforeSend` method, which was called before the `MessageTransport` plugin method called `send`.
- The `MessageRequest` plugin did another task, which is to handle post-processing after sending. This task was performed by its `afterSend` method.
- Different messaging destinations could run on different cluster node as long as you had more than one node with the `messaging` server role. All messaging plugins for the same destination ran on the same node.

In this release:

- PolicyCenter supports optionally distributing payload transformation across many servers in the cluster. The new distributed payload transformation work runs as a process across potentially multiple servers in the cluster, potentially long before the actual send time of the message. To use this feature, enable Distribute Request Processing for one or more destinations. To view the Messaging editor, click **configuration→config→Messaging**, then click `messaging-config.xml`. Select the **Distribute Request Processing** checkbox
- You can continue to use the `MessageRequest` plugin as in previous releases. In that case, it handles the before send processing (`beforeSend` method) and the after send processing (`afterSend` method). However, you can optionally use two new interfaces that separate out the two tasks. There are new fields in the destination configuration in the Messaging registry for these two plugin interfaces:
  - The new `MessageBeforeSend` plugin interface handles just the `beforeSend` method. If you use the new Distribute Request Processing feature for that destination, you can set the **Before Send Plugin** field in the destination configuration in the Messaging editor. If this field is set, the destination ignores the **Request Plugin**

field. If you do not use the new Distribute Request Processing feature, instead implement the `MessageRequest` plugin and set the **Request Plugin** field.

- The new `MessageAfterSend` plugin interface handles just the `afterSend` method. Implement the `MessageAfterSend` plugin and set the **After Send Plugin** field. If this field is set, the destination ignores the **Request Plugin** field.

• There is a new `Message` property called `Bound`. It is a `boolean` flag that specifies whether the message request processing is complete. You can optionally set it after message creation if you know that the message does not need processing. After the server performs the before send processing by calling the appropriate `beforeSend` method, the server sets `Message.Bound` to `true`. To correctly handle message resending, your `beforeSend` method must check `Message.Bound` and behave accordingly because the message may have the `Bound` property already set to `true`. In message retry, expect the application to call your `beforeSend` method again.

> **WARNING** You must ensure that the `beforeSend` request processing is idempotent, which means that it could be called multiple times and have the same effect. Your code must ensure that repeated calls cause the same results on the `Message` entity instance and the return value of the method. This warning applies to `beforeSend` methods implemented in either `MessageRequest` or `MessageBeforeSend` plugin interfaces. This warning applies independent of whether you use the new Distribute Request Processing.

See also

• *Integration Guide*

New and changed in 9.0.0

This part includes information about PolicyCenter 9.0.0.

See also

- "New and changed in 9.0 maintenance releases" on page 57

New and changed in the application in 9.0.0

New in the application in 9.0.0

Contingencies on a policy

PolicyCenter manages work associated with creating or changing a policy through policy transactions. A policy transaction is completed when the policy is bound and issued, or when the policy transaction is withdrawn. However, the policy may still require additional work. You can use *contingencies* to manage this additional work and take actions, including policy change or cancellation transactions, if conditions are not met in a timely manner.

You can associate activities, notes, and documents with a contingency.

See also

- *Application Guide*

Business rules for underwriting issue types

In PolicyCenter, underwriting authority has a new **Underwriting Rules** user interface which enables business users to create and modify underwriting rules. From the user interface, you can make underwriting rules available without restarting PolicyCenter.

In previous releases, underwriting rules were defined as underwriting issue types defined in the `UWIssueType` system table (`uw_issue_type.xml`). This table is replaced by the **Underwriting Rules** interface.

After upgrading from PolicyCenter 7.0 and 8.0, your legacy rules implemented in Gosu code will continue to work. You can choose whether to keep the Gosu implementation, or move the rule to the **Underwriting Rules** framework.

See also

- "Upgrading Underwriting Authority" in the *Configuration Upgrade Guide*
- *Application Guide*
- *Configuration Guide*

Integration with spotlight

PolicyCenter offers integration with Guidewire Spotlight. This integration provides a way to exchange data between PolicyCenter and Spotlight. This integration allows PolicyCenter to display and utilize results from location-specific risk assessments evaluated in Spotlight, as well as the inclusion of a map of the location.

See also

- *Application Guide*

Changed in the application in 9.0.0

Document management changes

The Document Assistant has been removed from the application because fewer browsers are supporting Java. As part of this change, there are new action buttons associated with documents, and there are more places that you can edit and upload documents. Additionally, it is now possible to upload more than one document at a time.

See also

- *Application Guide*

Payment screen improvements

This release contains many improvements to the **Payment** screen in the policy transaction wizards for each line of business. The major changes include:

- Improvements to performance and reliability.
- Up-front payments using a payment gateway – The agent can collect up-front payments, such as a down payment or deposit, using a payment gateway. The base configuration contains a stand-alone implementation.
- For all installment plans, even those not selected, you can view payment schedules with payment amounts and due dates generated by the billing system.

See also

- *Application Guide*

Changes in Rating Management

The changes in Rating Management include the following changes to rate books:

- **Rate Books** screen – In search results, there is a new **Activity** which displays **Export to spreadsheet in progress** while the rate book is being exported to spreadsheet. A similar message appears when exporting to XML.
- Rate Book screen – You can now export the rate book to spreadsheet format or XML. The exported file contains rate book details, included rate tables, and names of included rate routines. **Export to Spreadsheet** displays a progress bar. After export completes, you can select **Download to Spreadsheet**. In spreadsheet format, you can make changes to the included rate tables and import the rate tables back into PolicyCenter. In spreadsheet format, changes to rate book details and rate routines are ignored.

  The progress bar also appears when exporting to XML. XML files enable you to move rate books between systems. Do not edit or change XML file for the rate book.

See also

- *Application Guide*

Changes to Product Designer localization

The folder structure for localized files has been changed to make it compatible with PolicyCenter. Refer to `productdesigner/language-config/readme.html` for details on the new folder structure and for instructions on localizing Product Designer.

Login screen no longer saves user information

PolicyCenter 9.0.0 removes the **Keep me logged in** check box on the **Login** screen. In addition, PolicyCenter no longer stores a browser cookie that contains login information.

Change to VersionList property for base classes of revisioned entities

Revisioned entities in a `PolicyPeriod` have a `VersionList` property that contains a version list that is customized for the concrete revisioned entity type but adding a `VersionList` suffix. For example, for a `Building` entity instance, its `VersionList` property has the type `BuildingVersionList`. This property name and contents did not change, and none of that Gosu code is affected on upgrade.

However there were related changes:

- You might have written code that was general purpose such that it declared variables or function arguments as the base type `EffDated` or `EffDatedBase`. If so, there is a minor change if you accessed the `VersionList` property on either of those base classes. In version 9.0, the `VersionList` property on `EffDated` or `EffDatedBase` was renamed `VersionListUntyped`. Due to an unrelated change to base classes, during upgrade any use of `EffDatedBase` in your code must change to `EffDated`.
- On all concrete revisioned entity types (such as `Building`), from Gosu there is a new `VersionListUntyped` property. The value of `VersionListUntyped` is identical to the value of the `VersionList` property, but cast to the type `EffDatedVersionList`.

chapter 15

New and changed in configuration in 9.0.0

Changes to PolicyCenter configuration

Plugin replaces API for high volume quoting

For handling high volume quote requests, the Quoting Processor API has been deprecated and replaced by the Quoting Processor plugin.

See also

• *Configuration Guide*

Policy search criteria enhancement deprecated

In this release, the `PolicySearchCriteriaEnhancement` object and its extension class are deprecated. In its place, use `PolicyPeriodSearchCriteria.gs`, `PolicyPeriodQueryBuilder.gs`, and `PolicyPeriodSummaryQueryBuilder.gs` as applicable.

Modifying policy search is now similar to modifying search for accounts and activities, among others. To change policy search, modify the Gosu class, `gw.policy.PolicyPeriodSearchCriteria.gs`.

If you modified policy search in a previous release, you can reimplement your modifications in the `PolicyPeriodSearchCriteria.gs` class.

See also

• *Configuration Guide*

Entities referencing coverable and modifiable interfaces

In previous releases, an entity could provide an implementation of the `Modifiable` interface and separately provide an implementation of the `Coverable` interface. In this release, because both interfaces have a `PolicyLine` getter, you cannot implement both interfaces using different delegate classes. To implement both interfaces, you can implement a single adapter that implements both interfaces.

Product model structure uses CodeIdentifier

In PolicyCenter 8.0.4, the structure of some product model files changed. A new `CodeIdentifier` attribute was added to all product model patterns. The `CodeIdentifier` property is significantly longer than `PublicID`, thus enabling you to create more descriptive and recognizable identifiers for product model objects.

In PolicyCenter 8.0.4 and later 8.0 releases, the `CodeIdentifier` attribute is set to the same value as the `PublicID`.

Beginning in this release, Product Designer no longer assigns the same value to `CodeIdentifier` and `PublicID` for most new product model patterns. `PublicID` is now automatically generated and is no longer human readable. The corresponding pattern objects in PolicyCenter will also have diverging values for `CodeIdentifier` and `PublicID`. However, for `Product` and `PolicyLine` product model patterns, Product Designer will continue to assign the same value to the `CodeIdentifier` and `PublicID` properties. Their corresponding PolicyCenter object instances, `Policy` and `PolicyLine` respectively, reference the pattern by `PublicID`.

Background Information

In PolicyCenter 8.0.4, a `CodeIdentifier` property was added to product model patterns and the product model objects using these patterns. `CodeIdentifier` replaced `PublicID` as the identifier of product model objects in the Product Designer user interface and in Studio. Starting in PolicyCenter 9.0.0, when you create a new product model pattern, the `CodeIdentifier` of the pattern is set to the **Code** specified in Product Designer. The `PublicId` is set to an automatically generated value, with the exception of `Policy` and `PolicyLine` patterns, as noted above. (In PolicyCenter 8.0.4 and later 8.0 releases, Product Designer sets both `CodeIdentifier` and `PublicID` properties to the same value.)

In Product Designer screens starting with PolicyCenter 8.0.4, the **Code** field displays the `CodeIdentifier` property, not the `PublicID`, for product model objects.

Accessing product model pattern through generated type

Gosu expressions now access many product model objects using `CodeIdentifier` instead of `PublicID`. As a developer who writes code accessing product model objects, Studio makes this change largely transparent to you. Usually your code references a product model object through its generated type. This code does not require any change.

For example:

- The Liability - Bodily Injury and Property Damage coverage existed prior to the addition of the `CodeIdentifier` attribute. Your code accessing this coverage through the generated type `PALiabilityCov` does not need to change.

```
// Create line-level coverages
_liabilityCov = paLine.PALiabilityCov
_medPaymentCov = paLine.PAMedPayCov
```

In prior releases, `paLine.PALiabilityCov` accessed `PublicID`. In this release, `paLine.PALiabilityCov` accesses `CodeIdentifier`. The change is transparent because the upgrade trigger set the `CodeIdentifier` to the same value as `PublicID`.

- You add a new coverage, Glass, in PolicyCenter 9.0 with a `CodeIdentifier` of `HOPGlass`. Its `PublicID` is automatically generated and set to a value such as `x743oqa128`. Just as before, you can access the coverage through the generated type `HOPGlass`:

```
// Create line-level coverages
_glassCov = hopLine.HOPGlass
```

Code identifier necessary changes

Your implementation requires changes if you explicitly reference `PublicID` in product model objects. This is necessary when you directly access product model objects from the database. Use the `getByCodeIdentifier` or `getByPublicID` methods.

Some areas where you may need to make changes are:

• Product model scripts
• Gosu code
• System tables
• Product model web services

Product Model Scripts and Gosu Code

In guided editing, Studio displays the generated type, which is the value of `CodeIdentifier`. However, some code, such as getters for product model pattern lookup which evaluate each pattern in turn, must access the patterns by `CodeIdentifier` or `PublicID`. When accessing these identifiers, always use the `getByCodeIdentifier` or `getByPublicId` methods.

For example:

```
schedPropCov = ClausePatternLookup.getCoveragePatternByCodeIdentifier("PATowingLaborCov")
```

or:

```
newVehicle.createCoverage(ClausePatternLookup.getCoveragePatternByPublicID("PATowingLaborCov"))
```

System Tables

If a new system table has a `Code` column that references a product model pattern, Guidewire recommends that you use `CodeIdentifier` instead of `PublicID`.

How to work going forward with existing system tables depends on how those system tables are utilized. Most system tables with data columns that reference product model patterns generally referenced the code of the product model pattern. That code was previously mapped to `PublicId`, but now maps to `CodeIdentifier`. Ideally, you can continue to use code when entering data about new product model patterns. However, in order to do so, you must review how those system tables are utilized within PolicyCenter, in particular in system table verification and other Gosu code. If the system tables are accessed explicitly using `PublicId` you must do one of the following:

• Change the access in PolicyCenter to use `CodeIdentifier` instead of `PublicID`.
• Enter `PublicID` directly into the system table. However, unlike `CodeIdentifier`, the `PublicID` for newly created patterns is not human readable.

The first method is recommended, but you can use the second if the first is not possible.

If the existing system table only utilizes `Policy` or `PolicyLine` codes, nothing needs to be changed since `CodeIdentifier` and `PublicID` are the same for them.

> **Note:** For all product model patterns that existed prior to PolicyCenter 9.0.0, `PublicID` and `CodeIdentifier` will always be the same, so all methods will work for them.

Product Model Web Services

Generally speaking, it is best to identify product model objects using the property `CodeIdentifier` not `PublicID`. `ProductModelAPI` web service primarily identifies product model objects using the property `PublicID`. From an external system, you can get the `PublicId` for a product model object by calling the `ProductModelAPI` web service method `getPublicIdForCodeIdentifier`.

If you develop your own web services that manipulate product model types, Guidewire recommends using the property `CodeIdentifier` (not `PublicID`) for arguments and return values.

Changes to configuration files

PolicyCenter 9.0.0 contains three configuration files related to security.

`security-config.xml`	Writable	Use to define custom security handlers.

security-config-app.xml	Read-only	Defines internal security permissions for the PolicyCenter application.
security-config-pl.xml	Read-only	Defines internal security permissions used by the underlying application code.

Order of precedence

A permission declaration with the same `entity` and `permKey` values can appear in more than one of the files related to security. If a permission declaration appears in more than one file, the definition in `security-config.xml` takes precedence, followed by the definition in `security-config-app.xml`.

See also

- *System Administration Guide*

Changes to configuration APIs

This topic includes significant additions and changes to Gosu APIs, specifically, changes to core APIs in `gw.*`.

> **Note:** Changes to query builder, integration, or web service APIs are described in other parts of the Guidewire documentation, such as the *Integration Guide*.

New method initializes reinsurance for policy term

Added a new `PolicyPeriod.initializeReinsuranceForTerm` method which specifically initializes reinsurance for a policy period. The new method consults the `ReinsuranceConfigPlugin` in order to determine the correct value for the `PolicyTerm.GenerateReinsurables` flag. You can call the `initializeReinsuranceForTerm` method from Gosu code.

The new method is called from `PolicyPeriodPlugin` where appropriate, and also from two methods in `PolicyImpl` which create new periods: `createDraftPeriod` and `createLegacyPeriod`.

These code changes result in `ReinsuranceConfigPlugin#shouldGenerateReinsurables` being called later in the lifecycle of a policy period. This corrects an issue where the method was being called so early that the plugin could not do calculations based on things like the policy period start-of-term.

Changes to the data model

Changes to data model entity definitions

The following changes were made to the data model definitions:

Element	Change
`<column>`	• Added attribute `required`
`<delegate>`	• Added attribute `requiresType` • Added attribute `setterScriptability` • Removed attribute `effdatedOnly` • Removed attribute `requires` • Removed attribute `subpackage`
`<forceindex>`	• Removed attribute `db2`
`<foreignkey>`	• Added attribute `archivingOwner` • Added attribute `required` • Removed attribute `owner`
`<implementsEntity>`	• Removed attribute `adapter` • Attribute `impl` is no longer required

Element	Change
`<nonPersistentEntity>`	• Removed attribute subpackage
`<subtype>`	• Removed attribute subpackage
`<typekey>`	• Added attribute required
`<viewEntity>`	• Added subelement `<implementsInterface>` • Removed attribute subpackage

Entity types can declare required properties

You can now declare required properties for an entity type during object instantiation. To specify that a required entity property is required, set the attribute called `required` to the value `true`. The default is `false`. The `required` attribute is supported on the `<column>`, `<typekey>`, and `<foreignkey>` definitions of an entity type.

The `required` attribute does not overlap with the behavior of the existing attribute called `nullok`. The `nullok` attribute affects only the value of that property at commit time, which is the time the object is written to the database. In contrast, the `required` attribute affects only object instantiation. If the required property is an object type, the value must be non-null at the time of object instantiation or else Gosu throws an exception.

Code that uses the `new` operator to instantiate an entity type must include all required properties when constructing an instance of that entity as arguments to the constructor. If the constructor had other constructor arguments, the required properties appear after the other arguments.

During upgrade, this change may result in compilation errors in expressions that use the `new` operator to create entities with newly required properties that are omitted. To fix the compilation error, add the required fields after the other arguments.

For example, suppose an entity called `MyEntity` has one required parameter that is an `Address`. The code `new MyEntity()` is now a compilation error because the required property is missing from the argument list. To fix the error, get a reference to a non-null `Address` entity instance and pass it to the constructor, such as `new MyEntity(myAddress)`.

Some built-in entity types now have required fields. For new entity types, you can add your own custom required properties. However, you cannot add required fields to built-in entity types using data model extensions.

Required bundle arguments

As in previous releases, entity constructors contain an optional parameter that represents the bundle for the new entity instance. The value could be a bundle reference or another entity instance that is already in a writable bundle that you want to use.

If you provide the optional bundle parameter, the constructor argument list must include the bundle parameter before any required properties.

Required fields for product model entities

In PolicyCenter, all `Clause, PolicyLine, Exclusion`, and `PolicyCondition` entities now have their pattern code required. However, you cannot directly instantiate those entities nor their subtypes. To directly instantiate these entities with a specific pattern code, you must instantiate related classes in the `productmodel` package. These product model classes extend the entity types, and they have public constructors that support instantiation and omit the pattern code.

For example, to instantiate the business owners line, instantiate the type `productmodel.BOPLine`.

Typecode identifier customization

PolicyCenter changed how typelist codes become Gosu programmatic identifier codes that appear as properties on the typelist type. For example, typecode `abc` becomes programmatic identifier code `TC_ABC`.

The typecode definition element `<typecode>` has a new optional attribute called `identifierCode`. If you do not set `identifierCode`, PolicyCenter creates identifier codes using the same algorithm as in 8.0, but does not handle

conflicts. If you set `identifierCode`, PolicyCenter creates the identifier by using the `identifierCode`. Use `identifierCode` to resolve identifier name conflicts in a way that is meaningful to you.

See also

- *Gosu Reference Guide*

Delegate attribute requiresType replaces attribute effdatedOnly

There is a new feature in configuring entity delegates. To declare that the delegate must be implemented by an entity of a general type such as `retireable` or `versionable`, add the attribute `requiresType` on the `<delegate>` element.

Possible values for the `requiresType` attribute are the values of the `type` attribute on the `<entity>` element. See the *Configuration Guide*. Some of the general types extend others. For example, `editable` extends `versionable`, and `versionable` extends `keyable`. An entity can implement the delegate if the implementing entity is the specified type or one that extends it. For example, if the `requiresType` attribute is `keyable`, then an implementing entity could have type `keyable`, `versionable`, or `editable`.

For PolicyCenter, the `requiresType` attribute replaces the `effdatedOnly` attribute. PolicyCenter upgrades existing data model definition files to convert entities with `effdatedOnly` set to `true` to have `requiresType` set to value `effdated`.

Changes to Guidewire Studio

Studio updates do not require application updates

Guidewire Studio is installed as a self-contained application that you can update independently of PolicyCenter. When a new release of Studio is available, a message appears at the top of the Studio window. You can then have Studio automatically download and install the update, without needing to update your full PolicyCenter application.

See also

- *Configuration Guide*

Studio editors provide editable XML

Many visual editors in Guidewire Studio—such as the Page Configuration editor or Typelist editor—also contain a tab that displays the XML code that defines the item shown in the editor. The label of this tab has changed from **XML** to **Text**. You can now also edit the XML code on the **XML** tab to make changes to the item. Valid changes to the XML code are automatically reflected in the visual editor. Invalid XML code may cause the visual editor to display the item incorrectly.

Running Guidewire Studio

To run Guidewire Studio, run the command `gwb studio`. The script `studio` is no longer available.

Changes to configuration parameters

See the *Configuration Guide* for a discussion of the parameters that you can use to configure Guidewire PolicyCenter

Removed configuration parameters

The following configuration parameters have been removed in PolicyCenter 9.0:

- AllowDocumentAssistant
- AllowSoapWebServiceReferenceNamespaceCollisions
- AllowDivergentProductModelIdentifiers
- ClusterMulticastAddress
- ClusterMulticastPort
- ClusterMulticastTTL
- ClusterProtocolStack
- ClusterProtocolStackOption1
- ClusterProtocolStackOption2
- DocumentAssistantJNLP
- InputHelpTextOnFocus
- JGroupsClusterChannel
- JGroupsWatchdogHeartbeatIntervalSecs
- JGroupsWatchdogMissedHeartbeatsBeforeReset
- LoadSoapServicesOnStartup
- LoggerCategorySource
- LoggersShowLog4j
- LoggersShowPredefined
- UseDocumentAssistantToDisplayDocuments
- UseSafeBundleForWebServiceOperations
- WizardNextShortcut
- WizardPrevNextButtonsVisible
- WizardPrevShortcut
- WSIEntityExpansion
- WorkQueueBulkInsertIdAllowance

Added configuration parameters

The following configuration parameters have been added in PolicyCenter 9.0:

Parameter	More information
BillingSystemArchiveEnabled	*Configuration Guide*
BillingSystemArchivePolicyPeriodDays	*Configuration Guide*
BizRulesDeploymentEnabled	*Configuration Guide*
BizRulesDeploymentId	*Configuration Guide*
BizRulesLeafSearchNumOfHops	*Configuration Guide*
DisableSequenceUtil	*Configuration Guide*
FinalDocumentsNotEditable	*Configuration Guide*
MaximumFileUploadCount	*Configuration Guide*
MaximumTotalUploadSize	*Configuration Guide*
MultipleLocationRiskAssessmentEnabled	*Configuration Guide*
PrintCsvDefaultBaseFileExtension	*Configuration Guide*
PrintCsvMimeType	*Configuration Guide*

Parameter	More information
PrintDefaultBaseFileName	*Configuration Guide*
PrintPdfDefaultBaseFileExtension	*Configuration Guide*
PrintPdfMimeType	*Configuration Guide*
PurgeRateBookExportResultEnabled	*Configuration Guide*
PurgeRiskAssessmentTempStoreDays	*Configuration Guide*
RateBookExportResultAgeForPurging	*Configuration Guide*
RiskAssessmentIntegrationEnabled	*Configuration Guide*
RiskAssessmentThumbnailMapEnabled	*Configuration Guide*
SingleLocationRiskAssessmentEnabled	*Configuration Guide*
SmallRateTableRowLimit	*Configuration Guide*
SpotlightLoginURL	*Configuration Guide*
SpotlightRiskAssessmentServiceURL	*Configuration Guide*
SpotlightInteractiveServiceURL	*Configuration Guide*
SpotlightThumbnailMapURL	*Configuration Guide*

Changed configuration parameters

The following configuration parameters have been changed in PolicyCenter 9.0:

Parameter	Change
ClusterMemberPurgeDaysOld	The minimum value is now 1. The parameter is now required.
DefaultApplicationCurrency	Can no longer be set by environment.
DefaultApplicationLocale	Can no longer be set by environment.
DefaultApplicationCurrency	Can no longer be set by environment.
MigrateToLargeIDsAndDatetime2	Default changed from `false` to `true`.
PDFMergeHandlerLicenseKey	Can no longer be set by environment.
UISkin	Default changed from `Titanium` to `theme-9`.

Changes related to PCF files

Use type-specific input and cell PCF elements instead of generic ones

The PCF elements Input and Cell are no longer allowed. Instead, use input and cell elements that are associated with particular value types. For example, Date Input, Text Input, Date Cell, Text Cell, and so on.

Accordingly, the new `valueType` property of an input or cell element is required. In most cases, `valueType` has a default value based on the type of element; for example, `valueType` for a Text Input defaults to `java.lang.String`. In cases where there is no default possible, such as for TypeKey Input or Range Input, you must set `valueType` explicitly.

PCF symbols are case-sensitive

PCF elements, properties, and other symbols are now case-sensitive. Use the proper case when referring to them programmatically, such as in Gosu code or other configuration files.

Other PCF changes

The following are changes to the configuration of PCF files:

- The Search Panel element now requires a `searchCriteriaType` property.
- The Data Series and Dual Axis Data Series elements now require a `dataValueType` property.
- A Range Input that maps to multiselects must now set the `multiSelect` property to `true`.
- Iterator elements now require a `valueType` property.
- The Variable element now requires a `type` property, even when the `initialValue` property is set.

Number And DateTime Types have been replaced

To conform with Gosu type changes, the PolicyCenter make the following changes to types used in PCF files:

Removed type	Replace with type
Number	java.lang.Double
DateTime	java.util.Date

PCF changes due to upgrade

The configuration upgrade tools perform necessary modifications to PCF files.

Changes related to display keys

Display keys are case-sensitive

Display keys are now case-sensitive. Use the proper case when referring to them programmatically, such as in Gosu code or other configuration files.

Display key folder and file structure is flattened

In Guidewire Studio, the **configuration**→**config**→**Localizations** folder, which contained the `display.properties` files, no longer contains subfolders for each locale. Instead, the locale is now included in the file name, and all files are located at the top level of the **Localizations** folder. For example, the file `en_US/display.properties` is now named `display_en_US.properties`.

The underlying directory and file structure in the file system also reflects these changes.

Access display keys with DisplayKey.get

The use of `displaykey.*` to refer to display keys in Gosu, PCF, and other configuration files is no longer valid.

To refer to display keys, use the method `DisplayKey.get`. For example:

Old reference	New reference
displaykey.Web.Address.Default.PostalCode	DisplayKey.get("Web.Address.Default.PostalCode")

You must include the following `uses` statement in any Gosu program that calls `DisplayKey.get`:

```
uses gw.api.locale.DisplayKey
```

Changed in globalization in 9.0.0

Localizations Folder in Studio has no subfolders

In PolicyCenter 8.0, the **Localizations** folder in Guidewire Studio had subfolders for each location, such as **en_US**, **fr_FR**, **ja_JP**, and so on. Each of these location subfolders held property files, like `display.properties` and `localization.xml`, for a location,

In PolicyCenter 9.0, the **Localizations** folder has no subfolders. Each property file has a suffix to indicate the location. For string definition files for example, the `display_en_US.properties` file is directly under **Localizations** in 9.0, and is not in a subfolder.

See also

• *Globalization Guide*

New and changed in installation and upgrade in 9.0.0

Changed in installation in 9.0.0

Apache ant installation requirement removed

Guidewire no longer requires Apache Ant for general configuration and administration tasks. It is only necessary to install Ant if you plan to develop OSGi plugins using the IntelliJ IDEA with OSGi Editor.

See also

• *Integration Guide*

New in upgrade in 9.0.0

Upgrade Tools

The procedure for upgrading to PolicyCenter 9 has changed significantly. In PolicyCenter 9, there are new tools available for performing the configuration upgrade. Do not rely on your past knowledge of the upgrade process to begin the upgrade procedure without reviewing the new upgrade tools and documentation.

The InsuranceSuite Upgrade Tools include the following:

Preupgrade Tool – A command-line tool that prepares configuration files for upgrade in the source configuration.

This tool runs upgrade steps in the customer configuration (the older version you are upgrading from). For example, if you are upgrading from PolicyCenter 8.0 to PolicyCenter 9.0, the preupgrade tool will modify the PolicyCenter 8.0 configuration.

Upgrade Tool – A command-line tool that runs a series of automated configuration upgrade steps in the target configuration.

This tool runs upgrade steps after copying the customer configuration to a new temporary location in the target version.

Merge Tracker Tool – A Guidewire Studio plugin that tracks progress and assists in completing the manual part of the configuration upgrade process.

Smart Merge Tool – A context aware three-way merge tool for InsuranceSuite application configuration files. As the name suggests, this tool understands the semantics of InsuranceSuite configuration files. Smart Merge can be launched from Merge Tracker or from the command line.

Upgrade Inspections – A set of Guidewire Studio code inspections that identify changes needed after automated configuration upgrade changes are complete. These configuration upgrade specific inspections can be run from the Merge Tracker toolbar from the Merge Tracker right-click menu.

For more information see the *PolicyCenter Configuration Upgrade Guide.*

To upgrade your PolicyCenter Configuration, you must first download the *PolicyCenter Configuration Upgrade Guide* and the InsuranceSuite Upgrade Tools, as follows.

Download the InsuranceSuite upgrade tools and documentation

About this task

Guidewire does not include the InsuranceSuite Upgrade Tools in the PolicyCenter installation.

You can access the latest version of the InsuranceSuite Upgrade Tools and the *PolicyCenter* Configuration Upgrade Guide using the following instructions.

Procedure

1. Visit the Guidewire Community:
   - Guidewire Customers - `https://community.guidewire.com`
   - Guidewire Partners - `https://partner.guidewire.com`
2. Select the **Resources** tab.
3. Under **Product Group**, select **InsuranceSuite**.
4. Under **Product**, select **InsuranceSuite Upgrade Tools**.
5. Under **Release**, select the most recent release.
6. Download the software, release notes, and documentation by clicking each corresponding download link.

Rolling configuration upgrade of PolicyCenter cluster members

It is now possible to update each application instance in a PolicyCenter cluster individually. Guidewire calls this type of upgrade a rolling upgrade or a configuration deployment. With a configuration deployment, it is possible to stop an individual PolicyCenter server and deploy configuration changes to it without stopping the other servers in the cluster. This type of upgrade is only possible with select application configuration changes.

See also

- *System Administration Guide*

Custom application version labels

It is now possible to create custom application version labels that show in PolicyCenter, for example, in the Server Tools **Upgrade and Versions** screen. Use these version labels to identify a specific application version during an upgrade process.

See also

- *System Administration Guide*

Upgrade-related system tools command options

There are new `system_tools` command options that relate to full (database) upgrade and rolling (configuration) upgrade. They are:

- `-cancelshutdown`
- `-completefailedfailover`
- `-components`
- `-evenifincluster`
- `-nodesfailed`
- `-nodes`
- `-requestcomponenttransfer`
- `-scheduleshutdown`
- `-shutdowndelay`
- `-startfullupgrade`
- `-terminatebatchprocesses`
- `-verifyconfig`

See also

- "Administration command prompt tools" on page 102
- *System Administration Guide*

Upgrade flag required for upgrade

To prevent the unexpected upgrade of a PolicyCenter server, Guidewire requires that you set a PolicyCenter upgrade flag before starting the upgrade process:

- To start a full upgrade, you must first click **Start Full Upgrade** in the Server Tools **Upgrade and Versions** screen on any cluster member. At the completion of the upgrade, PolicyCenter removes the flag automatically.
- To start a rolling upgrade, you must first click **Start Rolling Upgrade** in the Server Tools **Upgrade and Versions** screen on any cluster member. After completing the upgrade of all servers in the PolicyCenter cluster, you must click **Rolling Upgrade Complete** on the **Upgrade and Versions** screen to remove the upgrade flag

See also

- *System Administration Guide*

Changed in upgrade in 9.0.0

Changes to the database configuration file

Guidewire has made the following changes to file `database-config.xml`:
- "Automatic database upgrade removed" on page 99
- "Index statistics element removed" on page 100
- "Attribute numbuckets on histogram statistics element now required" on page 100

Automatic database upgrade removed

Guidewire has removed the `autoupgrade` attribute on the `<database>` element in `database-config.xml`. Instead, you must explicitly set an upgrade flag before starting either a full or rolling upgrade.

See also

- "Upgrade flag required for upgrade" on page 99

Index statistics element removed

Guidewire has removed the `<indexstatistics>` element from file `database-config.xml`. This change affects the following PolicyCenter functionality:

- The `system_tools -getdbstatisticsstatements` command no longer generates separate SQL statements for index statistics.
- The Server Tools **Database Catalog Statistics Information** download reports no longer contain statistics statements for indexes.

Attribute numbuckets on histogram statistics element now required

Guidewire now requires that you specify a value for attribute `numbuckets` on the `<histogramstatistics>` element in file `database-config.xml`. If you do not provide a value for this attribute, the PolicyCenter server generates an error and refuses to start.

New and changed in system administration in 9.0.0

New in system administration in 9.0.0

Batch process load balancing

In 9.0, Guidewire provides a default implementation of a lightweight prioritization strategy to use to load balance the running of batch processes within the PolicyCenter cluster. File `batch-process-config.xml` provides a way to control the execution of multiple batch processes.

See also

- *System Administration Guide*

Messaging and startable server load balancing

In 9.0, Guidewire provides several default strategies for managing the load balancing of messaging destinations and startable services (plugins) within the PolicyCenter cluster. Use the following Gosu class to implement load balancing of these items.

```
DefaultBackgroundTaskLoadBalancingPlugin
```

See also

- *System Administration Guide*

System property pass through option

In 9.0, it is possible to specify Java system properties with the PolicyCenter command prompt tools using the following syntax.

```
-Dgw.passthrough.javaProperty=someValue
```

See also

- *System Administration Guide*

Graceful server shutdown

In 9.0, there is no longer a single batch server. Thus, the order in which you shutdown the cluster servers becomes important.

See also

- *System Administration Guide*

New in Guidewire server tools

In 9.0, Guidewire has added the following server tool screens accessible to system administrators.

Screen	Description
Oracle Outlines	Guidewire has added an **Oracle Outlines** screen to the **Info Pages** of the PolicyCenter**Server Tools**. You use this screen to view information about stored outlines in the Oracle database. This screen is available only if the database is Oracle. See the *System Administration Guide* for more information.

Changed in system administration in 9.0.0

Administration command prompt tools

The following list describes changes to the administration command prompt tools in 9.0.

Command	Command options
system_tools	Added the following command options: • `-cancelshutdown` • `-completefailedfailover` • `-components` • `-evenifincluster` • `-nodefailed` • `-nodes` • `-requestcomponenttranser` • `-scheduleshutdown` • `-startfullupgrade` • `-verifyconfig` Many of these new command options work specifically with upgrade, for example, `startfullupgrade`. **See also** • *System Administration Guide*
table_import	Added the following command option: • `-numthreadsintegritychecking` **See also** • *System Administration Guide*
usage_tools	Guidewire has removed this command entirely.

Changes to logging infrastructure

Guidewire has made the following changes to the PolicyCenter 9.0.0 logging infrastructure in 9.0.

Logging parameters removed

Guidewire has removed the following logging-related configuration parameters:

- `LoggerCategorySource`
- `LoggersShowLog4j`
- `LoggersShowPredefined`

This change affects the behavior of the Server tools **Set Log Level** screen. See the *System Administration Guide* for more information.

Logger class deprecated

Guidewire has deprecated class `gw.api.util.Logger`. Instead, use the following class to send information to application log files or the console window for debugging purposes:

```
gw.api.system.PCLoggerCategory
```

Modifications to the server registry element in config.xml

In 9.0, Guidewire has modified the server `<registry>` element in file `config.xml` so that:

- You define the valid and available server roles using the `roles` attribute on the `<registry>` element
- You define the environment, server ID, and roles assigned to a server using attributes on the `<server>` subelement of the `<registry>` element
- You define system properties for use with the servers in a PolicyCenter cluster using the `<systemproperties>` subelement of the `<registry>` element

See also

- *System Administration Guide*

Removal of support for JGroups clustering

In 9.0, Guidewire support for clustering no longer relies on JGroups. Instead, PolicyCenter clusters use PPP protocol for direct server-to-server communication and leverage the database for distributing broadcast messages.

As a consequence, the following configuration parameters are obsolete:

- `ClusterMulticastAddress`
- `ClusterMulticastPort`
- `ClusterMulticastTTL`
- `ClusterProtocolStack`
- `ClusterProtocolStackOption1`
- `ClusterProtocolStackOption2`
- `JGroupsClusterChannel`
- `JGroupsWatchdogHeartbeatIntervalSecs`
- `JGroupsWatchdoMissedHeartbeatsBeforeReset`

In place of these configuration parameters, use the following mechanisms to configure clustering:

- Add configuration parameters to cluster plugin implementations
- Use system property overrides at server startup

See also

- *System Administration Guide*

Removal of dedicated batch server requirement

In previous Guidewire PolicyCenter releases, each PolicyCenter cluster required a single server dedicated to batch processing. The other servers in the cluster handled various other tasks.

With the release of PolicyCenter 9.0.0, you can to the following:

- Assign one or more roles to each individual server in the PolicyCenter cluster
- Add custom server roles to those provided in the base configuration, which are `batch`, `messaging`, `scheduler`, `startable`, `workqueue`, `ui`
- Assign multiple servers the `batch` server role

See also

- "Modifications to the server registry element in config.xml" on page 103
- *System Administration Guide*

Setting server system properties

In 9.0, Guidewire has modified the way that you set server system properties using the `-D` JVM option syntax. The exact command syntax depends on whether you are using the development QuickStart server (Jetty) or a production-level server such as Tomcat, for example:

Server	JVM option syntax
QuickStart (Jetty)	`-Denv=…` `-Dserverid=…`
Tomcat	`-Dgw.pc.env=…` `-Dgw.pc.serverid=…`

See also

- *System Administration Guide*

Changes in Guidewire server tools

In 9.0, Guidewire has made the following changes to the server tool screens accessible to system administrators.

Screen	Change
Batch Process Info	Guidewire has made changes to the way this screen presents information on batch processes and work queues. See the *System Administration Guide* for details.
Cluster Info	Guidewire has renamed the **Cluster Info** screen to **Cluster**. From **Cluster**, you can navigate to a **Cluster Member** and a **Cluster Component** screen. Previously, to view this screen, you had to enable clustering through the `ClusteringEnabled` configuration parameter in `config.xml`. This is no longer true. The clustering screens are visible to all with the proper logon credentials for Server Tools. See the *System Administration Guide* for more information.
Loaded Gosu Classes	Guidewire no longer supports the functionality supplied by the **Loaded Gosu Class** screen, even though the screen still exists.
Set Log Level	Guidewire has removed the logging configuration parameters that controlled the behavior of this screen. Thus, it is no longer possible to configure this screen to show different hierarchies of logger categories. See the *System Administration Guide* for more information.
Startable Plugins	Guidewire has renamed the **Startable Plugins** screen to **Startable Services**. In addition, the screen summary information now includes the host on which the startable services is running. See the *System Administration Guide* for more information.
Upgrade Info	Guidewire has renamed and modified the **Upgrade Info** screen, which now has the name **Upgrade and Versions**. The updated screen is no longer part of the Server Tools **Info Pages** screens. It is now its own separate category on the Server Tools screen. See the *System Administration Guide* for more information.

Screen	Change
Work Queue Info	Guidewire has slightly modified the name and functionality of the buttons that appear in the **Actions** column of the **Work Queue** table on the **Work Queue Info** screen. In addition, the **Work Queue** table now contains a **Cluster-wide State** column. See the *System Administration Guide* for more information.

New and changed in Gosu in 9.0.0

New in Gosu in 9.0.0

Structural types

Use structural typing to write code that works with objects with similar features but no common inheritance and interface declarations. Define structural types similar to defining interfaces, by specifying the common properties and method signatures. However, use the `structure` keyword, not the `interface` keyword.

Structural types are statically weaker than interfaces regarding the amount of enforced type information, but their flexibility supports situations where interfaces are ineffective or impossible. Structural types extend static typing to include a broader set of real world situations but still support concise code that catches common coding problems at compile time.

See also

• *Gosu Reference Guide*

Dynamic types and expando objects

Gosu supports dynamic language features that permit coding styles similar to non-statically typed languages. The `dynamic` type allows dynamic assignment as well as dynamic dispatch of property access and method invocation. Gosu also provides support for *expando objects*, which simplify dynamic property and method access for typical code contexts.

See also

• *Gosu Reference Guide*

Checked arithmetic

In PolicyCenter version 8.0 and earlier versions, numeric values could exceed their defined bounds in arithmetic operations. For example, if you multiplied the maximum integer value by 2, the result by definition exceeds the range of integer values. Not only is the result value incorrect, the result could be positive when you expect it to be negative, or negative when you expect it to be positive. Because no Gosu exceptions occurred, it was difficult to protect against overflow errors, which could cause unexpected behavior or security issues.

In version 9.0, Gosu includes an optional feature called *checked arithmetic*. If checked arithmetic is enabled, Gosu behavior for the standard arithmetic operators changes for addition, subtraction, and multiplication. In nearly all cases, the result is the same and the behavior is the same. In the rare case that arithmetic overflow occurs, Gosu

throws the exception `ArithmeticException`. To enable checked arithmetic, set Java system property `checkedArithmetic` to `true`.

Gosu checked arithmetic includes protection only for the operators `+`, `-`, and `*`. There is no protection for division, which only affects the expression `Integer.MIN_VALUE / -1`.

There are special cases in which arithmetic overflow behaviors are desirable for operators `+`, `-`, and `*`. For example, some common hash algorithms rely on arithmetic overflow. To handle typical use cases in overridden `hashCode` methods, Gosu always compiles `hashCode` methods with checked arithmetic disabled.

For other cases in which arithmetic overflow behaviors are desirable, you can use three new Gosu operators that ensure unchecked arithmetic independent of the Java system property `checkedArithmetic`. The new operators are the standard arithmetic operators prefaced with an exclamation point character: `!+`, `!-`, and `!*`.

For example, with Java system property `checkedArithmetic` to `true`:

```
var four = new Integer(4)
var y = Integer.MAX_VALUE  * 2 + four // This line throws ArithmeticException
```

In contrast, the following example uses the unchecked arithmetic operator `!*`, which is only for special circumstances in which overflow is desirable:

```
var four = new Integer(4)
var x = Integer.MAX_VALUE !* 2 + four // This line does not throw ArithmeticException
print(x) // print "2"
```

Because the arithmetic in the second example is unchecked, the result that prints 2 successfully is possibly unexpected and invalid.

JSON support

Gosu has native support for JavaScript Object Notation (JSON) data format. JSON is an open-standard format that uses human-readable text to transmit data objects consisting of attribute-value pairs, hierarchical data structures, and arrays. Web sites often send or receive small amounts of JSON data as a lightweight alternative to the XML standard. Creation of JSON data and parsing can be implemented in any language or operating system, and is especially popular due to efficient web browser client-side parsing in JavaScript.

See also

• *Gosu Reference Guide*

Command-prompt arguments for a Gosu program

You can run a Gosu program that retrieves values from the command prompt. To access the values in the Gosu program, use the `Gosu.RawArgs` property, which is an array of `java.lang.String`.

See also

•

Changed in Gosu in 9.0.0

See also

• For changes in Gosu APIs that relate to the data model, see "Changes to the data model" on page 88.

Entity and typelist case sensitivity changes

In version 8.0, the core Gosu language was case sensitive but types for entities and typelists remained case insensitive for many contexts.

In version 9.0, entity and typelist types are case sensitive in general for the following:

- Entity type names
- Entity method names
- Entity property names
- Typelist names
- Typelist value names

For most required changes, these differences cause compilation errors you can easily find and fix in Studio. For example, you might need to change `entity.address` to `entity.Address`. Fix all compilation errors.

Some required changes do not manifest as compilation errors but are still critical to change in version 9.0. If you pass any of the case sensitive names as `String` values to any API, it is best practice to ensure that all `String` values correctly capitalize the text. Otherwise, you see no compilation errors in Studio but your code generates run time errors that might be difficult or impossible to debug.

The following APIs changed to require case sensitive `String` values:

- Use of array syntax for entity property names: `object[PropertyName]` syntax. For example, `myObject["PrimaryContact"]`.
- Use of Gosu reflection APIs, such as methods of the class `gw.lang.reflect.ReflectUtil`.

Even though the property names are case-sensitive now, on an entity type you cannot create two entity properties with names that differ only by case. Similarly, in one typelist you cannot create two typecodes with names that differ only by their case.

Some APIs remain case insensitive

The following entity and typelist APIs remain case insensitive for `String` values:

- Database query API methods that take `String` values to identify fields
- Entity methods that get and set properties with a `String` value for the property name.
- Entity methods that check the contents of a bundle: `isFieldChanged`, `getOriginalValue`, `getAddedArrayElements`, `getChangedArrayElements`, `getRemovedArrayElements`, `isArrayElementChanged`, `isArrayElementAddedOrRemoved`. See the *Gosu Reference Guide*.
- API methods that find a typekey by its code (as a String) is case-insensitive, for example:

```
typekey.TYPELISTNAME.get("CODE").
```

- Entity path expressions (sometimes called bean path expressions) in view entity data model definitions remain case insensitive. See the *Configuration Guide*. The following example shows an entity path in the `path` attribute:

```
<viewEntityName name="RelActAssignedUserName" path="RelatedActivity.AssignedUser"/>
```

Entity and typelist type system changes

Entity supertype behavior changes

PolicyCenter changed how Gosu handles supertypes in the entity type information property `Supertype`. This is the property accessed with an entity type such as `Address.Type.Supertype` or with an entity instance as `myObj.IntrinsicType.Supertype`.

In version 8.0:

- If an entity has a supertype, the `Supertype` property is the supertype entity as type `gw.entity.IEntityType`.
- If an entity has no supertype, the `Supertype` property is `null`.

In version 9.0:

- If an entity has a supertype, the `Supertype` property is the supertype entity as type `gw.lang.reflect.IType`.
- If an entity has no supertype, the `Supertype` property is `java.lang.Object`.

If you want the version 8.0 behavior, change your code to use the new property `Supertype_Entity` instead of `Supertype`. This new property is intended only for upgrading existing code. Do not use it for new code.

In type information, entity types are never final

In 9.0, PolicyCenter changed how Gosu handles entity finality from Gosu using the entity type information property `Final`. This is the property that you access with an entity type such as `Address.Type.Final` or with an entity instance as `myObj.IntrinsicType.Final`.

In version 8.0:

- If the data model definition declared an entity with the `final` attribute set to `true`, the `Final` property has the value `true`. Otherwise it has the value `false`.

In version 9.0:

- The `Final` property always has the value `false`.

If you want the version 8.0 behavior, change your code to use the new property `Final_Entity` instead of `Final`. This new property is intended only for upgrading existing code. Do not use it for new code.

Typelist types are not Java enumerations

In 9.0, PolicyCenter typelist types are no longer directly equivalent to Java enumerations from Gosu APIs. This changes the behavior of the typelist type information property `Enum`. This is the property that you access with code such as `ContactType.Type.Enum`.

In version 8.0:

- The typelist type information property `Enum` returns `true`
- In method declaration values, you could define default values for typecode parameters, in other words parameters of typelist types. See the *Gosu Reference Guide*.

In version 9.0:

- The typelist type information property `Enum` returns `false`. Search your code for uses of the `Enum` property that depend on this behavior.
- In method declaration values, you cannot define default values for arguments of typecodes. If you want a similar behavior, define an additional method declaration that does not contain that typelist type argument. From the new method, call the original method and pass your original default typecode as an argument.

Entity and typelist types extend the IJavaType interface

In 9.0, All entity and typelist types now extend the interface `gw.lang.reflect.java.IJavaType`. During upgrade, search your code for any code that checks if a type is an instance of `IJavaType`.

Base entities are removed

In version 8.0 and earlier releases, you could reference an entity type in Gosu using either its natural name (=as defined in the data dictionary or with the `Base` suffix. In version 9.0, base entities are gone.

In regular code in Studio, any code that referenced a base entity is now a compiler error, and can be fixed by removing the `Base` suffix. For example, if any Gosu code references `AddressBase`, change that code to instead say `Address`.

You may have document templates or other Gosu templates that do not automatically show compiler errors in Studio. You must review your document template descriptors to find any Gosu code that uses entity names with the `Base` suffix.

Entity instance static literal syntax removed

In version 8.0, Guidewire deprecated entity instance static literals using IDs. In 9.0, Guidewire no longer supports this syntax in any way. Use query builder APIs instead. Use the PolicyCenter upgrade tools to update all existing code in Studio to the new style automatically.

Entity instance static literals with IDs look similar to object instantiation but do not use the `new` operator:

```
entityType(entityId)
```

Replace these expressions with query builder expressions such as:

```
Query.make(Policy).compare(Policy#PublicID, Relop.Equals, myPublicId).select().AtMostOneRow
```

See also

- *Gosu Reference Guide*

Property getter methods required for new script parameters

In 9.0, To make a new script parameter accessible from Gosu code, you must create a getter method for the new parameter in the script parameters enhancement file.

See also

- *Configuration Guide*

Normalization of Gosu code

As part of automatic upgrade, PolicyCenter 9.0 reformats your Gosu code according to standard rules for spacing and whitespace. This process is called normalization. This process simplifies the other upgrade steps related to Gosu code, and makes your code easier to read.

Accessing type metadata properties and methods

Gosu exposes some type metadata as methods and properties associated with each Gosu type.

In version 8.0, methods (such as `isAssignableFrom`) or properties (such as `Supertype`) were moved to the `Type` property. The corresponding methods and properties directly on the type were deprecated.

In version 9.0, the deprecated methods and properties were removed and are now compilation errors. To fix the error, before the metadata method or property, add the word `Type` followed by a period character. Use the PolicyCenter upgrade tools to update existing code to the new style.

For example, this type metadata method call worked, but was deprecated in version 8.0.

```
String.isAssignableFrom(Number)
```

That statement is a compilation error in version 9.0. Instead, use the following code.

```
String.Type.isAssignableFrom(Number)
```

Also in version 9.0, type metadata methods and properties specific to Gosu entities and typekeys are accessible from the item's `TYPE` property and `get` method. The code sample below demonstrates the appropriate syntax.

```
var myTypekeys = new Typekey[10]
NoteSecurityType.TYPE.get().getTypekeysByCategories(myTypekeys)
```

The `TYPE` property is available only on Gosu entity and typekey data types. The PolicyCenter upgrade tools can update existing code to the new syntax.

Number, DateTime, Bean, List Types removed

The Gosu language no longer contains the types `Number`, `DateTime`, `Bean`, or `List`. In earlier releases, these were type aliases for other types. In version 9.0, you must use new names for these types. Use the PolicyCenter upgrade tools to update all existing code in Studio to the new style automatically.

The following table lists what types to use for any new code that you write

Removed type	Replace with type
Number	java.lang.Double

Removed type	Replace with type
DateTime	java.util.Date
Bean	Object
List (when not qualified with a package)	For interfaces, replace with java.util.List. For classes, replace with java.util.ArrayList

See also

- "Changes to list instantiation" on page 112

Changes to list instantiation

In 9.0, Guidewire made the following changes to the List type in Gosu:

- The type List (with no package) is no longer a type alias for java.util.List. See "Number, DateTime, Bean, List Types removed" on page 111.
- In new expressions, you cannot use the interface type List or java.util.List to instantiate a list as a shortcut to create an instance of java.util.ArrayList. If you get related compilation errors in new expressions, change your new expression to explicitly create an instance of ArrayList instead.

Find Expressions removed

In 9.0, the Gosu language no longer supports find expressions. Gosu retains equivalent functionality for retrieving information from the application database with the query builder APIs. Configuration upgrade tools for Version 9.0 convert legacy find expressions to equivalent query builder code.

For example, the find expression in the following example Gosu code from earlier versions of PolicyCenter returns Address instances that are located in the city of Chicago.

```
// Query the database for addresses in Chicago.
var findQuery = find (a in Address where (a.City == "Chicago"))

 // Print the addresses in the query result.
for (a in findQuery) {
  print (a.AddressLine1 + " / " + a.City + ", " + a.PostalCode)
}
```

After upgrade, the preceding find expression is converted to the following query builder code, which also returns addresses that are located in the city of Chicago.

```
uses gw.api.database.Query

 // Query the database for addresses in Chicago.
var findQuery = Query.make(Address).compare("City", Equals, "Chicago").select()

 // Print the addresses in the query result.
for (a in findQuery) {
  print (a.AddressLine1 + " / " + a.City + ", " + a.PostalCode)
}
```

See also

- *Gosu Reference Guide*

Block syntax in order by removed from query API

In 9.0, the Gosu language no longer supports block syntax as an argument to the query builder orderBy method. Gosu retains equivalent functionality for ordering rows from the application database by using a method that returns an object that implements the IQuerySelectColumn interface. Configuration upgrade tools for Version 9.0 convert legacy arguments for the query builder orderBy method to equivalent query builder code.

For example, the query builder `orderBy` method in the following example Gosu code from earlier versions of PolicyCenter returns `Address` instances that are ordered by their postal code.

```
uses gw.api.database.Query

 var select = Query.make(Address).select()
// Specify to sort the result by postal code.
select.orderBy(\ row -> row.PostalCode)
```

After upgrade, the preceding block is converted to the following query builder code, which also returns addresses that are ordered by their postal code.

```
uses gw.api.database.Query
uses gw.api.database.QuerySelectColumns
uses gw.api.path.Paths

 var select = Query.make(Address).select()
// Specify to sort the result by postal code.
select.orderBy(QuerySelectColumns.path(Paths.make(Address#PostalCode)
```

See also

• *Gosu Reference Guide*

Block syntax in row query select removed from query API

In 9.0, the Gosu language no longer supports block syntax as an argument to the query builder `select` method. Gosu retains equivalent functionality for selecting columns from the application database for a row query. The `select` method now accepts an array of objects that implement the `IQuerySelectColumn` interface. Configuration upgrade tools for Version 9.0 convert legacy arguments for the query builder `select` method to equivalent query builder code.

For example, the query builder `select` method in the following example Gosu code from earlier versions of BillingCenter returns rows that contain `ChargeAmount` and `ProducerName` columns.

```
uses gw.api.database.Query

 // Query for invoice items.
var queryInvoiceItems = Query.make(InvoiceItem)

 var resultInvoiceItems = queryInvoiceItems.select(
  \ invoiceItem -> ({
      "ChargeAmount" -> invoiceItem.Charge.Amount,
      "ProducerName" -> invoiceItem.PolicyPeriod.PrimaryPolicyCommission.ProducerCode.Producer.Name
    })
  )

 // With a row query, the result contains only the data you need.
for (row in resultInvoiceItems) {
  var chargeAmount = row.get("ChargeAmount")
  var producerName = row.get("ProducerName")

   print(chargeAmount + " charged to producer " + producerName)
}
```

After upgrade, the preceding block is converted to the following query builder code, which also returns rows that contain `ChargeAmount` and `ProducerName` columns.

```
uses gw.api.database.Query
uses gw.api.database.QuerySelectColumns
uses gw.api.path.Paths

 // Query for invoice items.
var queryInvoiceItems = Query.make(InvoiceItem)

 var resultInvoiceItems = queryInvoiceItems.select({
    QuerySelectColumns.pathWithAlias("Charge", Paths.make(InvoiceItem#Charge)),
    QuerySelectColumns.pathWithAlias("ProducerName",
```

```
            Paths.make(InvoiceItem#PolicyPeriod,
                    PolicyPeriod#PrimaryPolicyCommission,
                    PolicyCommission#ProducerCode,
                    ProducerCode#Producer,
                    Producer#Name))
    })

 // With a row query, the result contains only the data you need.
 for (row in resultInvoiceItems) {
   var charge = row.getColumn("Charge")
   var producerName = row.getColumn("ProducerName")

    print(charge + " charged to producer " + producerName)
 }
```

See also

- *Gosu Reference Guide*

Intersect deprecated in query API

In version 9.0, the `intersect` method on the `Query` class is deprecated. Query intersection often causes a performance problem. A better choice is to use a more efficient query type. For example, if both sides of the `INTERSECT` clause query the same table, use a single query. Use an `AND` operator to combine the restrictions from both sides of the `INTERSECT` to restrict the result.

See also

- *Gosu Reference Guide*

Java style constructors removed

Guidewire deprecated support for Java-style constructors in PolicyCenter 3.0.

In version 9.0, Gosu removes support for Java-style constructor declarations entirely. For example, for a class called `ClassA`, instead of naming the constructor `ClassA`, use the Gosu native constructor name `construct`. Use the PolicyCenter upgrade tools to update all existing code in Studio to the new style automatically.

Implicit coercions removed

In 9.0, all implicit coercions are eliminated, as well many explicit coercions using the `as` keyword. Different coercions have different types of replacements. The application automatically upgrades these coercions in your Gosu code as specified in the following table. Unless otherwise specified in the rightmost column, both implicit and explicit coercions are disallowed.

Previous coercion from this type	Previous coercion to this type	Upgrader replaces implicit coercions with this new Gosu code
gw.pl.persistence.core.entity.KeyableBean	gw.pl.persistence.core.Key	`value?.ID`
entity.*ENTITYNAME*[]	entity.TQuery	`value.asQuery()`
String	typekey.*TYPELIST*	For `String` literals in which the programming context defines it clearly as a typecode, the replacement is the code literal on the typelist type. For example, consider a variable declared as the typelist type `typekey.ADDRESSTYPE`. To assign the typekey code `HOME`: • In 8.0 it was valid to assign the value `"HOME"`. At compile time, Gosu converted it to the full value

Previous coercion from this type	Previous coercion to this type	Upgrader replaces implicit coercions with this new Gosu code
		`AddressType.TC_HOME` if the programming context was the type `ADDRESSTYPE`. • In 9.0, the new form is `typekey.AddressType.TC_HOME` or the more concise form `TC_HOME`. For non-constant values, the upgrade tool converts the implicit coercion to `typekey.`*TYPELIST*`.get(value)`.
`String`	`EncryptedString`	`new EncryptedString(value)`
`String`	`org.apache.commons.lang.enums.Enum`	Direct reference to the constant on the enumeration class.
Numeric value	`gw.api.financials.CurrencyAmount`	`gw.api.upgrade.Coercions.makeCurrencyAmountFrom(value)`
`T[]`	`java.util.List<T>`	`value.toList()`
`T[]`	`java.util.Set<T>`	`value.toSet()`
`java.util.Collection<T>`	`T[]`	`value.toTypedArray()`
`String`	`char[]`	`toCharArray()`
`String`	`String[]`	`new String[] { value }`
`Object`	`java.util.Date`	`gw.api.upgradeCoercions.makeDateFrom(value)`
`Object`	`String`	Implicit coercion is replaced by an explicit cast: `value as String` This is one of the few implicit coercions that can be replaced by an explicit coercion using the `as` keyword.
`Object`	`java.lang.StringBuffer`	`new StringBuffer(value as String)`
`Object`	`java.lang.StringBuilder`	`new StringBuilder(value as String)`
Numeric values	Numeric values	Replaced by method calls on the class `gw.api.upgrade.Coercions`. The method names have the form make*TYPE*From. In some cases there are multiple methods for types that have `Object` and primitive versions. For example: • For `java.lang.Integer`, the method is `makeIntFrom`. • For the Gosu primitive `int`, the method name is `makePIntFrom`.

Previous coercion from this type	Previous coercion to this type	Upgrader replaces implicit coercions with this new Gosu code
		This is one of the few implicit coercions that can be replaced by an explicit coercion using the as keyword. However, the behavior of as is different in rare edge cases. Thus, the upgrade tool uses a safer but more verbose upgrade transformation.

PolicyCenter business data coercions

Previous coercion from this type	Previous coercion to this type	Upgrader replaces implicit coercions with this new Gosu code
String	T where T is option coverage term type	`gw.api.upgrade.PCCoercions.makeOptionCovTerm<T>(value)`
	T where T is package coverage term type	`gw.api.upgrade.PCCoercions.makePackageOptionCovTerm<T>(value)`
	T where T is product model type	`gw.api.upgrade.PCCoercions.makeProductModel<T>(value)`

XML API changes

PolicyCenter changed the Gosu XML API. The changes you must make in your code to accommodate the changes depends on which XML classes you used and whether you used schemas (XSDs) to create Gosu types.

XSD type code generation

Support for XSD types are now implemented using code generation. After adding or changing any XSDs, you must regenerate the code generation with menu items in Studio, such as **Generate Xml Classes**.

IMPORTANT This code generation step is no longer required as of version 9.0.2.

Important changes to legacy XML API based on XMLNode

In 9.0, there are important changes to the legacy XML API based on the `XMLNode` class:

Legacy XML API no longer loads XSDs

The legacy XML API (which is based on the `XMLNode` class) no longer supports loading an XSD and creating Gosu types from it. This change only affects XSDs that were registered to use the legacy XML API in PolicyCenter 8.0, which required an entry in the configuration file:

```
config/registry/compatibility-xsd.xml
```

If you added any lines to that configuration file, you must be aware of the change. Immediately convert your code to use the standard XML API based on `XmlElement`. See the *Gosu Reference Guide*.

Because generated XSD types have a different structure with the current XML API, you may get compiler errors to fix the syntax of related code. For example, working with an arbitrary type called `MyXMLType`, the code `MyXMLType.Choice.ComplexType` must be change to `MyXMLType.ComplexType`.

Legacy XML API using untyped XMLNode instances is deprecated

The use of the `XMLNode` class as untyped XML nodes still works but is deprecated. Do not write new code that uses `XMLNode`. Begin to convert any `XMLNode` untyped node code to use the newer XML API based on the `XmlElement` class.

If you use the deprecated XML API, you might need to make manual changes due to minor package changes. In previous releases, the `XMLNode` class and the related interface `IXMLNode` both existed in two packages. In version 9.0, only the `gw.xml` package is valid for these types. Depending on which package you used, you might need to modify your code:

- If you used the class `gw.api.xml.XMLNode`, change all usages to `gw.xml.XMLNode`.
- If you used the interface `gw.api.xml.IXMLNode`, change all usages to `gw.xml.IXMLNode`.

Changes to XmlElement if you do not use XSDs

In 9.0, if you directly instantiate the `XMLElement` class rather than use schemas (XSDs) to create Gosu types, there are required changes.

Your code must change usages of special property names that now appear with a dollar sign ($) prefix. To fix the compile errors, add the dollar sign prefix for the following `XmlElement` properties: `$AttributeNames`, `$Children`, `$DeclaredNamespaces`, `$MixedContent`, `$Namespace`, `$NamespaceContext`, `$QName`, `$SimpleValue`, `$Text`, `$TypeData`, and `$TypeInstance`.

For example, if your code with a compile error says `obj.Text`, change it to `obj.$Text`.

Changes to XmlElement using XSDs

In 9.0, any reference to XML element attributes or properties called `Class` must change to `_Class`. If you get compile errors related to this change, add the underscore prefix.

Customizing XSD type code generation to exclude types

In 9.0, it is possible to customize the way PolicyCenter converts XSDs to internal Java code, which is a process called *code generation*. By default, code generation happens for every XSD type. However, for better Studio performance, you can optionally omit code generation for some XSDs in some contexts or all contexts. The new `gwxmlmodule.xml` file controls this configuration.

See also

- *Gosu Reference Guide*

Special handling of XSD enumerations with more than 2000 entries

If you use XSDs very large enumerations, there are changes that might affect your XML code that uses those XSDs. In PolicyCenter 9.0, if an XSD defines an enumeration (an `<xs:enumeration>` element) with more than 2000 entries, PolicyCenter converts the enumeration to a `String` value (an `<xs:string>` element).

This affects your Gosu code that uses the enumeration, as well as the overall type safety of related code. In addition to changing the type of any relevant properties, the Gosu compiler cannot perform compile-time verification of the individual enumeration values.

During upgrade, you might get compilation errors due to this change. Change the XML code that uses those properties by getting and setting `String` values instead of enumerations.

See also

- *Gosu Reference Guide*

XSD types always have lowercase package names

In 9.0, the generation of XSD types always has lowercase packages names in Gosu.

For example, suppose an XSD file is at the path `gsrc/example/MyTest.xsd` and contains an element called `Address`. In 8.0, the type was `example.MyTest.Address`. In 9.0, the type is `example.mytest.Address`. If you get Gosu compilation errors due to this change, fix the capitalization of the package names.

XSD substitution group API changes

In 9.0, the Gosu API changed slightly for the XSD feature called substitution groups. You may get compilation errors in some cases due to type incompatibilities. To fix the issue, call the `cast` method (with no arguments) on the type you want to assign. For example:

```
xml.Address = new schema.UKAddress().cast()
```

See also

• *Gosu Reference Guide*

GX model changes

In 9.0, there are several changes in the Guidewire XML (GX) Modeler tool in Studio:

• For any GX model files, the product automatically upgrades all GX model source files to a new format in version 9.0. To identify the types within the model file itself, the new format has `xmltype` attributes on all property mappings. The product automatically determines the correct type. This change does not affect the API or usage of the tool in Studio. This changed is mentioned to help understand file differences that you notice during product upgrade from an earlier release.

• If any models for a Gosu class contained a mapping for the `IType` property, that mapping is gone with no visible error. The `IType` property is no longer directly mappable. Alternatively, add and map a new Gosu enhancement property with similar information.

Gosu properties file support removed

PolicyCenter 9.0.0 no longer provides native Gosu support for properties files (`.properties` files).

See also

• *Gosu Reference Guide*
• "Changed in Gosu in 9.0.0" in the *Configuration Upgrade Guide*

Any uses lines must be at the top of the file

In 9.0, any use of the `uses` keyword to include types must appear at the top of the file below the package declaration, but not within method declarations. In previous releases, `uses` lines were sometimes permitted in method declarations. The application automatically upgrades some configuration data that is implemented as Gosu classes, such as rule set data.

For other code contexts, fix any compilation errors that relate to this change.

The typeis operator compile-time changes

In version 9.0, the Gosu operator `typeis` is more restrictive in its use. The left side of the `typeis` operator must be an object of a compile-time type that is possible at run time to be the type specified after the operator. This is a new requirement that is enforced at compile time by the Gosu editor. For example, `typeis` works if the type on the right side is a subtype or supertype of the compile-time type of the left-hand side of the operator.

For example, the following code was valid in version 8.0 but is a compile error in version 9.0:

```
var b : Boolean = true
var x = (b typeis java.lang.Integer)  // check if the type is a java.lang.Integer
```

The Gosu compiler knows that a Boolean object can never be a `java.lang.Integer`, so it is an error. This change increases the chance of detecting a certain class of accidental errors.

Generally speaking, any effects of this change result in compile errors but no run time errors. Fix any compilation errors that relate to this change.

If you use any dynamically-compiled code, test it or carefully review it for its use of the `typeis` operator. For example, the application dynamically compiles Gosu code embedded in templates used for Gosu document production.

Changes to annotations

Annotations are a Gosu (and Java) syntax of adding metadata to a class declaration, or to part of a class, such as a method declaration. The "at" sign indicates usage of the annotation, for example `@WebService`. Some annotations take arguments.

In version 8.0, custom Gosu annotations were defined as classes that extend the interface `IAnnotation`. In version 9.0, the `IAnnotation` interface is deprecated, although any built-in annotations that use that annotation are not deprecated due to this change. Instead there is a new syntax for native annotations defined in Gosu using the `annotation` keyword. If you created any custom annotations with `IAnnotation`, begin to convert your annotations to the new syntax.

For the most part, using existing annotations from Gosu is unchanged. However, in version 8.0, native Java annotations required named arguments when used from Gosu, whereas Gosu `IAnnotation` annotations did not require named arguments. In version 9.0, argument names are now always optional for all annotations.

See also

- *Gosu Reference Guide*

chapter 20

New and changed in integration in 9.0.0

New in integration in 9.0.0

Payment gateway

There is a new PolicyCenter 9.0 feature for optional configuration of a payment gateway.

To implement a payment gateway, write your own implementation of the `PaymentGatewayConfigPlugin` and `PaymentGatewayPlugin` plugin interfaces. In the default configuration, there is exactly one implementation of each interface. The implementation of the `PaymentGatewayConfigPlugin` plugin interface is called `DefaultPaymentGatewayConfigPlugin`. The implementation of the `PaymentGatewayPlugin` plugin interface is called `StandAlonePaymentGatewayPlugin`. These examples are provided for demonstration only, and are not intended for production use.

See also

- *Integration Guide*

SystemToolsAPI Web service has new cluster management methods

In 9.0, the `SystemToolsAPI` web service has three new methods related to clusters:

- `getClusterState` – Returns a list of all nodes in the cluster, their roles, and what distributed components they run.
- `nodeFailed` – Cleans and releases resources (batch processes, plugins, and message destinations) reserved by a specified node.
- `completeFailedFailover` – Completes component failover.

See also

- *Integration Guide*

GX model constructor deprecated

The best way to create an instance of a GX model is to call the `create` static method. It is not a best practice to call the GX model constructor that uses the `new` keyword and that takes an `object` parameter. Hence, Guidewire is deprecating the constructor. The GX model `create()` method uses the following syntax:

```
var xmlInst = YOURPACKAGE.MODELNAME.TYPENAME.create(object)
```

An example of a call to the `create()` method is as follows:

```
var modelSysB = com.mycompany.addressmodel.Address.create(myAddress, {labelSysB})
```

The deprecated constructor call uses the following syntax:

```
var xmlInst = new YOURPACKAGE.MODELNAME.TYPENAME(object)
```

An example of a call to this constructor is as follows:

```
var xmlAddressData01 = new com.mycompany.addressmodel.Address(myAddress)
```

Guidewire recommends that customers use the `create` method instead of the constructor taking an `object` parameter. This constructor will still exist for creating a GX model object, but Guidewire discourages its use. Consistent with deprecating the constructor, Guidewire is removing instructions and examples that call it.

Changed in integration in 9.0.0

IPolicyTermPlugin Plugin interface new method

In 9.0, there is a new method named `calculatePeriodEndFromBasedOn` on the `IPolicyTermPlugin` plugin interface. The method calculates the period end from a based-on `PolicyPeriod`, with optional date reconciliation. Date reconciliation is useful for edge cases, like for two consecutive half year terms, ensuring they collectively last exactly one year.

See also

- *Integration Guide*

ProductModelTypelistGenerator Method trimTypeCode Moved and renamed

In 9.0, the method to trim a product model typecode in PolicyCenter to the 50-character length that a ClaimCenter integration requires has changed to `CCProductModelCodeMappingUtil.getCCTypeCodeForPMType`. The parameter that the new method takes is also different from the parameter that the previous method used. The new method takes an object that provides a typecode instead of the typecode itself. For example, the new method takes a `PolicyPeriod.Policy.Product` object instead of a `PolicyPeriod.Policy.ProductCode` property or a `Coverage.Pattern` object instead of a `Coverage.PatternCode` property.

Territory code plugin removed

In 9.0, the territory code plugin was previously deprecated and is now removed. If you need to customize the logic of lookup of territory codes, in Studio edit the Gosu class `gw.lob.common.TerritoryLookupCriteria`. That class has a `search` method that takes the object instance fields and generates a database query based on the fields. The database query returns a set of `Territory` objects that satisfy the criteria.

Upgrade free-text search from 7.0 to 9.0

If you used free-text search with PolicyCenter 7.0, you must upgrade the Guidewire Solr Extension in addition to upgrading your installation of PolicyCenter. The Guidewire Solr Extension that PolicyCenter 8.0 includes is based on Apache Solr 4. The version included with early releases of PolicyCenter 7.0 is based on Apache Solr 3.

For complete instructions on how to upgrade from Apache Solr 3 to Apache Solr 4, see "Upgrade free-text search from 7.0 to 8.0" on page 214

See also

Timing for free-text search upgrade

Guidewire recommends that customers plan ahead for additional fields to the Solr Extension index. Add fields to the index before or during a PolicyCenter upgrade. Do not wait until after an upgrade to do this.

The reason for such advance planning is that the addition of fields to the Solr Extension index requires executing the Solr batch load process. The time and resource cost for running batch load is substantial when a PolicyCenter instance has many policies. Add a field to the Solr Extension index as soon as you determine that the field would be useful and feasible for free-text search.

Guidewire recommends that customers time order the steps for an upgrade conceptually as follows:

1. With a long-term perspective, discern which entity fields would be useful and feasible as search fields for the basic search page. Apply this step with respect to any entity fields other than the default free-text search fields.
2. In the new target system, configure free-text search for these additional entity fields. Team members do not need to configure the basic search page for these fields at this moment. They can configure the basic search page after the upgrade.
3. Perform the upgrade from the current PolicyCenter system to the new system.
4. Run the free-text batch load command.

Configuring free-text search fields before upgrading avoids executing the free-text batch load process a second time after the new target database is full of policies. The preceding guidance applies without regard to whether the old source system uses free-text search.

RPCE web services fully removed

In version 8.0, PolicyCenter deprecated the older style of web services called RPCE.

In version 9.0, PolicyCenter removes RPCE web services from the product. Replace any RPCE web service code with WS-I web services.

As part of this change, the following data model attributes are no longer valid, because they were specific to RPCE web services: `exportable`, `exportasid`, and `SoapNullOk`. Because WS-I web services are not permitted to have arguments or return values that contain entity instances, these attributes have no meaning. Instead, design your web services to use data transfer objects (DTOs), either custom Gosu classes or XML/XSD types.

WS-I web service changes to regen-soap-api and @WsiGenInToolkit

In 8.0, most of the behavior of the `gwpc regen-soap-api` tool related to RPCE web services, which are no longer in the product. However, the tool also generated WSDL for locally published web services for web services with the `@WsiGenInToolkit` annotation. This tool added WSDL in the `PolicyCenter/soap-api` folder, not into the source code hierarchy.

In 9.0, the `gwpc regen-soap-api` tool is removed, including the WS-I behavior for local WSDL. In effect, there is no meaning to the `@WsiGenInToolkit` annotation in 9.0.

If you need the WSDL to test your web services, that feature is still available and generates the WSDL into the source code hierarchy in the `wsi.local` package. See the *Integration Guide*.

If you need the WSDL for external systems, you can still get the WSDL from a running server. See the *Integration Guide*.

Cluster methods in MaintenanceToolsAPI moved to SystemToolsAPI

In 9.0, Guidewire moved the following methods relating to clustering from web service `MaintenanceToolsAPI` to web service `SystemToolsAPI`:

• `getClusterState`
• `nodeFailed`
• `completeFailedFailover`

Document management changes

Document Assistant removed

In previous releases, the Guidewire Document Assistant provided interactive features to create, view, and edit documents. In version 9.0, there is no Document Assistant.

PolicyCenter no longer provides interactive viewing and editing of documents itself. Instead, PolicyCenter uses your web browser's capabilities for displaying documents or downloading them to the local computer system.

See also

- *Application Guide*
- *Integration Guide*

Client-side document production removed

In version 8.0, PolicyCenter provided client-side production of Microsoft Word and Microsoft Excel documents on Windows.

In version 9.0, client-side document production is removed entirely. However, there is now server-side production of Word and Excel documents. Guidewire strongly recommends using Word format instead of RTF for document production.

See also

- *Application Guide*

Document upload multiple file upload and editing

In 9.0, it is possible to upload multiple files at once. In the Actions menu, look for an action called **Upload Documents** or **Attach Documents**. You can drag multiple files from the local file system directly into the browser window.

To edit multiple files, navigate to the list of documents on an object. Click the checkbox at the top of the column to select all documents. Next, click the **Edit Details** button and follow the instructions on screen.

To control the maximum number of files to upload, modify the `config.xml` parameter `MaximumFileUploadCount`. To control the maximum size of a particular file, modify the `MaximumFileUploadSize` parameter.

See also

- *Application Guide*

Document upload drag and drop

In 9.0, if there is a **Browse** button to find a local document to upload, it is possible to drag files directly into the field next to the **Browse** button.

The user interface for multiple-file upload also supports drag and drop.

See also

- *Application Guide*

Document template descriptor context objects can refer to earlier context object symbols

In version 9.0, with document template descriptor definitions, Gosu expressions in context object definitions can refer to symbols from previously-defined context objects.

For example, suppose the XML file that defines a document template descriptor creates context objects in the following order: `Recipient1`, `Recipient2`, `Recipient3`. The definition for `Recipient2` can contain a Gosu expression that references the symbol `Recipient1`. Similarly, the definition for `Recipient3` can contain a Gosu expression that references the symbol `Recipient1` or `Recipient2`.

See also

- *Integration Guide*

Document template descriptor supports date type

In version 9.0, with document template descriptor definitions, the context object type attribute supports an additional type called `date`. Specify that type for date values.

See also

- *Integration Guide*

New requirement for implementations of IDocumentMetadataSource plugin

If you write your own implementation of the `IDocumentMetadataSource` plugin interface, there is a required change during upgrade to version 9.0.

After retrieving a document, call the new `setRetrievedFromIDMS` method on the `Document` entity instance. The method sets `doc.PersistenceRequired` to `false` and marks the document as coming from the `IDocumentMetadataSource` plugin.

New event for failure to save asynchronous document

In 9.0, if a document fails to store in the database, PolicyCenter triggers a `FailedDocumentStore` event on the relevant `Document` entity instance. The event is important because it is the only notification that document storage failed. Use this event to create a notification for administrators or users.

See also

- *Integration Guide*

PDF production plugin parameter removed

In 9.0, to configure PDF production, you must edit plugin parameters in the plugins registry `IPDFMergeHandler.gwp`. Although you still must edit that file to add the license number for the PDF production, the plugin parameter `BatchServerOnly` is no longer used. PDF document production now happens always on all servers.

See also

- *Integration Guide*

Public ID length is 64 characters

In version 8.0, the maximum length of public IDs was 20 characters in ClaimCenter, BillingCenter, and ContactManager. PolicyCenter supported a maximum length of 64 characters.

In version 9.0, all Guidewire InsuranceSuite applications support 64 character public IDs.

Changes to Guidewire Java API in 9.0.0

PolicyCenter version 8.0 introduced the Guidewire Java API version 8.0. The Guidewire Java API version 7.0 was deprecated.

PolicyCenter version 9.0 introduces the Guidewire Java API version 9.0. The upgrade path for your code depends on your previous version:

- The Guidewire Java API version 7.0 remains deprecated but available in PolicyCenter 9.0. Guidewire strongly recommends that you upgrade your Java code to use the Guidewire Java API version 9.0 to avoid future upgrade issues. Additionally, Gosu code that accessed Java code that used the Java API version 7.0 was upgraded

automatically for different default behavior. See "Changes to Gosu code that accesses Java API version 7.0 types" on page 127.

• The Guidewire Java API version 8.0 is removed and unavailable in PolicyCenter 9.0. You must upgrade your Java code from PolicyCenter 8.0 to use the Guidewire Java API version 9.0, or your code will not compile. See "Java API changes for entity and typelist data" on page 126.

Additional changes to the Guidewire Java API and Java configuration as described in the following topics.

Java API library generation changes

PolicyCenter changed the command line scripts that regenerate the Java API libraries and documentation.

To generate the libraries for the Java API version 9.0, at a command line prompt, change your working the application directory. Next, type the following command:

```
gwb genJavaApi
```

To generate the libraries for the deprecated Java API version 7.0, instead use the following command:

```
gwb genJavaApi -Ddeprecated=true
```

Java API reference documentation changes

In 9.0, the Java API Javadoc includes the specification of the plugin definitions for Java plugin interfaces, entity types, typelist types, and other types available from Java.

View the Java API reference at a new location:

```
PolicyCenter/javadoc/
```

The Javadoc contents are now static. The script that regenerates the Java API (`gwb genJavaApi`) does not regenerate the Javadoc. Therefore, your own data model changes are not reflected in the Javadoc documentation. However, your changes to entity types, typecodes, and new properties are available from Java code in your Java IDE.

Java API changes for entity and typelist data

The important differences between versions 8.0 and 9.0 of the Java API are primarily about accessing entity data and typelist data. The following tables compare the Java API in both versions.

Differences in entity types

Java API feature	Usage in 8.0	Usage in 9.0
Java access entity types through what kind of Java type	Java interfaces	Java classes
Number of generated types for each entity type	1, 2, or 3, depending on the entity type and whether you extended the entity type	1
Same type name for Java entity types compared to Gosu	No, though it depended on which entity interface. Some interfaces had suffixes such as `CoreExt` or `Ext`.	Yes
Same package names for Java entity types compared to Gosu	No	Yes
Entity instantiation uses the standard new operator as in Gosu or typical Java code	No. In version 8.0, you instantiated an entity using code such as `Address.TYPE.newInstance(b)`	Yes.

Java API feature	Usage in 8.0	Usage in 9.0
		However, entity constructors from Java require a bundle argument, even if there is a current bundle. See the *Integration Guide*. In contrast, the bundle argument is optional in Gosu in contexts where there is a current bundle.
Access to Gosu enhancements from Java requires reflection APIs	Yes	Yes

Differences in typelist types

Java API feature	Usage in 8.0	Usage in 9.0
Typekeys	Yes	The typecode constants defined on the typelist classes now represent the actual instances (typekeys) of the typelist classes.
Getting an individual typecode from the typelist class.	To get a comparable instance of the typecode, you needed to use the typelist type get method.	Directly access instances from the typelist classes as static fields with the same names as before with the TC_ prefix. The typecode get method is no longer required, and is now deprecated.
You can use the == operator to compare two typekeys for equality.	No	Yes
Same package names compared to Gosu	No	Yes

See also

- *Integration Guide*

Changes to Gosu code that accesses Java API version 7.0 types

The Guidewire Java API version 7.0 remains deprecated but available. While your legacy code works in this release, Guidewire strongly recommends that you upgrade your Java code to use the version 9.0 API to avoid future upgrade issues.

Additionally, any Gosu code that accessed Java code that used the Java API version 7.0 was upgraded automatically to accommodate different default behavior. The result of the automatic upgrade is the same run time behavior as in 7.0 and 8.0.

In previous releases, PolicyCenter had two implicit behaviors when calling Java code that used the Java API version 7.0:

- For function arguments or return values, Gosu implicitly converted entity types to the external entities through a proxy.
- For function arguments or return values, Gosu implicitly cloned any collections.

Both behaviors no longer occur, so any code that used the Java API version 7.0 requires explicit conversion of entity types and explicit collection cloning. For existing code that you have not yet converted to Java API version 9.0, an automatic upgrade step converts any affected function calls. Your code is replaced by code that does those actions explicitly rather than implicitly. Although the code may be harder to read after conversion, the run time behavior is the same as in version 7.0 and 8.0. If you need to modify your code, Guidewire recommends upgrading your code to use the Java API version 9.0.

The configuration upgrade tools find all method calls to libraries and classes in the plugin directories that indicate the Guidewire Java API version 7.0. These method calls correspond to method calls on Java types that were defined in Java classes and libraries within plugin directories with paths that do not include `/basic/`.

The directories that trigger the code upgrade are:

- `PolicyCenter/modules/configuration/plugins/PLUGIN_DIR/lib`

- `PolicyCenter/modules/configuration/plugins/PLUGIN_DIR/classes`

The `PLUGIN_DIR` value represents a plugin directory as specified for that plugin implementation in the Studio Plugins editor. For use with general Gosu code not associated with a plugin, `PLUGIN_DIR` has the value `Gosu`. For code that applies to multiple plugin directories, `PLUGIN_DIR` has the value `shared`.

For any Gosu code that calls methods on those types, any Gosu in Studio is upgraded automatically.

After the upgrade to version 9.0 is complete, there is no behavior difference between the plugin directory subdirectory that has `/basic/` in the path. See "Changed location for Java plugin directories for non-osgi Java code" on page 128.

IMPORTANT After you upgrade any Java code to the Guidewire Java API 9.0, you can remove any of the added conversion code. However, remember that the behavior of collection is different in 9.0, because collections are not cloned as they were in the 7.0 version of the API. Before moving code to production, audit any related code that might be affected by this change in collection behaviors.

Changed location for Java plugin directories for non-osgi Java code

During upgrade to version 9.0, the location of Java files affects how Gosu code upgrades occur related to the deprecated Java API version 7.0. See "Changes to Gosu code that accesses Java API version 7.0 types" on page 127.

After upgrade to version 9.0 is complete, there is no behavior difference between plugin directory subdirectories with `/basic/` in the path compared to omitting that part of the path.

The old location works in this release but is now deprecated. Move your Java classes and libraries to the new standard location, which omits the `/basic/` in the path.

This change affects all classes and libraries in the directories:

- `PolicyCenter/modules/configuration/plugins/PLUGIN_DIR/basic/classes`

- `PolicyCenter/modules/configuration/plugins/PLUGIN_DIR/basic/lib`

The `PLUGIN_DIR` value represents a plugin directory as specified for that plugin implementation in the Studio Plugins editor. For use with general Gosu code not associated with a plugin, `PLUGIN_DIR` has the value `Gosu`. For code that applies to multiple plugin directories, `PLUGIN_DIR` has the value `shared`.

For example, move files in the following directory:

- `PolicyCenter/modules/configuration/plugins/PLUGIN_DIR/basic/lib`

Move these files to the directory

- `PolicyCenter/modules/configuration/plugins/PLUGIN_DIR/lib`

See also

- *Integration Guide*

Additional directories in plugins/Gosu are for internal use only

In 9.0, the `PolicyCenter/modules/configuration/plugins/Gosu` directory now has two subdirectories called `gclasses` and `idea-gclasses`. These new subdirectories are for internal use as compilation output paths. Never use, modify, or rely upon the contents of these directories.

New and changed in 8.0 maintenance releases

This part includes information about PolicyCenter 8.0 maintenance releases.

New and changed in PolicyCenter 8.0.6

There are no new features or major changes in PolicyCenter 8.0.6.

Changed in PolicyCenter 8.0.5

For resolved issues and known issues in 8.0.5, see the release notes.

Secure transport (HTTPS) for free-text search

In 8.0.5, it is possible to configure PolicyCenter and the Guidewire Solr Extension to use the Hypertext Transfer Protocol Secure (HTTPS) communication protocol to exchange data. By default, PolicyCenter and the Guidewire Solr Extension use the unsecured HTTP protocol to exchange data. HTTPS provides a more secure exchange of data than HTTP by authenticating the PolicyCenter and Guidewire Solr Extension websites and associated web servers before connecting to exchange data. In addition, HTTPS provides bidirectional encryption of exchanged data.

See also

• *Configuration Guide*

Upgrade trigger becomes work queue

In 8.0.5, the `Contact.AccountHolderCount` upgrade trigger has been replaced by Account Holder Count batch processing, which is implemented as a work queue.

On `Contact` objects, the `AccountHolderCount` value sometimes has an incorrect value. Account Holder Count batch processing finds `Contact` objects with incorrect `AccountHolderCount` values, and updates the value. Search performance can be affected when the value of `AccountHolderCount` is set incorrectly.

See also

• *System Administration Guide*

New purge temporary policy periods batch processing

Sometimes policy period (`PolicyPeriod`) objects in a temporary (`Temporary`) status exist in the database. Purge Temporary Policy Periods batch processing In 8.0.5 removes these temporary policy periods from the database.

See also

• *System Administration Guide*

WSS4J and xmlsec library changes

In 8.0.5, the copies of `WSS4J` and `xmlsec` Java libraries provided by Guidewire are no longer available for customer use. If you want to use these Java libraries, include your own copies of the libraries. By using your own copy, you can also use older or newer versions of the library than the built-in versions.

See also

- *Integration Guide*

New and changed in PolicyCenter 8.0.4

For resolved issues and known issues in 8.0.4, see the release notes.

New in PolicyCenter 8.0.4

Add new audit types

In 8.0.4, it is possible to add new audit types and create audit schedules using the new audit types. For audit types, you can now define which properties are associated with their audit schedules, and create audit schedules based upon that definition in Product Designer.

See also

- *Configuration Guide*

New in Rating Management

Guidewire Rating Management includes several new features.

Cascaded lookup in rate books

If a rate book has *cascaded lookup* enabled, the query looks for matching rate tables or rate routines in the primary rate book, then in alternate rate books.
See also the *Application Guide*.

Rate book storage self-contained

A rate book can own or reference rate tables. If the rate book is in stage status, you can now make the rate book self-contained. A self-contained rate book owns all of the rate tables it includes.
See also the *Application Guide*.

Multiple values in rate table columns

In a rate table definition, you can specify an exact match parameter as accepting multiple values. For example, a rate table definition has jurisdiction parameter that accepts multiple values. In a rate book that uses this definition, one row of the rate table content can supply a rate factor for the jurisdictions California, Oregon, and Washington.

See also the *Application Guide* and the *Application Guide*.

Merge rate books

If two rate books have the same policy line, you can merge the rate books. Rate book properties, rate routines, and rate tables from each rate book are merged into a new rate book.

See also the *Application Guide*.

Export rate books to spreadsheet

You can now export a rate book to a multi-tab spreadsheet in Excel format.

See also the *Application Guide*.

Improved UI navigation in Product Designer

In 8.0.4, it is possible to use the arrow keys to expand or collapse the nodes in the Product Model navigation tree. In addition, the corresponding Product Model page appears when you press the up and down arrow keys to scroll through the tree.

Product Designer also remembers the collapsed or expanded state for each subsection on a page. For example, if you expand the **Advanced** subsection on a policy lines page, then that subsection appears expanded on any other page on which it appears. These settings are stored per user, and are retained even if you log out of Product Designer.

Quote cloning for business intelligence

Some insurance companies have a need to capture quote information that users and other processes generate through the course of the day. Without quote cloning, these interim quotes are not available for capture during nightly processing for various business intelligence (BI) needs. In 8.0.4, quote cloning provides one approach for retaining some of this information long enough that it can be more easily captured for BI needs.

See also

- *Application Guide*

Changed in PolicyCenter 8.0.4

Changes in Rating Management

This topic describes changes to Guidewire Rating Management in 8.0.4.

Change parameter set in rate routine

When you create a new version of a rate routine, you can change the parameter set.

Rate table definitions must have argument source set

New in this release, all rate table definitions must have an argument source set. The argument sources can be left unspecified. The upgrade modifies existing rate table definitions and rate table lookups in rate routines that do not specify an argument source set.

The upgrade creates an `EmptyParameterSet` parameter set that applies to all policy lines. This parameter set contains no parameters.

For all rate table definitions that do not specify an argument source set, the upgrade:

- Sets `EmptyParameterSet` as the source of argument objects. The **Source of Argument Objects** appears on the **Argument Sources** tab. The name `EmptyParameterSet` is also the code of this parameter set.
- Adds a `<system default>` argument source set. The argument source for each parameter is set to null. The name `<system default>` is also the code of this argument source set.

For all rate table lookups in rate routines that do not specify an argument source set, the upgrade sets the argument source set to `

See also the *Upgrade Guide*.

Change in-use rate table definitions

You can now make certain types of change parameters and factors in rate table definitions used in rate books.

See also the following:

- *Application Guide*

Label changes in search screens

When searching for Rating Management components (rate table definitions, rate books, parameter sets, or rate routines):

- **Policy Line, Underwriting Company, Jurisdiction**, or **Offering** set to **<not specified>** matches Rating Management components regardless of policy line, underwriting company, jurisdiction, or offering.
- **Policy Line, Underwriting Company, Jurisdiction**, or **Offering** set to **<applies to all>** matches Rating Management components that apply to all policy lines, underwriting companies, jurisdictions, or offerings. In previous releases, this was **<none>**.

Export/import rate table to spreadsheet

Improvements to export rate table content to Excel spreadsheet including:

- Neutral color scheme.
- Spreadsheet contains the PolicyCenter version number in **Name Manager**. Spreadsheets exported from previous versions of PolicyCenter do not have this information.
- Column headings use freeze panes to maintain header position while scrolling through rate table content.

You can import spreadsheets created in previous version of PolicyCenter.

See also the *Application Guide*.

Changes to relaxation order in rate book matching

Rate book matching finds the most appropriate rate book to use for policy rating. Rate book matching relaxes optional rate book attributes to select the most appropriate rate book. In this release, the relaxation order of the optional attributes has changed.

In the previous release, rate book matching relaxed the optional attributes in the following order:

- Offering
- Jurisdiction
- Underwriting company
- Policy line

In this release, rate book matching relaxes the optional attributes in this order:

- Offering
- Underwriting company
- Jurisdiction
- Policy line

If you have rate tables from prior releases that rely upon the old relaxation order, through configuration, you can change the relaxation order to match the previous release.

See also the *Application Guide*.

Policy exception batch processes

For performance reasons, Guidewire now disables the Policy Exception batch processes in the base 8.0.4 configuration. These batch processes are:

- `OpenPolicyException`
- `BoundPolicyException`
- `ClosedPolicyException`

These batch processes call the Policy Exception rules to perform actions on the `PolicyPeriod` entity. In the base configuration, the rules are empty.

If you have not defined Policy Exception rules in a prior release, you can accept the upgrade changes to `scheduler-config.xml`.

If you need to upgrade policy exception rules from a prior release, you must merge your changes into the rules and enable and schedule the batch processes in `scheduler-config.xml`.

See also

- *Rules Guide*

Change to query against EffDated entity

Fixed an issue where a query against an `EffDated` entity incorrectly included instances in temporary branches. To include `EffDated` entities in temporary branches, call `withFindTemporaryBranches(true)` on the query object.

See also

- *Gosu Reference Guide*

New and changed in PolicyCenter 8.0.3

This topic describes the new and changed features in the PolicyCenter 8.0.3 maintenance release. For resolved issues and known issues in 8.0.3, see the release notes.

New in PolicyCenter 8.0.3

Product model loader

In 8.0.3, the Product Model Loader extracts product model data from the running PolicyCenter server, and stores the information in tables in the PolicyCenter database. Policy data combined with the Product Model Loader data can be extracted, transformed, and loaded (ETL) into a data warehouse or data store for analysis or reporting.

See also

- *Product Model Guide*

Account lockout in Product Designer

Product Designer 8.0.3 implements an account lockout policy. If you enter the incorrect user name or password three times in succession, your account is locked for three minutes. You must wait until the lockout period has elapsed before you attempt to log in again.

Security features in Product Designer

Product Designer 8.0.3 supports SSL in a single-user or a multiple-user configuration. See the *Product Designer Guide*.

This version also supports the features listed below.

- Mitigation against cross-site request forgery.
- Product Designer can no longer be embedded inside a frame by other web sites.

Changed in PolicyCenter 8.0.3

Changes in Rating Management

This topic describes changes made to Rating Management.

Specifying rate table lookup

In 8.0.3, you can choose whether to load a specific rate table into memory or access it from the database by specifying a lookup value in the user interface. By default, PolicyCenter loads rate tables into memory. Loading the rate table in memory can provide quick access. Although accessing the table on disk is slower, loading a large table into memory may slow system performance even more.

The `RateTableManagementMemoryRowThreshold` parameter in `config.xml` is now obsolete. Previously, this parameter determined whether to load the table into memory.

See also

- *Application Guide*

In/not in conditional operators

In 8.0.3, Guidewire added In and Not In conditional operators in rate routine steps. You can choose these operators if the left operand is a typekey value. The right operand choices are limited to a single typekey or a list of typekeys in a typelist. Local variables and functions meeting this criteria are permitted.

See also

- *Application Guide*

Changes to rate routines that do not calculate properties on the cost

In 8.0.3, parameter sets need to specify whether they will include Cost components or not. The database default for this Boolean field is `true`. When defining a parameter set in the user interface, you specify this value with the **Include Cost** field.

Adding rate routines that do not calculate properties on the cost no longer requires adding the associated parameter set to the `noCostDataParamSets` variable in *LOB*`RateRoutineConfig.gs`.

See also

- *Application Guide*

void functions in rate routines

In 8.0.3, in the instruction column of a rate routine step, it is now possible to have a `void` function that has no return value. In the sample data, the `logAmount` function provides an example.

See also

- *Application Guide*
- *Configuration Guide*

Can edit parameters sets in draft rate books

In 8.0.3, it is possible to edit parameter sets in draft rate books. You edit parameter sets on the **Parameters** tab of the **Parameter Sets** screen. The **Edit** button is disabled when a rate book with status other than **Draft** references a parameter set through an included rate table definition or rate routine.

See also

- *Application Guide*

Generic Rating Management components

In 8.0.3, it is possible to specify rate tables, rate routines, rate books, and parameter sets as generic and available to all policy lines. In previous releases, these Rating Management components could only be specific to a policy line. In the sample data, generic Rating Management components begin with `Generic` or `GX`. In the user interface, **Generic** appears in the name.

See also

- *Application Guide*

Parameter set wrappers

In 8.0.3, parameter set *wrappers* enable you to combine similar parameter sets and rate routines. You may have parameters sets that are essentially the same, except that the coverage changes by policy line, for example. Instead of a parameter set for each policy line, define one parameter set that uses a wrapper to select the coverage by policy line or other characteristic of the policy.

See also

- *Application Guide*

Value provider on rate table definition factors

In 8.0.3, it is possible to now specify a value provider on the **Factors** tab of a rate table definition.

See also

- *Application Guide*

Product model file restructuring

In 8.0.3, Guidewire has restructured the product model files in the `config/resources/productmodel` directory to facilitate adding and removing lines of business and state deliverables. The restructuring does not affect the Product Designer user interface. The restructuring included:

- New directory structure that separates files by policy line, product, and jurisdiction
- Lookup files organized into jurisdiction-specific and non-jurisdiction-specific files

Support for Oracle partitioned indexes

In 8.0.3, PolicyCenter now supports partitioning Oracle database indexes by hash partition, range partition, or local partition, which is partitioning based on the partitioning of the table for the index. You define index partitioning in the `database-config.xml` file by using the `<ora-index-partitioning>` XML element, which is a subelement of `<ora-index-ddl>`, which is itself a subelement of `<ora-table-ddl>`. This new subelement replaces the element `<ora-index-hash-partitioning>`, which is no longer valid.

See also

- *Installation Guide*

Free-text search

The Guidewire Solr Extension included with PolicyCenter 8.0.3 uses the Simple Logging Facade for Java (SLF4J) API. If you used free-text search with releases of PolicyCenter prior to 8.0.3, you must do the following:

• Modify the setup of the application server dedicated to the Guidewire Solr Extension

• Install the release of the Guidewire Solr Extension included with PolicyCenter 8.0.3.

No other configuration changes are required.

New and changed in PolicyCenter 8.0.2

This topic describes the new and changed features in the PolicyCenter 8.0.2 maintenance release. For resolved issues and known issues in 8.0.2, see the release notes.

New in PolicyCenter 8.0.2

Multicurrency integration between BillingCenter and PolicyCenter

In 8.0.2, BillingCenter and PolicyCenter now support InsuranceSuite integration for single currency and multicurrency configurations.

See also

• *Application Guide*

New plugin interface InboundIntegrationMessageReply

In 8.0.2, it is now possible to easily implement a message reply plugin that listens to a JMS queue using WebLogic or WebSphere. There is a new plugin interface called `InboundIntegrationMessageReply`, which has all the main features of the inbound integration plugin but provided as a message reply plugin not a startable plugin. For related upgrade information, see "InboundIntegrationPlugin Plugin interface changed name" on page 146,

Duplicating a product offering in Product Designer

In 8.0.2, it is now possible to duplicate a product offering in Product Designer. To duplicate an offering, click the offering row and then click **Duplicate**. Change the characteristics of the duplicated offering to make it unique.

New AllowsPremiumAudit property for policy period and policy line

In 8.0.2, the `PolicyPeriod` entity type provides a new `boolean` property `AllowsPremiumAudit`. You can use this property to provide different behavior for all combinations of whether a policy period allows both or neither of premium reporting and premium audit. For example, you can use this property to filter the payment plans to display for a new policy period. In the base configuration, the default value of this property is `false`. The Workers Compensation line of business sets this property to `true`.

Changed in PolicyCenter 8.0.2

Removed configuration parameters

The following configuration parameters have been removed from PolicyCenter 8.0.2:

- `BucketSizeForHistogramsOnAllIndexedColumns`
- `CollectHistogramsOnAllIndexedColumns`

See the *System Administration Guide*.

Changes to Rating Management

The following changes were made to Guidewire Rating Management in 8.0.2:

- "Interpolation with and without relaxing" on page 144
- "Match operators in rate table definitions" on page 144

Interpolation with and without relaxing

In 8.0.2, it is possible to define an interpolated parameter as either with or without relaxing. An interpolated parameter without relaxing requires that both bounds for the input parameter be found at the same level of relaxation. An interpolated parameter with relaxing allows the bounds to be found at different levels of relaxation. With relaxing, the algorithm finds the bound above or below the input parameter at some level of relaxation, then further relaxes the query to find the other bound.

Added **Interpolation - with Relax** and **Interpolation - no Relax** match operators on the **Parameters** tab of rate table definitions.

See also

- *Application Guide*

Match operators in rate table definitions

In 8.0.2, Guidewire added **Greater Than** and **Less Than** match operators on the **Parameters** tab of rate table definitions.

In rate tables, the **Less Than Or Equal Match** and **Greater Than Or Equal Match** match operators were implemented in a way that many found non-intuitive. With this release, the comparison is reversed.

PolicyCenter now uses a new match operator code, so as to preserve existing uses of these match operators. The old match operators have **(deprecated)** appended to the label. In addition, the previous documentation incorrectly described the behavior of these operators. The correct description of the deprecated operators is:

- **Less Than Or Equal Match (deprecated)** – The parameter value must be less than or equal to the policy value.
- **Greater Than Or Equal Match (deprecated)** – The parameter value must be greater than or equal to the policy value.

In the new implementation, the match operators are defined as:

- **Less Than Or Equal Match** – The policy value must be less than or equal to the parameter value.
- **Greater Than Or Equal Match** – The policy value must be greater than or equal to the parameter value.

See also

- *Application Guide*

Complex schedules

In 8.0.2, changes to the schedule interface enable you to create complex schedule data models with one-to-many relationships from a schedule item to many coverages. Previously, schedule data models were limited to a one-to-one relationship. As a result of this change, a new property called `ScheduledItemMultiPatterns` was added and the existing `ScheduledItemPattern` was deprecated. The deprecated property is fully backward-compatible, but will eventually be removed. Guidewire recommends that you convert your code to use the new property. If you use the new property, modify your code to handle the possibility that multiple patterns can be returned.

Changes to the product model

Rate is relative for rate modifiers

In 8.0.2, in Product Designer, there is a new **Rate is Relative** field in which you can specify a rate modifier relative to 0 or 1.

See also

- *Product Model Guide*

Added peril to the list of coverage term model types

In 8.0.2, it is now possible to specify **Peril** as a coverage term model type. In Product Designer, the **Peril** model type can be selected for coverage terms on conditions, coverages and exclusions.

See also

- *Product Model Guide*

Related changes to PolicyPeriod and PaymentPlanSummary entities

In 8.0.2, it is now possible to access denormalized payment-plan properties in `PolicyPeriod` through the new property `SelectedPaymentPlan`. This property is a foreign key to the payment plan summary for the policy period. The following table shows the relationship between these `PolicyPeriod` and `PaymentPlanSummary` properties and other changes to entity properties.

Policy period property	Payment plan summary property
`PaymentPlanName` — Removed. Use `SelectedPaymentPlan.Name`.	`Name` — Accessible by `PolicyPeriod.SelectedPaymentPlan.Name`.
`ReportingPatternCode` — Removed. Use `SelectedPaymentPlan.ReportingPatternCode`.	`ReportingPatternCode` — New property accessible by `PolicyPeriod.SelectedPaymentPlan.ReportingPatternCode`.
`PaymentPlanID` — Replaced by `SelectedPaymentPlan`, foreign key to a `PaymentPlanSummary` entity instance.	
`SelectedPaymentPlan` — New property replacement for `PaymentPlanID`, foreign key to a `PaymentPlanSummary` entity instance.	
	`PaymentPlanType` — New property, type key to `PaymentMethod`. Either `Installments` or `ReportingPlan`.
	`PaymentCode` — Unused, removed.

See also

- *Integration Guide*

New AllowsPremiumAudit property filters payment plans

In 8.0.2, the **Payment** screen displays the available payment plans for a policy period based on whether the policy period allows both or neither of premium reporting and premium audit. This screen now uses the new `AllowsPremiumAudit` property to determine which payment plans to display. In the base configuration, the **Payment** screen now displays all plans for a policy period that allows premium audit. For a policy period that does not allow premium audit reporting, the **Payment** screen displays plans that are not reporting plans. In the previous release, the **Payment** screen checked whether the line of business was Workers Compensation to determine whether to display reporting plans.

Changes to integration

InboundIntegrationPlugin Plugin interface changed name

In 8.0.2, two related changes happened for the inbound integration APIs:

- The `InboundIntegrationPlugin` plugin interface is now called `InboundIntegrationStartablePlugin`. Update any existing code, as well as items in the Plugins Registry in Studio.
- You can now easily write a message reply plugin that listens to a JMS queue using WebLogic or WebSphere. There is a new plugin interface called `InboundIntegrationMessageReply`, which has all the main features of the inbound integration plugin but provided as a message reply plugin not a startable plugin.

If you previously used the `InboundIntegrationPlugin` interface, you now have to determine whether to change each usage to one of two options. You may not get compile errors for usages of the `InboundIntegrationPlugin` interface because that type name is used internally. Do not use the type `InboundIntegrationPlugin` directly. Instead, search for all usages of `InboundIntegrationPlugin` and replace with one of the following choices.

- `InboundIntegrationMessageReply` – loads as `MessageReply` plugin, and has all the methods on the `MessageReply` interface.
- `InboundIntegrationStartablePlugin` – loads as a startable plugin, which is the same behavior as the `InboundIntegrationPlugin` in the previous release.

To use this new plugin interface, implement `InboundIntegrationMessageReply`. Remember to edit your messaging destination in Studio just like you would for a regular `MessageReply` plugin implementation.

IBillingSystemPlugin Plugin interface has two new methods

In 8.0.2, the `IBillingSystemPlugin` plugin interface has two new methods for syncing producers and producer codes: `syncOrganization` and `syncProducerCode`.

IBillingSystemPlugin Plugin retrieves payment plans as Gosu objects

In application releases prior to 8.0.2, the `IBillingSystemPlugin` plugin interface method `retrieveAllPaymentPlans` returned an array of `PaymentPlanSummary` entities. This method now returns the payment plans as an array of Gosu objects that implement the `PaymentPlanData` interface. To save the selected payment plan when the user binds the `PolicyPeriod`, set the `SelectedPaymentPlan` property of the `PolicyPeriod` by calling the `PaymentPlanData` method `createPaymentPlanSummary`.

The new property, `SelectedPaymentPlan`, in `PolicyPeriod` is a foreign key to the `PaymentPlanSummary` entity instance.

See also

- *Integration Guide*

PolicyPaymentPlugin Plugin retrieves payment plans as Gosu objects

In application releases prior to 8.0.2, the `PolicyPaymentPlugin` plugin interface method `filterReportingPlans` used a parameter that was an array of `PaymentPlanSummary` entities and returned an array of `PaymentPlanSummary` entities. This method now uses a parameter that is an array of Gosu objects that implement the `PaymentPlanData` interface. This method returns the reporting payment plans as an array of Gosu objects that implement the `PaymentPlanData` interface.

See also

- *Integration Guide*

New and changed in PolicyCenter 8.0.1

This topic describes the new and changed features in the PolicyCenter 8.0.1 maintenance release. For resolved issues and known issues in 8.0.1, see the release notes.

New in PolicyCenter 8.0.1

New in Rating Management

This topic describes what's new in Guidewire Rating Management in 8.0.1.

Extracting and purging rating worksheets

In 8.0.1, Guidewire Rating Management added two batch processes for extracting and purging rating worksheet data. The extract batch process copies the rating worksheet data to files in a location defined by the system. After worksheet data has been extracted, the purge batch process removes the rating worksheet containers from the database.

See also

- *Application Guide*

Rate table normalization

In a rate table that contains a range parameter and any other parameter, the table may have ranges that overlap. In 8.0.1, for tables that fall within specified configuration parameters, PolicyCenter normalizes the rate table to speed up rate factor lookups. For a specific rate table, you can choose whether or not it is normalized.

See also

- *Application Guide*

Editing long rate routines

In 8.0.1, if editing a long rate routine, you can edit the rate routine by section. A long rate routine has more steps than the value of the indexing threshold, the `RateRoutineIndexingThreshold` parameter in `config.xml`. In the base configuration, this parameter is set to 150.

See also

- *Application Guide*

High volume requests for quotes

Some insurers need to handle high volumes of quote requests generated by external applications, such as the web sites of *comparative raters*. A comparative rater's web site enables its customer to view and choose between quotes from multiple insurers. Insurers who receive these requests must generate large volumes of quotes. The comparative rater's customer expects to see information quickly, therefore PolicyCenter must quickly generate the quote. The insurer may have multiple PolicyCenter instances that handles these requests. These instances are referred as *quote-only instances* of PolicyCenter. To avoid database performance issues, the quote-only instances are not clustered.

Most of these high volume requests do not result in a policy. Therefore, it is desirable to save the quotes in an external database rather than the PolicyCenter database. In general, this external database is partitioned to concurrently handle requests from multiple instances of PolicyCenter.

For these types of insurers, Guidewire recommends a quoting architecture and provides an API and plugin interface for generating quotes in 8.0.1. To handle the high volume of quote requests, the architecture uses multiple instances of PolicyCenter. All instances leverage the same PolicyCenter version and configuration. One or more *quote-only instances* process quotes directly from comparative raters. The architecture uses a single *system-of-record* (SOR) to complete submissions and issue policies. To handle high-volumes, the SOR can be a cluster of servers.

See also

- *Configuration Guide*

Affinity groups

Some insurers write policies based on the insured belonging to a certain group. For these insurers, the lines of business in the base configuration enable you to associate an affinity group with a policy. In 8.0.1, it is possible for users who have the required **Affinity Group Administration** permission to define affinity groups in the **Administration** tab. On the **Policy Info** screen, you can select an affinity group from those that meet required criteria (product line, producer code, and date range) and associate it with a policy.

See also

- *Application Guide*

New support for OSGi plugins

In 8.0.1, you can now optionally implement plugin interfaces in Java using the OSGi standard. The OSGi framework is a Java module system and service platform that helps cleanly isolate code modules and any necessary Java libraries. Guidewire recommends OSGi for all new Java plugin development. To simplify OSGi configuration, PolicyCenter includes an application called IntelliJ IDEA with OSGi Editor separate from Guidewire Studio.

See also

- *Integration Guide*

Gosu using clause now supports finally clause

In 8.0.1, the Gosu `using` clause syntax now has an additional feature for adding additional cleanup code. You can optionally add a `finally` clause that runs after the statement body, even if exceptions occur in the body of the `using` clause.

See also

- *Gosu Reference Guide*

Changed in PolicyCenter 8.0.1

Changes to Rating Management

This topic describes changes to Guidewire Rating Management in 8.0.1.

Configuring rating worksheets

In this release, there are fewer steps for configuring rating worksheets for other lines of business.

See also

- *Configuration Guide*

Memory threshold parameter

The memory threshold parameter which determines how PolicyCenter performs a rating query is now the `RateTableManagementMemoryRowThreshold` parameter in `config.xml`. Formerly this parameter was `memory.threshold` in `rating.properties`.

Rating worksheet data

Rating worksheet data is now stored in PolicyCenter on the `WorksheetContainer` entity. Previously, rating worksheet data was stored in entities that implemented the `RatingWorksheet` delegate. The upgrade deletes entities that implement the `RatingWorksheet` delegate. Therefore, rating worksheet data from previous releases is not retained. If you wish to extract and retain rating worksheet information from prior releases, you must do this before upgrading.

See also

- *Application Guide*

Adding match operations

The steps for configuring a new match operation have changed. If you added a match operation in previous versions of PolicyCenter, carefully review the changes to the API and adjust your implementation appropriately. Pay special attention to the method that filters a list, because it now has ordering requirements.

See also

- *Configuration Guide*

Changes to rate routine plugin interface

PolicyCenter provides the Rate Routine plugin (`IRateRoutinePlugin`) interface to let you modify the processing of rate routines in PolicyCenter. This plugin now has two methods for filtering which properties on **Parameters** the user can select in an **Instruction** or **Operand** of a rate routine step. These methods are:

```
includeProperty
filterIrrelevantItems
```

See also

- *Configuration Guide*

Changes to web service transaction IDs

The 8.0.1 release changes one server behavior and one client behavior for WS-I web services:

- For WS-I web service configuration on the server, this release changed the behavior of the annotation `@WsiCheckDuplicateExternalTransaction`. If you add this annotation to your implementation class, the SOAP header `<transaction_id>` is now required in each request rather than optional.
- If you have Gosu code that is a WS-I client, there is a new utility method you can call to easily set the transaction ID for the next call. On the API object, call `api.initializeExternalTransactionIdForNextUse(transactionIDString)`. After the call completes, the application removes the transaction ID from that API object, which prevents accidental duplicate use.

See also

- *Integration Guide*

Change to inbound integration configuration

The 8.0.1 release changes how you configure inbound multi-threaded integrations such as the built-in file and JMS integrations. In previous releases, you added configuration parameters in the Plugins registry in Studio. In this release, you set a single parameter `integrationservice` and then do the rest of the configuration in the new file `inbound-integration-config.xml`.

Also, the API details for file and JMS integrations changed. There is a new plugin interface called `InboundIntegrationHandlerPlugin`. Register an implementation of this plugin to add your custom logic to your file and JMS integration.

See also

- *Integration Guide*

Gosu suppress warnings annotation

In 8.0.1, the Gosu language now provides limited support for the Java annotation `@SuppressWarnings`, which tells the compiler to suppress warnings.

See also

- *Gosu Reference Guide*

Gosu compound assignment operators for logical AND and logical OR

In 8.0.1, the Gosu language has two new compound assignment operators, which are operators that apply an operation to a variable then re-assign the variable to the result:

- The new operator `&&|` performs the logical AND operation to the previous value.
- The new operator `||=` performs the logical OR to the previous value.

Both operators work with the primitive type `boolean` or the object type `Boolean` on either side of the operator. For example, suppose you have two `boolean` variables called `needsUpdate` and `flagTest`. Consider the following code:

```
needsUpdate ||= flagTest
```

That expression has the meaning of

```
needsUpdate = (needsUpdate or flagTest)
```

Do not confuse these new operators with the other operators `&=` and `|=`, which apply bitwise AND and bitwise OR operations.

See also

- *Gosu Reference Guide*

New and changed in 8.0.0

This part includes information about PolicyCenter 8.0.0.

See also

New and changed in the application in 8.0.0

New in the application in 8.0.0

Account holder info screen

New to this version of PolicyCenter, the **Account Holder Info** screen provides a summary of a PolicyCenter account holder. This summary screen consolidates information retrieved from PolicyCenter, as well as information requested from BillingCenter, ClaimCenter, and ContactManager, or equivalent third-party systems of record. This screen provides a snapshot of the account holder's value to the organization.

The **Account Holder Info** screen is an example of a summary screen which consolidates information about an account holder from PolicyCenter and other applications. You can modify the **Account Holder Info** screen or add similar screens that display information for other account contact roles such as a **Secondary Account Holder** role.

See also

• *Application Guide*

Policy data spreadsheet import/export

In PolicyCenter, policy data spreadsheet import/export enables you to export policy data to and from a spreadsheet. You can review and revise the exported data in a spreadsheet editor. You can import data from a spreadsheet into PolicyCenter.

You can use policy data spreadsheet import/export to review or enter large amounts of data for commercial policies. You can review existing policy data in a spreadsheet, add or update the data, then import that data into PolicyCenter.

Policy data spreadsheet import/export uses the Office Open XML Workbook (`.xlsx`) spreadsheet format.

With policy data spreadsheet import/export you can:
- Export a template to a spreadsheet. The template provides just the column headings and typelists for fields needed in new submission jobs.
- Export policy data to a spreadsheet. The spreadsheet provides a snapshot of a current job that can be used for review purposes or to make modifications for most jobs, including submission, change, renewal, and rewrite jobs.
- Import updated or newly added policy data from a spreadsheet into PolicyCenter. Prior to committing the import, you can preview the changes that the import operation will make to the policy, and then accept or reject the entire import operation.
- Configure export formats that specify the fields to export within each supported coverable.
- Extend this functionality to handle spreadsheet import/export for additional coverables and other lines of business.

In the default configuration, policy data import/export is implemented for buildings and locations in the commercial property line of business.

See also
- *Application Guide*

Multicurrency policies

In PolicyCenter 8.0, PolicyCenter provides support for multicurrency policies. Through multicurrency, PolicyCenter provides the ability to write policies that provide insurance for assets in different currencies. For example, an insurer offers a commercial property policy that can include properties in more than one country. With multicurrency, the policy values the assets in the currency of the asset location.

In PolicyCenter 8.0, PolicyCenter is always configured as a multicurrency system, even if only one currency is defined. The data model and the business logic do not change when the display is set to single currency (the default). The multicurrency user interface elements are only visible in the base configuration when you enable multicurrency display mode. Even when multicurrency display is not enabled, the currency-related fields are still populated within the data model. In single currency display mode, PolicyCenter does not display the user interface to change those fields.

See also
- *Application Guide*
- *Configuration Guide*
- *Globalization Guide*

Service tier field on account

Service tiers enable the insurer to provide special handling or value-added services for certain customers, typically high-value customers. In PolicyCenter 8.0, the **Account File Summary** screen has a new **Service Tier** field. You can set this field to **Platinum**, **Gold**, or **Silver**, in decreasing order of service. You can configure these levels of service. In the base configuration, the service tier has no effect inside of PolicyCenter.

In the base configuration, you set the service tier in PolicyCenter. If you are integrated with ClaimCenter or BillingCenter, PolicyCenter then propagates the service tier on an account to:
- **BillingCenter** – The service tier on the BillingCenter account. Every time a submission is bound and issued in PolicyCenter, PolicyCenter sends the service tier on the account to the BillingCenter account.
- **ClaimCenter** – The service tier on the ClaimCenter policy. When an agent files a claim in ClaimCenter, ClaimCenter retrieves policy and account information from PolicyCenter. This information includes the service tier on the account.

The `Account` object has a new optional `ServiceTier` property. The typekey values for `ServiceTier` are configurable.

The integration between PolicyCenter and ClaimCenter 8.0 maps the `Account.ServiceTier` value to a ClaimCenter `Policy.CustomerServiceTier`.

The integration between PolicyCenter and BillingCenter 8.0 maps the `Account.ServiceTier` value to a BillingCenter `Account.ServiceTier`.

Data change menu link

In the **Administration** tab, the new **DataChange** menu link enables you to push important data changes to the production server. Guidewire recommends that you use this feature sparingly and only when mission-critical data needs to be updated on running production systems.

See also

- *System Administration Guide*

Changing the screen layout

You can now adjust several aspects of the screen layout, such as the size, ordering, and visibility of list view columns. These changes are stored as layout preferences, and persist across sessions until cleared.

See also

- *Application Guide*

Changed in the application in 8.0.0

Policy transactions

In PolicyCenter 8.0, the term *policy transaction* is used consistently in the user interface to refer to submissions, policy changes, and other policy transactions. In previous versions, these were sometimes referred to in the user interface as *work orders* or, less frequently, as *jobs*.

Policy transactions are implemented as jobs in the data model. Policy transactions are referred to as jobs in PCF files, Gosu classes, and other configuration files. Therefore, the configuration documentation refers to policy transactions as jobs.

Error messages in log files refer to policy transactions as jobs.

Changes to the team tab

In previous versions of the **Team** tab, the left sidebar showed individual group users, **In Queue**, and **Other**. In this version, **Other** is now labeled **Misassigned** and lists misassigned activities and jobs.

The **Team** screens show the job workload for groups or individual users. With this version, the **Team** screen determines whether a user has work to do on a job depending upon whether the user is a *by-role* or *by-activity* user.

With this version, renewal statistics are for the current week rather than the current month.

See also

- "Changes to team tab configuration" on page 168
- *Application Guide*

Administration menu items reorganized

In PolicyCenter 8.0, the **Administration** menu items have been reorganized to be more similar to other Guidewire Suite applications. The menu items are grouped into the following categories:

- **User & Security**
- **Rating**
- **Business Settings**
- **Monitoring**
- **Utilities**

Rate routine label changed

In Rating Management in PolicyCenter 8.0, when editing a rate routine, the **Policy Data** menu item was renamed **Parameters** because it lets you select parameters in the parameter set. This menu item appears in the **Instruction** and **Operand** columns.

Service tier field on account

In PolicyCenter 8.0, the **Account File Summary** screen has a new **Service Tier** field. The user can set this field to **Platinum**, **Gold**, or **Silver**. The `Account` object has a new optional `ServiceTier` property. The typekey values for `ServiceTier` are configurable.

The integration between PolicyCenter and ClaimCenter maps the `Account.ServiceTier` value to a ClaimCenter `Policy.CustomerServiceTier`.

The integration between PolicyCenter and ClaimCenter maps the `Account.ServiceTier` value to a BillingCenter `Account.ServiceTier`.

My submissions screen

In PolicyCenter 8.0, the **My Submission** screen displays the following fields:

Field	Description
Primary Insured	This field is unchanged.
Effective Date	The `PolicyPeriod.EditEffectiveDate` from the policy period used to determine the `Status`. By default, this field is used to sort the list.
Quote Needed	For submission policy transactions, this field displays `Submission.DateQuoteNeeded`. For issuance policy transactions, this field is blank.
Transaction #	This field was previously labeled **Job #**.
Type	This field is unchanged.
Status	This field is unchanged.
Issued	This field displays `Policy.Issued`.
Product	This field was previously labeled **Policy Type**.
Producer	This field is visible if user does not have the View Producer Desktop Details permission. This is a new permission.
Underwriter	This field displays the user with the Underwriter role in the `Job.RoleAssignments` array.

The **Create Date** column has been removed.

My other policy transactions screen

In PolicyCenter 8.0, the **My Other Work Orders** screen has been renamed **My Other Policy Transactions**. In addition, this screen now displays the following fields:

Field	Description
Type	This field is unchanged.
Primary Insured	This field is unchanged.
Effective Date	This field displays `PolicyPeriod.EditEffectiveDate` from the query used to determine the **Status**. By default, this field is used to sort the list.
Transaction #	This field was previously labeled **Work Order #**.
Status	This field is unchanged.
Product	This field was previously labeled **Policy Type**.
Producer	This field is visible if user does not have the View Producer Desktop Details permission. This is a new permission.
Underwriter	This field displays the user with the Underwriter role in the `Job.RoleAssignments` array.

The **Create Date** column has been removed.

Region and region search screens combined

In PolicyCenter 7.0, the **Administration** menu had **Region Search** and **Regions** menu items. In PolicyCenter 8.0, for usability, these two screens have been combined into the **Administration→Users & Security→Regions** screen. When you go to the **Regions** screen, the screen displays the first page of regions. In the base configuration, if there are more regions than fit onto one page, you can search to filter the list of regions.

The `SearchOn` property controls the visibility of search fields on the **Regions** screen. In the base configuration, the values for the `SearchOn` property are:

- `ALWAYS` – Always display search fields on the **Regions** screen.
- `NEVER` – Never display search fields on the **Regions** screen.
- `AUTO` – Display search fields if the number of regions is two or more pages. This is true if the number of regions is greater than `ListViewPageSizeDefault`.

The `SearchOn` property is defined in `RegionsUIHelper.gs` in the `gw.admin` package. In the base configuration, this property is set to `AUTO`.

The `ListViewPageSizeDefault` parameter in `config.xml` defines the number of rows to display in the list view.

Reinsurance attachment inclusions

A change was made to how PolicyCenter handles reinsurance program spanning more than one term. In PolicyCenter 7.0, if you exclude a risk from coverage by an agreement in a later term, PolicyCenter excludes the risk in prior terms. In PolicyCenter 8.0, PolicyCenter does not exclude the risk in prior terms. In addition, if you exclude a risk in a term, PolicyCenter automatically excludes it in later terms. PolicyCenter excludes the agreement automatically by creating an inclusion row for the later terms.

Changed in configuration in 8.0.0

For changes related to web services, see "Existing WS-I web services changes" on page 204.

Important changes to the configuration module

Guidewire has changed the structure and meaning of modules in PolicyCenter version 8.0.

In prior versions of PolicyCenter, all your modified files were in the directory `PolicyCenter/modules/ configuration`, also known as the *configuration module*. All the unmodified base configuration files were stored in a separate read-only directory. The first time you edited a base configuration file, Studio copied the file to the configuration module — a process often referred to as "copy on edit".

In other words, the configuration module previously contained:

- Your modified versions of base configuration files
- Your new configuration files

In version 8.0, the configuration module now contains all the following files in one merged hierarchy:

- Every base configuration file, whether modified or unmodified
- Your new configuration files

You can now edit all files directly in their location in the configuration module. If necessary, you can use Studio to revert any modified base configuration file to its original version. In Studio, right-click on the file name, then click **Revert to Base**.

Changes to the data model

The following attributes of data model entity definitions have been removed:

- `deletefk`
- `ondelete`

The following attribute of the `entity` data model element has been removed:

- `supertypeEntity`

Changes to the generate dictionary command

The `gwpc regen-dictionary` command has changed with a parameter to generate XML versions of the *PolicyCenter Data Dictionary* and the *PolicyCenter Security Dictionary*. The command also generates associated XSD files. Use the XML and XSD files to import the *Data Dictionary* and the *Security Dictionary* into third-party database design tools.

See also

- *Configuration Guide*

Changes to Guidewire Studio

This topic describes the changes that Guidewire has made to PolicyCenter Studio in version 8.0.0.

Studio now built on IntelliJ IDEA community edition

Guidewire Studio has been completely redeveloped as extensions to the popular IntelliJ IDEA development environment. While many PolicyCenter configuration editors will be familiar, navigating Studio will be a new experience. You can learn more about using IntelliJ IDEA at: `http://www.jetbrains.com/idea/documentation`

Gosu tester

The Gosu Tester is now called the Gosu Scratchpad.

Display keys editor

The Display Keys editor now provides a view of the display keys in a text format. To modify display keys, you now make your changes directly to the text of the `display.properties` file. For more information, see the *Configuration Guide*.

Studio features that are no longer available

The following Studio features are no longer available:
- Constraining the debugger to intercept calls only from a particular user.
- "Debug and break" to have the debugger automatically stop at the first line of Gosu code executed.
- Logging Studio diagnostic information.
- Localizing the Studio interface.
- Product model editors
- Script Parameters editor (you now edit the XML file directly).

Editing the product model

The product model editors have been removed from Guidewire Studio. You now use Product Designer to edit the PolicyCenter product model.

Product Designer is a web-based, multiuser tool included with PolicyCenter that provides easier access to product model editing for business analysts. To start Product Designer in single-user mode, go to the `productdesigner/bin` folder of your PolicyCenter installation directory and run `launchPD.bat`.

For complete information on installing, using, and administering Product Designer, see the . For field-level information while editing the product model, click the **Help** button while viewing any page in Product Designer.

Changes to configuration parameters

This topic describes the changes that Guidewire has made the PolicyCenter configuration parameters in version 80.0.

Removed configuration parameters

The following configuration parameters have been removed in PolicyCenter 8.0.0:

- `AllowActiveXAutoInstall`
- `CacheActive`
- `DisableHashJoinForProximitySearch`
- `DisableIndexFastFullScanForProximitySearch`
- `EnableExtRendering`
- `ExperimentalBrowserPerf`
- `LegacyExternalEntityArraySupport`
- `RIProgramCacheMaxDuration`
- `UseDocumentNameAsFileName`

Added configuration parameters

The following configuration parameters have been added in PolicyCenter 8.0.0:

- `ActivityStatisticsWindowSize`
- `AlwaysShowPhoneWidgetRegionCode`
- `ClusterMemberPurgeDaysOld`
- `ClusterMemberRecordUpdateIntervalSecs`
- `ClusterStatisticsMonitorIntervalMins`
- `ConsistencyCheckerThreads`
- `DefaultApplicationLanguage`
- `DefaultNANPACountryCode`
- `DefaultPhoneCountryCode`
- `DefaultXmlExportIEncryptionId`
- `DocumentAssistantJNLP`
- `JGroupsClusterChannel`
- `MaxAllowedDate`
- `MinAllowedDate`
- `PluginStartupTimeout`
- `WorkQueueThreadPoolMaxSize`
- `WorkQueueThreadPoolMinSize`
- `WorkQueueThreadsKeepAliveTime`

Changed configuration parameters

Guidewire has made the following specific changes to configuration parameters in PolicyCenter 8.0.0.

Parameter	Change
`AllowActiveX`	Renamed to `AllowDocumentAssistant`, and default changed to false.
`ClusterProtocolStack`	Default value changed.
`PolicyChangesStatisticsWindowSize`	Renamed to `OtherWorkOrdersStatisticsWindowSize`.
`UISkin`	Default value changed to Titanium.
`UseGuidewireActiveXControlToDisplayDocuments`	Renamed to `UseDocumentAssistantToDisplayDocuments`.

Changes related to PCF files

Template page PCF files removed

The Configuration Upgrade Tool deletes PCF template pages. These pages have a `<TemplatePage>` root element. The upgrade also removes `<EntryPoint>` elements that reference template pages.

Typical uses of `<TemplatePage>` were for reports generated from a template for PolicyCenter user interface users. The closest replacement for this context is a custom servlet. See the *Integration Guide*. Your servlet implementation class can use Gosu templates and render a Gosu template. See the *Integration Guide*. Also see the *Gosu Reference Guide*.

For more programmatic (rather than user interface) access to Gosu templates, replace template pages by a custom web service. Your web service implementation class can use Gosu templates and render a Gosu template. See the *Integration Guide*.

Miscellaneous PCF changes

The Configuration Upgrade Tool performs the following modifications to PCF files:

- Removes the `reflectOnBottom` attribute. This attribute was used to display the a virtual toolbar at the bottom of a page. The attribute was removed because the user interface needs to match the server configuration. No alternative configuration is available.

    Note: This change was reversed in a later release. The `reflectOnBottom` attribute was restored.

- Converts all `postOnChange` attributes on a value widget to a child `PostOnChange` node. For example, the upgrade converts:

```
<Input id="xxx" postOnChange="true" onChange="someMethod()" disablePostOnEnter="doEvaluation()"/>
```

to:

```
<Input id="xxx">
  <PostOnChange onChange="someMethod()" disablePostOnEnter="doEvaluation()"/>
</Input>
```

- Removes the `showNoneSelected` attribute from all `DetailView` inputs that are bound to a value. Setting `showNoneSelected=false` would suppress the **None Selected** option from drop-down lists and would default to the first option. This type of configuration was incorrect because the selection of the option was generally programmatically incorrect and was often used as a shortcut instead of specifying an explicit default. Verify all

removals to ensure there is not any dependent logic. If there is, specify an explicit default in the page configuration.

- Removes the `showNoneSelected` attribute from all `<ValueCellType>` nodes. See the above note about removal of the `showNoneSelected` attribute from all `DetailView` inputs that are bound to a value.
- Removes the `numDataEntriesPerRow` and `transposed` attributes from `RowIteratorNode` elements. Transposed lists are a relatively rare configuration. If you had one in your configuration, use a traditional list view.
- Removes `<DetailViewPanel>` elements from `<ButtonCell>`, `<ButtonInput>`, and `<ToolbarButtonType>` elements. Detail views can no longer be embedded inside buttons.
- Converts `valueWidth` attributes on cell widgets to `value` attributes. As of 8.0, PolicyCenter sizes cells by heuristics rather than content, so `valueWidth` is no longer necessary.
- If all cells in a row have the `useHeaderStyle="true"` property, the upgrade moves the property to the row level. A list can only have one header. See below.
- Updates rows to rename the `useHeaderStyle` property to `renderAsSmartHeader`. The property is renamed because the header functionality is more than styling. When a row is rendered as a smart header, all the row header interactive features are made available.
- Renames `<ContentCell>` elements to `<Link>`.
- Converts `<Cell>` elements within `<ColumnFooter>` to `<TextCell>` elements.
- Removes any element that is not a `<TextCell>` element from `<ColumnFooter>` elements.
- Removes `<ColumnHeader>` elements from `<CellType>` elements.
- Remove `<DetailViewPanel>` from `<ContentCell>`. The upgrade performs the following steps. After the automatic upgrade, review your `<ContentCell>` configurations to manually verify the configuration and make any changes. Content cells cannot have editable detail views embedded in them. Review all removals to ensure functionality. If editable content is needed within a row of data, the recommended configuration is a list detail panel.
  - For any `<ContentCell>` that contains a `<DetailViewPanel>`, the upgrade renames the `<ContentCell>` to `<FormatCell>`.
  - For other types of `<ContentCell>`, the upgrade renames the element to `<LinkCell>`.
  - Removes elements that are not allowed in the `<FormatCell>`, such as `<DetailViewPanel>` and `<InputColumn>`. This strips out unnecessary container elements. No content will be removed.
  - Renames inputs in the `<DetailViewPanel>` to `<TextInput>` unless they are `<ContentInput>`, `<TextInput>`, or `<NoteBodyInput>`.
  - Removes attributes that were allowed on specific input elements but not on `<TextInput>`.
- Removes the `useHeaderStyle` attribute from all cells that can be bound to a value. The header style in 8.0 is a lot more extensive. Smart header capabilities have been added, in addition to the styling. Header capabilities are at

the row level as opposed to the cell. If you are interested in highlighting content, there are a few other ways to achieve that. Review the PCF reference for a full list of attributes for that particular cell variant.

- Removes the `compress` attribute from `<DetailViewPanel>`.
- Removes the `compress` attribute from `<ListViewPanel>`.
- Removes the `compressIfSingleChild` attribute from `<InputGroup>`.
- Comments out `<ProgressCell>` elements. This was an uncommon widget that Guidewire has removed. If you were using it on some page and would like to continue to do so, create a list detail panel, and use the `ProgressInput` in the detail section instead.
- Removes the `refreshOnProgressComplete` attribute from `<ListViewPanel>` and `<Row>` elements. This is part of the removal of the `<ProgressCell>` widget.
- Removes the following attributes from `<ChartPanel>`:
  - `bgColor`
  - `border`
  - `displayPlotOutline`
  - `orientation`
  - `sameSeriesColor`
  - `threeD`
  - `tooltip`

  Guidewire cleaned up the `<ChartPanel>` schema as a part of simplification and a move to a more interactive experience.
- Removes the following attributes from `<DomainAxis>`:
  - `autoRange`
  - `autoRangeIncludesZero`
  - `tickUnit`
  - `upperMargin`
- Removes the `<Interval>` element.
- Removes the following attributes from `<RangeAxis>`:
  - `autoRange`
  - `autoRangeIncludesZero`
  - `tickUnit`
  - `upperMargin`
- Removes the `percentComplete` attribute from `<DataSeries>`.
- Removes the following from `<DualAxisDataSeries>`:
  - `autoRangeIncludesZero`
  - `lowerMargin`
  - `tickUnit`
  - `tooltip`
  - `upperMargin`
- Removes the following chart types from the `<ChartType>` enumerator:
  - `Waterfall`
  - `Gantt`
- Renames the following chart types in the `<ChartType>` enumerator:
  - `Dial` → `Gauge`
  - `Polar` → `Radar`
  - `Ring` → `Pie`
  - `StackedArea` → `Area` (There is no longer a distinction between a stacked and a non-stacked area.)
  - `XYStep` → `XYLine`
  - `XYStepArea` → `XYArea`

Changes to search criteria

In PolicyCenter 7.0, Guidewire represented search criteria using entities, Java, and Gosu classes. In PolicyCenter 8.0, Guidewire has converted most of the search criteria entities to Gosu classes that you can extend. These classes typically extend the `EntitySearchCriteria` or `SearchCriteria` class.

Guidewire has not converted the `PolicySearchCriteria` entity.

See also

• *Configuration Guide*

Gosu classes that extend EntitySearchCriteria

The `EntitySearchCriteria` abstract class provides a standard way of searching for entities. The Gosu classes that extend the `EntitySearchCriteria` class (in the `gw.search` package) are:

Gosu class	Package	Notes
AbstractClassCodeSearchCriteria	gw.lob	
BOPClassCodeSearchCriteria	gw.lob.bop	
CPClassCodeSearchCriteria	gw.lob.cp	
GLClassCodeSearchCriteria	gw.lob.gl	
WCClassCodeSearchCriteria	gw.lob.wc	
AccountSearchCriteria	gw.account	
AccountSearchCriteria700	gw.webservice.pc.pc700.account	This is for backwards compatibility.
ActivitySearchCriteria	gw.activity	
AgreementSearchCriteria	gw.reinsurance.search	
FacultativeSearchCriteria	gw.reinsurance.search	
FormPatternSearchCriteria	gw.forms	
HistorySearchCriteria	gw.history	
IndustryCodeSearchCriteria	gw.product	
MessageSearchCriteria	gw.system.messaging	
PolicyLocationBoundingBoxSearchCriteria	gw.policylocation	
PolicyLocationSearchCriteria	gw.policylocation	
ProducerCodeSearchCriteria	gw.product	
ProgramSearchCriteria	gw.reinsurance.search	
RateBookSearchCriteria	gw.rating.rtm.query	
RateRoutineSearchCriteria	gw.rating.rtm.query	
RateTableDefinitionSearchCriteria	gw.rating.rtm.query	
RILocationRiskProximitySearchCriteria	gw.reinsurance.search	
RiskClassSearchCriteria	gw.lob.common	
SharedContactAccountSearchCriteria	gw.account	
TaxLocationSearchCriteria	gw.lob.common	
TerritoryLookupCriteria	gw.lob.common	

Gosu classes that extend SearchCriteria

The `SearchCriteria` abstract class provides a standard way of searching for object others than entities. The Gosu classes that extend the `SearchCriteria` class (in the `gw.search` package) are:

Gosu class	Package	Notes
BillingAccountSearchCriteria	gw.plugin.billing	
ClausePatternSearchCriteria	gw.productmodel	This class replaces `CoveragePatternSearchCriteria`.
PCZoneSearchCriteria	gw.api.admin	
UWIssueTypeSearchCriteria	gw.job.uw	

Changes to configuring PolicyCenter

This topic describes changes related to configuring PolicyCenter features.

Changes to team tab configuration

In PolicyCenter 8.0, configuration information about the **Team** tab has been moved from the *Application Guide* to the *Configuration Guide*.

The `PolicyChangesStatisticsWindowSize` parameter in `config.xml` was renamed to `OtherWorkOrdersStatisticsWindowSize`.

The User batch process is now the Team Screens batch process. This batch process collects summary counts for team screens.

See also

- "Changes to the team tab" on page 157
- *Configuration Guide*

Changes to configuring underwriting authority

In PolicyCenter 8.0, the Policy Evaluation (New) rules that raised underwriting issues have been replaced by:

- Evaluator Gosu classes for each line of business (`PA_UnderwriterEvaluator.gs` for example)
- A default evaluator class (`DefaultUnderwriterEvaluator.gs`).

To write code that evaluates and raises your own underwriting issue types, modify these evaluator classes.

See also

- *Configuration Guide*

Changes to the PolicyCenter product model

This topic describes changes to the PolicyCenter Product Model in version 8.0.

All coverage terms must have value types

In PolicyCenter 8.0, **Value Type** is a required field for all coverage terms. You must specify a value type when configuring a coverage term in Product Designer. Validation within Product Designer prevents you from committing changes that include a coverage term that is missing a value type. However, it is possible to have missing value types in your configuration resulting from upgrades.

See also

- "Specifying ValueType on Coverage Terms" in the *Configuration Upgrade Guide*

Changes to validation in PolicyCenter 8.0

In PolicyCenter 8.0, product model validation and verification has changed due to the introduction of Product Designer. Within Product Designer, each change you make to the product model can be validated using a Validate link on the page where you made the change. Additionally, you can validated all changes in a change list prior to committing your changes to the PolicyCenter configuration. Even if you do not explicitly validate your changes, performing a Commit operation runs validation on all changes. The commit cannot proceed until all validation errors have been fixed. However, the validations run by Product Designer are only a subset of a complete product model verification.

To perform thorough product model verification, you must do one of the following:

- Start the PolicyCenter Server and, if it fails to start, check the system console log for product model error messages.
- Compile one or more product model files in Studio and check the **Log** window at the bottom of the Studio window for errors.

See also

- *Product Model Guide*

New documentation

In PolicyCenter 8.0, to better describe product model configuration to the intended audiences, the PolicyCenter documentation set includes the following new documentation:

- *Product Model Guide* – Explains all aspects of configuring a line of business, using both Studio and Product Designer. Much of the content of this guide was part of the *Configuration Guide* in previous PolicyCenter releases.
- *PolicyCenter Product Designer Guide* – Describes how to install, configure, administer, and use Product Designer to edit product model configuration files. All content in this guide is new for PolicyCenter 8.0.0.
- *Product Designer Help* – Field-level help for each Product Designer page. Click **Help** in Product Designer to display help for the current page.

Changes to configuring lines of business

This topic describes changes related to configuring PolicyCenter lines of business.

Lines of business modularization

In PolicyCenter 8.0, most of the line-of-business code has moved from rule sets, plugins, and non-line-of-business PCF files and Gosu classes to policy-line-methods classes in the `gw.lob` package. For example, workers' compensation code has been moved to the policy line methods class for workers' compensation, `WCPolicyMethods` in the `gw.lob.wc` package.

Guidewire recommends that you define line-of-business code in the policy line methods classes. Avoid putting line-of-business code in generic locations such as the rule sets, plugins, and non-line-of-business PCF files and Gosu classes. This recommendation is intended to make upgrade easier by grouping line-of-business changes in the `gw.lob` package.

Changes to cost wrapper

In PolicyCenter 7.0, the `CostWrapper` class contained line-of-business-specific code. In PolicyCenter 8.0, line-of-business-specific code in `CostWrapper` has moved to line-of-business classes. The following line-of-business classes which map costs to the correct `CostWrapper`:

- `GL_CostWrapperCollectionEnhancement`
- `WC_CostWrapperCollectionEnhancement`

These classes replaced the `CostWrapperCollectionEnhancement` class. The `CostWrapper` is now an abstract class.

Changes to industry code enhancement

In PolicyCenter 7.0, the `IndustryCodeEnhancement` class in the `gw.systables` package contained line-of-business-specific code. In PolicyCenter 8.0, this code has moved to the `BOP_IndustryCodeEnhancement` class in the `gw.lob.bop.systables` package.

See also

- "Line-of-business modularization changes to plugins and classes" on page 212
- *Configuration Guide*

Registering lines of business

In PolicyCenter 8.0, as a result of modularizing lines of business, every line of business must be registered. Registering a line of business means defining three properties in the `InstalledPolicyLine` typelist. The three required properties are:

- Package name
- Public ID
- Policy line subtype

Various parts of the product configuration use one or another of these identifiers.

> **Note:** The PolicyCenter server checks its policy line entities against the parameters in the `InstalledPolicyLine` typelist and refuses to start if it finds a policy line without a corresponding typecode.

See also

- *Product Model Guide*

Product type property

Certain user interface features apply only to commercial lines, while others apply only to personal lines of business.

In PolicyCenter 8.0, a new **Product Type** setting in Product Designer enables you to specify either **Personal** or **Commercial** for each product you define. This new property requires that you manually upgrade `Product` patterns from prior releases.

See also

- "Add Product Type to Product Patterns" in the *Configuration Upgrade Guide*

Policy line configuration class

As a result of modularizing lines of business, every line of business must have a Gosu configuration class defined within its policy line package. The class extends the base `PolicyLineConfiguration` class and can override certain methods and properties as needed, such as properties for rate routine configuration.

Logical matcher replaces checkForDuplicates method

In PolicyCenter 7.0, when adding new policy line coverages, you were required to write code to check for duplicates in coverages, exclusions, and conditions. You then merged any duplicates found so as to present a list of clauses

with no duplicates. Beginning with PolicyCenter 8.0.0, logical matchers automatically resolve duplicates with no additional configuration.

Blank availability rows are not valid

Previously, it was possible and valid to create blank rows in an availability lookup table. Beginning with PolicyCenter 8.0.0, blank rows are invalid and, if present, prevent the PolicyCenter server from starting. Validation within Product Designer prevents you from committing changes that include blank availability rows. However, it is possible to have such rows in your configuration resulting from upgrades or due to manually editing a lookup table XML file outside of Product Designer.

See also

• *Product Model Guide*

Handling visibility of coverages

As a result of modularizing lines of business, the way PolicyCenter handles visibility of coverages has changed. Previously, all coverages were discovered in an iterator and the `visible` property of PCF pages was used to render available coverages. Beginning with this release, the needed coverages are selected within the iterator and then all available coverages are rendered without using the `visible` property.

See also

• *Product Model Guide*

Changes related to integrating with ContactManager

For changes in PolicyCenter related to integrating with ContactManager and changes in ContactManager itself, see the *Guidewire Contact Management Guide*.

In particular, see the following:

• *Guidewire Contact Management Guide*

New and changed in globalization in 8.0.0

In PolicyCenter 8.0, the *Globalization Guide* replaces the *Localization Guide*. The name change reflects the increased scope of the guide. In earlier releases, the guide described features for localizing PolicyCenter to a specific locale. In 8.0, the guide describes features for configuring Guidewire PolicyCenter to operate in a global environment.

New in globalization in 8.0.0

Separation of language and regional format

New in PolicyCenter 8.0 is a separation between language and regional format. As part of the change in how PolicyCenter 8.0 provides support for globalization, users can now set their preferences for language and for regional formats independently of each other.

The **International** choice on the **Options** menu ◉ provides both **Language** and **Regional Formats** submenus. The **International** choice on the **Options** menu ◉ is available only if you configure PolicyCenter 8.0 with multiple languages or regional formats.

Base configuration language support

In the base configuration, Guidewire has a single default display language, English. If your installation has more than one language, you can select among the multiple languages by using the PolicyCenter **Options**→**Language** menu. PolicyCenter uses the `LanguageType` typelist to defines the set of language choices that the **Language** menu shows.

File localization.xml

In 8.0, Guidewire splits file `localization.xml` into multiple files, each of which contains formatting and parsing rules for a single langue only. Each base configuration, each `localization` file resides in the following location in Studio:

configuration→**Localizations**→*LocaleCode*

Theterm *LocaleCode* is a standard ISO locale code, such as `en_US` or `ja_JP`.

In a language module, each `localization.xml` file is in a `config/locale.LocaleCode` folder.

In the PolicyCenter 8.0 release, the existence of a `localization.xml` file in each region is optional. If you do not supply the relevant language configuration data in a `localization.xml` file, PolicyCenter uses the ICU library

defaults for that language. See "The International Components for Unicode (ICU) Library" on page 21 in the Globalization Guide for details.

See also

- *Globalization Guide*

Base configuration region support

In 8.0, Guidewire provides support for the following core regions in the base configuration:
- United States (English)
- Great Britain (English)
- Australia (English)
- Canada (English)
- Canada (French)
- France (French)
- Germany (German)
- Japan (Japanese)

Regional formats control how PolicyCenter formats dates, times, numbers, monetary amounts, and the names of people and companies.

See also

- *Globalization Guide*

Base configuration currency support

In PolicyCenter 8.0, Guidewire provides support in the base configuration for the following currencies:
- `AUD` – Australian dollar
- `CAD` – Canadian dollar
- `EUR` – European Union euro
- `GBP` – British pound
- `JPY` – Japanese yen
- `RUB` – Russian ruble
- `USD` – U.S. dollar

File currencies.xml split into multiple currency.xml files

In the 7.x releases, Guidewire provides a single `currencies.xml` file that contains information on all defined currencies. In 8.0, Guidewire splits the `currencies.xml` file into multiple `currency.xml` files. Each `currency` file contains information on a single currency only.

PolicyCenter stores each `currency` file in its own location in Studio, in a separately labeled currency folder:

configuration→**config**→**currencies** *currencyCode*

For example, the `currency.xml` file for the Australian dollar is in the following location in Studio:

configuration→**config**→**currencies**→**aud**

The purpose of this change is to reduce merge conflicts during upgrade.

See also

- *Globalization Guide*

Base configuration geographical support

In PolicyCenter 8.0, Guidewire provides support in the base configuration for the following geographical areas:

- `AU` – Australia
- `CA` – Canada
- `DE` – Germany
- `FR` – France
- `GB` – Great Britain
- `JP` – Japan
- `US` – Unites States

File address-config.xml split into multiple address-config.xml files

In the 7.x releases, Guidewire provides a single `address-config.xml` file that contains address information for all supported countries. In 8.0, Guidewire splits the single `address-config.xml` file into multiple `address-config` files, each stored under the `geodata` folder in its own folder named after the country code.

For example, the `address-config.xml` file for the address-related information in Australia is in the following location in Studio:

configuration→config→geodata→AU

See also

- *Globalization Guide*

New country.xml files

New in 8.0 are `country.xml` files that set address formatting for a single country. For example, you can set which address fields to show, and in what order, for a specific country in this file.

PolicyCenter stores separate copies of `country.xml` under the `geodata` folder, each in its own *countryCode* folder.

For example, the `country.xml` file for the address-related information in Australia resides in the following location in Studio:

configuration→config→geodata→AU

See also

- *Globalization Guide*

File zone-config.xml split into multiple zone-config.xml files

In the 7.x releases, Guidewire provides a single `zone-config.xml` file that contains zone-related information for all supported countries. In 8.0, Guidewire provides multiple `zone-config.xml` files, each of which is used to define items like address autofill fields.

PolicyCenter stores the `zone-config` files under the `geodata` folder in individual *countryCode* folders.

For example, the `zone-config.xml` file for the zone information in Australia is in the following location in Studio:

configuration→config→geodata→AU

See also

- *Globalization Guide*

Phone number data and phone validator Gosu class

In 8.0, PolicyCenter uses a new Gosu `PhoneValidator` class as the default mechanism to validate phone number correctness.

```
gw.api.validation.PhoneValidator
```

See also

Globalization Guide

Automatic address completion and fill-in plugin functionality

Previous to 8.0, PolicyCenter used geographic zone definitions that you loaded into PolicyCenter to provide address autofill and autocompletion functionality. In 8.0, PolicyCenter provides a new plugin, `IAddressAutocompletePlugin`, that you use to interface to external services for autocomplete data.

In 8.0, to illustrate a common use, you can provide a plugin implementation that autofills an address using data that comes from custom code or from an external service. As you enter a postal code value into the address field, the plugin implementation can detect the postal code value and autofill other address fields based on the entered value.

The default Java implementation class for the plugin, `DefaultAddressAutocompletePlugin`, duplicates the pre-8.0 behavior.

The `AddressAutocompleteUtil` utility class from earlier releases now calls static methods on the Address AutoComplete plugin, so you do not need to modify your existing code.

> **Note:** Guidewire has configured the new (in 8.0) `GlobalAddressInputSet` PCF files to use the new autocompletion plugin functionality. If you have pre-8.0 PCF files that you use for address entry, then Guidewire strongly recommends that you convert these files to the use of the new `GlobalAddressInputSet` PCF files. See also "Changes to address configuration in 8.0" on page 179.

See also

- *Globalization Guide*
- *Integration Guide*

Ability to provide a sort order for the typecodes in a typelist

New in PolicyCenter 8.0 is the ability to set the sort order (`priority` value) for the typecodes in a typelist by language. PolicyCenter determines the typecode sort order from a file named after the typelist, with a file extension of `.sort`. The typecode `priority` values set in the `.sort` file overrides the typecode `priority` values set in the typekey definition file for that language. PolicyCenter stores the sort order information, by language, in the typelist table.

This functionality is useful primarily for configurations that support more than one language, with one of those languages being Japanese. For example, one can use a sort order file to provide a sort order for Japanese provinces, which are customarily ordered from North to South (Hokkaido, Aomori, Iwate, …).

Guidewire does not provide any sort order files in the base configuration. You must place any `.sort` file that you create in the appropriate `configuration` folder. For example, for Japanese, place the file in the following location in Studio:

configuration→config→Localizations→ja_JP

> **IMPORTANT** Any change that you make to a typelist sort order file triggers a database upgrade.

Language module installation utility

New in 8.0 is the ability to install a Guidewire-provided language module using the PolicyCenter `gwpc` utility. You use this utility to install or upgrade an 8.0 language module.

See also

- *Globalization Guide*

Globalization-related configuration parameters

The following configuration parameters, defined in `config.xml`, are new in PolicyCenter 8.0:

- `DefaultApplicationLanguage`
- `DefaultPhoneCountryCode`
- `DefaultNANPACountryCode`
- `AlwaysShowPhoneWidgetRegionCode`
- `MulticurrencyDisplayMode`

In 8.0, Guidewire has added configuration parameter `MulticurrencyDisplayMode` that controls whether the application operates in single currency rendering mode or multicurrency rendering mode. The application server does not start if you change the value of `MulticurrencyDisplayMode` from `MULTIPLE` to `SINGLE`. Changing the value of `MulticurrencyDisplayMode` from `SINGLE` (the default) to `MULTIPLE` is irreversible unless you drop the application database.

> **IMPORTANT** If you are integrating multiple Guidewire applications, you must set the value of `DefaultApplicationCurrency` and `MulticurrencyDisplayMode` to be the same in each Guidewire application.

See also

- *Configuration Guide*

Globalization-related typelists

The following typelists related to globalization are new or have a new function in PolicyCenter 8.0.

Typelist	Description
LocaleType	Use to define supported regions. The value of configuration parameter `DefaultApplicationLocale` must be a typecode from the `LocaleType` typelist.
LanguageType	Use to define supported languages. The value of configuration parameter `DefaultApplicationLanguage` must be a typecode from the `LanguageType` typelist. Typelist `LanguageType` exists in 7.x releases. However, in the 7.x releases, the typelist specifies an overloaded set of combined language and locale values. In 8.0, Guidewire constrains the `LanguageType` typelist to specify a set of language values only.

Ability to set first day of week by region

Guidewire has added an optional `firstDayOfWeek` attribute to the `<GWLocale>` element in `localization.xml`. You use this attribute to set the day on which a week begins for a particular region. If you do not provide a value for this attribute, then PolicyCenter uses the default ICU (International Components for Unicode) value for the region.

The base configuration uses the following ICU first days of the week for the listed regions:

Sunday	• en_AU
	• en_CA
	• en_US
	• fr_CA
	• ja_JP
Monday	• de_DE
	• en_GB
	• fr_FR

See also

- For more information how PolicyCenter uses the ICU library, see the *Globalization Guide*.

New in the PolicyCenter 8.0 data model

The following items are new in the PolicyCenter 8.0 data model:
- "New phone fields on Contact and Person entities" on page 178
- "New globalization-related fields on Contact and Address entities" on page 178
- "Monetary amount data model element in PolicyCenter 8.0" on page 178
- "Gosu MonetaryAmount entity in PolicyCenter 8.0" on page 179
- "Monetary amount widgets in PolicyCenter 8.0" on page 179
- "Attributes on <localization> column element" on page 179

New phone fields on Contact and Person entities

In 8.0, Guidewire has added the following new phone-related fields to `Contact` and `Person` entities:

Entity	New phone-related field
Contact	• FaxPhoneCountry • HomePhoneCountry
Person	CellPhoneCountry

New globalization-related fields on Contact and Address entities

In 8.0, Guidewire has added the following globalization-related fields to `Contact` and `Address` entities:
- `Contact.NameKanji`
- `Contact.FirstNameKanji`
- `Contact.LastNameKanji`
- `Contact.Particle`
- `Address.AddressLine1Kanji`
- `Address.AddressLine2Kanji`
- `Address.CityKanji`
- `Address.CEDEX`
- `Address.CEDEXBureau`

Monetary amount data model element in PolicyCenter 8.0

In 8.0, Guidewire uses a new data model element called `<monetaryamount>` to represent a specific money amount and its associated currency. You can use the `<monetaryamount>` element anywhere you can use a `<column>` element, for example, on `entity` or `extension` objects.

The `<monetaryamount>` element is different from other data model fields in that it corresponds to two columns in the database:
- The first column represents the numerical amount of money involved, for example 500.
- The second column represents the currency associated with the numerical amount, for example, Canadian dollars.

PolicyCenter implements that amount column internally as a standard database column with a datatype of `money` or one of its related derivatives such as `positivemoney` or `negativemoney`, for example. PolicyCenter implements the currency type column internally as a typekey of the `Currency` typelist.

Gosu MonetaryAmount entity in PolicyCenter 8.0

New in 8.0 is a Gosu `MonetaryAmount` entity that encapsulates a money amount and a currency. To create a new instance of the `MonetaryAmount` object, pass a `BigDecimal` amount and a `Currency` code to the constructor, for example:

```
new MonetaryAmount(500, Currency.TC_AUD)
```

It is possible to use `MonetaryAmount` objects with Gosu arithmetic operators. However, in any arithmetic operation, the currency types for `MonetaryAmount` objects must match. For example, the following operation throws an exception:

```
(500, USD) + (500, JPY)
```

Monetary amount widgets in PolicyCenter 8.0

In 8.0, PolicyCenter provides several new PCF widgets that work with monetary amounts within the PolicyCenter user interface. The following list describes these new widgets.

PCF name	Description
MonetaryAmountCell	Use to display `MonetaryAmount` values in PolicyCenter
MonetaryAmountInput	Use to enter `MonetaryAmount` values in PolicyCenter

Attributes on <localization> column element

New in 8.0, are two new attributes on the `<localization>` subelement on `<column>`. The following list describes each attribute:

Attribute	Type	Description
unique	Boolean	If you set this attribute to `true`, PolicyCenter prohibits duplicate values in the set of value for each language.
nullok	Boolean	If you set this attribute to `true`, PolicyCenter flags missing entries that it finds during a database consistency check.
		For example, suppose that PolicyCenter defines more than one language. In this case, PolicyCenter stores values for the default application language in the main database table of the entity. PolicyCenter stores the values for the additional languages in a separate localization table.
		During consistency checks, PolicyCenter identifies entries in the main database tables for which no corresponding entries exist in the localization table. Values identified as missing are warnings only. A missing language value does not prevent the database from starting.

Changes in globalization in 8.0.0

Changes to address configuration in 8.0

Previous releases of PolicyCenter used a large number of PCF files to render addresses within PolicyCenter. Many of these PCF files used one mode for each country.

With PolicyCenter 8.0, Guidewire provides three PCF modes that cover the countries/regions that PolicyCenter 8.0 supports in the base configuration. Each PCF mode defines the address field order used in that country/region and covers the following:

- The field order for the address-entry edit view
- The field order for the address read-only text view

The following table lists the new modes and the countries/regions that each covers.

PCF mode	Country
BigToSmall	Japan
PostcodeBeforeCity	France and Germany
Default	All other countries

ClaimCenter 8.0 changes the PCF files that render addresses and the `AddressOwner` classes to use the new `GlobalAddressInputSet`. Guidewire expressly designed `GlobalAddressInputSet` to be the only address input PCF for use in PolicyCenter. Guidewire strongly recommends that you convert any existing, pre-8.0 address entry PCF file to use the new `GlobalAddressInputSet` PCF.

PolicyCenter uses new Gosu class `AddressFormatter` to format the address display fields. You can extend `AddressFormatter` to handle address formatting for additional countries.

In addition to the new Gosu `AddressFormatter` class, PolicyCenter uses a new Gosu `NameFormatter` class to format names appropriately for internationalization.

See also

- *Globalization Guide*

Changes to admin data localization

Typically, admin data is customer-defined entities with fields that contain names or phrases that need to be shown in the user's language. For example, the description and subject fields on an activity pattern are administration data.

You enter localized admin data directly within Guidewire PolicyCenter, in a table that is visible at the bottom of the screen. This table contains a row for each enabled language in the application. The row contains, by language, fields for each element on that screen that you can localize. PolicyCenter only shows the table if there is more than one defined language in the application.

In previous releases, only a limited number of screens in the base configuration contained the localization table. PolicyCenter 8.0 provides the ability to localize a much larger set of administration data. It is also possible for you to configure the application and add or remove localization tables by updating the data model and the appropriate PCF file.

Guidewire builds the localization table as a single PCF. PolicyCenter includes this PCF (`LocalizedValuesDV.pcf`) in each parent PCF with localizable data.

Changes to globalization-related typelists

Guidewire has made the following changes to typelists in 8.0:

Typelist	Changes
Country	In 8.0, Guidewire implements the `Country` typelist consistently within the Guidewire core applications, including ContactManager. Compared with previous releases, Guidewire has changed the names of some countries on the typelist and added new country names. The names on the `Country` typelist reflect the Unicode Common Locale Data Repository (CLDR) country names.
Currency	Contains currency code and similar information for the supported currencies.
JurisdictionType	Although, PolicyCenter defines jurisdiction categories in this typelist, PolicyCenter 8.0 does not use any of the categories or types. Guidewire intends this typelist for use in future releases. **See also** • *Globalization Guide*

Typelist	Changes
State	Guidewire has modified the base configuration `State` typelist to include state definitions for the following countries: • AU • CA • DE • JP • US **See also** • *Globalization Guide*
StateAbbreviation	Guidewire has added the `StateAbbreviation` typelist in 8.0. This typelist includes abbreviations for states in the following countries: • Australia • Canada • Germany • United States The purpose of the typelist is to disambiguate abbreviations for state names that exist in multiple countries. For example, the abbreviation "WA" is used in the United States for the state of Washington and in Australia for the state of Western Australia. **See also** • *Globalization Guide*

See also

• *Globalization Guide*

Changes to files used with globalization

Guidewire has modularized certain files used during localization so that each copy of the file affects a single region, language, regional format preference, or currency only. This is in contrast to previous PolicyCenter releases in which a file such as `localization.xml` contained data about more than one locale. In PolicyCenter 8.0, Guidewire requires a separate `localization.xml` file to store information about the en_US locale and the fr_FR locale. As there can potentially be multiple copies of a file, the location of each copy of the file must be unique.

In particular, the following files have change their name, their functionality, their location, or all three:

File	Changes
• `address-config.xml` • `country.xml` • `zone-config.xml`	Files `address-config.xml`, `country.xml`, and `zone-config.xml` now contain information specific to a single country only. All base configuration copies of these files reside in Studio in the following location, with **XX** being the country code: **configuration→config→geodata→XX** Substitute a standard ISO country code for **XX**. In the base PolicyCenter 8.0 release, Guidewire provides support for the following countries: • AU • CA • DE • FR • GB • JP • US
`currencies.xml`	Contains currency code and similar information for the supported currencies. In single currency mode, currency is defined in `localization.xml` files under locale folders. See the *Globalization Guide*. In multiple currency mode, currency is defined in `currency.xml` files in currency folders, such as usd, located in the `currencies` folder.

File	Changes
`config.xml`	Globalization parameters new in PolicyCenter 8.0: • `DefaultApplicationLanguage` • `DefaultPhoneCountryCode` • `DefaultNANPCCountryCode` • `AlwaysShowPhoneWidgetRegionCode` Globalization parameters with modified functionality in PolicyCenter 8.0: • `DefaultApplicationLocale` • `MulticurrencyDisplayMode` **See also** • *Configuration Guide*
`display.properties`	There are now multiple `display.properties` files, one for each local folder. See the *Globalization Guide*.
`fieldvalidators.xml`	You define national field validators in `fieldvalidtors.xml` files located in country-specific folders in the `fieldvalidators` folder. **See also** • *Globalization Guide*
`language.xml`	This file is optional. In the base configuration, Guidewire uses this file to override the ICU default collation settings for the German language. All other base configuration languages use the ICU library default collation settings. **See also** • *Globalization Guide*
`localization.xml`	In 8.0, Guidewire splits file `localization.xml` into multiple files, each of which contains formatting and parsing rules for a single langue only. Each base configuration `localization.xml` file resides in the following location in Studio: **configuration→Localizations→*LocaleCode*** For *LocaleCode* substitute a standard ISO locale code, such as en_US. In the PolicyCenter 8.0 release, the existence of a `localization.xml` file for a language is optional. If you do not supply the relevant language-configuration data in a `localization.xml` file, then PolicyCenter uses the ICU library defaults for that language. **See also** • *Globalization Guide*
`nanpa.properties`	Area codes defined by the North American Numbering Plan Administration (NANPA). You find this file in Studio in the following location: **configuration→config→phone** **See also** • *Globalization Guide*

List sort methods support optional locale sensitive sorting

Some Gosu enhancement methods for sorting collections now support optional locale-sensitive sorting. The new method signatures are on the methods `sort`, `sortDescending`, `orderBy`, `orderByDescending`.

See also

• *Gosu Reference Guide*

New and changed in system administration in 8.0.0

New in system administration in 8.0.0

Phone number normalizer work queue

PolicyCenter 8.0 includes a Phone Number Normalizer work queue. This work queue calls the registered plugin that implements the `IPhoneNormalizerPlugin` interface to convert phone numbers from the PolicyCenter 7.0 single-field standard to the PolicyCenter 8.0 multi-field standard.

See also

• *Upgrade Guide*

Changed in system administration in 8.0.0

Changes to database configuration

As of PolicyCenter 8.0, the `<database>` element is located in a separate file, `database-config.xml`, rather than the `config.xml` file. The configuration upgrade includes an automated step to modify the database configuration. Guidewire has made the following changes to the database configuration:

• The `<database>` element, now in `database-config.xml`, no longer contains subelements with the following syntax:

```
<param name="name" value="value">
```

• For Oracle, the `<tablespacemapping>` elements have been replaced with a single `<tablespaces>` element. The `<tablespaces>` element is contained in an `<ora-db-ddl>` parent element. The `<tablespaces>` element includes the attributes `admin`, `index`, `op`, `staging`, `typelist`, and `lob`. These attributes correspond to the logical tablespaces defined in PolicyCenter. You can use these attributes to map tablespaces that you have created to the logical tablespaces.

• For SQL Server, the `<tablespacemapping>` elements have been replaced with a single `<mssql-filegroups>` element. The `<mssql-filegroups>` element is contained in an `<mssql-db-ddl>` parent element. The `<mssql-filegroups>` element includes the attributes `admin`, `index`, `op`, `staging`, `typelist`, and `lob`. These attributes

correspond to the logical tablespaces defined in PolicyCenter. You can use these attributes to map file groups that you have created to the logical tablespaces.

- If a `<tablegroup>` element was contained in a `<database>` element that had an `env` attribute defined, the upgrade copies the `env` attribute onto the `<tablegroup>` element.

- If any of the following `<database>` attributes are defined, the upgrade copies them over to the `<database>` element in `database-config.xml`: `addforeignkeys`, `autoupgrade`, `checker`, `dbtype`, `env`, `name`, `printcommands`. The schema for these attributes has not changed.

- If any comments exist within the `<database>` element, the upgrade copies these comments to the `<database>` element in `database-config.xml`.

- If the `driver` attribute of the `<database>` element equals `dbcp`, the upgrade adds a `<dbcp-connection-pool>` element and copies the `jdbcUrl` parameter to the `jdbc-url` attribute of the `<dbcp-connection-pool>` element. If the original configuration did not include a `jdbcUrl` parameter, then the upgrade logs an error. If a `passwordFile` attribute is specified on the `<database>` element of the old configuration, the upgrade transfers the `passwordFile` attribute to the `<dbcp-connection-pool>` element. The upgrade converts any of the following parameters defined in the old configuration to attributes on the `<dbcp-connection-pool>` element:
  - `maxActive`
  - `maxIdle`
  - `maxWait`
  - `minEvictableIdleTimeMillis`
  - `numTestsPerEvictionRun`
  - `testOnBorrow`
  - `testOnReturn`
  - `testWhileIdle`
  - `timeBetweenEvictionRunsMillis`
  - `whenExhaustedAction`

- If the `driver` attribute of the `<database>` element equals `dbcp` and any of the following attributes are set, the upgrade creates a `<reset-tool-params>` element within the `<dbcp-connection-pool>` element:
  - `collation`
  - `oracle.tnsnames`
  - `system.username`
  - `system.password`

The upgrade then transfers any of these attributes that are defined to the new `<reset-tool-params>` element.
- If the `driver` attribute of the `<database>` element equals `jndi`, the upgrade adds a `<jndi-connection-pool>` element and copies the `jndi.datasource.name` parameter to the `datasource-name` attribute of the `<jndi-connection-pool>` element. If the original configuration did not include a `jndi.datasource.name` parameter, then the upgrade logs an error.
- If the old configuration includes an `<upgrade>` element within the `<database>` element, the upgrade adds an `<upgrade>` element to the `<database>` element of the new configuration.
- If the old configuration contains an `<upgrade>` element that includes an `oracleMarkColumnsUnused` attribute, the upgrade converts the attribute to a `deferDropColumns` attribute, preserving the value.
- If the old configuration contains an `<upgrade>` element that includes a `verifySchema` attribute, the upgrade copies this attribute to `<upgrade>` element of the new configuration.
- If the old configuration contains an `<upgrade>` element that contains an `<oracleddloptions>` or `<sqlserverddlopts>` element, the upgrade logs a warning. You must upgrade these elements manually.
- If the old configuration includes a `<databasestatistics>` element within the `<database>` element, the upgrade copies the `<databasestatistics>` element to the `<database>` element of the new configuration.
- For Oracle databases, if the `<database>` element includes any of the following parameters, the upgrade creates an `<oracle-settings>` element within the `<database>` element of the new configuration:
  - `queryRewriteEnabled`
  - `statisticsLevel`
  - `stored.outlines`
  - `UseDbResourceMgrCancelSql`

The upgrade converts any of the above parameters to attributes on the new `<oracle-settings>` element. The attributes have the following names:
  - `query-rewrite`
  - `statistics-level-all` (if any value is set for `statisticsLevel` in the old configuration, the upgrade sets the `statistics-level-all` attribute to true in the new configuration. The value `ALL` was the only valid value for the `statisticsLevel` parameter in the old configuration.)
  - `stored-outline-category`
  - `db-resource-mgr-cancel-sql`
  - For SQL Server databases, if the `<database>` element includes either the `msjdbctracelevel` or `msjdbctracefile` parameter, the upgrade adds a `<sqlserver-settings>` element within the `<database>` element of the new configuration. The upgrade then converts the `msjdbctracelevel` and `msjdbctracefile` parameters to `jdbc-trace-level` and `jdbc-trace-file` attributes on the `<sqlserver-settings>` element.
  - For SQL Server databases, if the `unicodecolumns` parameter is defined in the old configuration, the upgrade adds a `unicodecolumns` attribute to the `<sqlserver-settings>` element of the new configuration. If the `<sqlserver-settings>` element has not yet been created, the upgrade creates the element.
  - If any `<tablespacemapping>` elements are defined in the old configuration, the upgrade creates an `<upgrade>` element within the `<database>` element of the new configuration if one does not yet exist. The upgrade then does the following, depending on the database type:
    - For Oracle, the upgrade adds an `<ora-db-ddl>` element within the `<upgrade>` element of the new configuration, if an `<ora-db-ddl>` element is not yet defined. The upgrade then adds a `<tablespaces>` element to the `<ora-db-ddl>` element and converts each `<tablespacemapping>` element to an attribute on the `<tablespaces>` element. The upgrade then adds an `<ora-lobs>` element to the `<ora-db-ddl>` element and sets the `<ora-lobs>` attribute `type` to `BASICFILE`.
    - For SQL Server, the upgrade adds an `<mssql-db-ddl>` element within the `<upgrade>` element of the new configuration, if an `<mssql-db-ddl>` element is not yet defined. The upgrade then adds a `<mssql-filegroups>` element to the `<mssql-db-ddl>` element and converts each `<tablespacemapping>` element to an attribute on the `<mssql-filegroups>` element.

Changes to work queue configuration

Guidewire has updated the `<work-queue>` element that defines the configuration of a work queue. These changes accommodate improvements to how work queues detect and handle orphaned work items.

The `<work-queue>` element now includes a `retryInterval` attribute. This attribute defines how long in milliseconds to wait before retrying a failed item. The default value is `0`, meaning PolicyCenter will retry processing the item immediately. During the upgrade, the Configuration Upgrade Tool sets `retryInterval` to `0` if `checkInAfterError` was `true`, or to the current value of `progressinterval` if `checkInAfterError` was not `true`.

The `<work-queue>` element also now includes a `retryLimit` attribute. This attribute defines how many times PolicyCenter retries a failed work item for the work queue. If you do not specify a `retryLimit` value for a work queue, PolicyCenter uses the value of the `WorkItemRetryLimit` configuration parameter in `config.xml` as the default value.

PolicyCenter 8.0 no longer includes the `minpollinterval`, `orphansFirst`, or `checkInAfterError` attributes. The Configuration Upgrade Tool removes these attributes.

Changes to scheduled purge batch processing

Batch processing that purges database records has been added to the default batch processing schedule. These processes include:

- Process History Purge
- Purge Workflows
- Purge Workflow Logs
- WorkItem Set Purge
- Work Queue Instrumentation Purge

Other batch processing that purges database records was already included in the schedule.

During the configuration upgrade, review the updated schedule in `scheduler-config.xml` and make changes as needed.

Changes to database statistics updating

Commands for updating database statistics are now options for the `system_tools` command. If you have custom scripts to schedule database statistics updating, you must modify those scripts.

PolicyCenter 7.0	Use `maintenance_tools` with one of the following options:
	
	`-startprocess dbstatistics`
	`-startprocess incrementaldbstatistics`

PolicyCenter 8.0	Use `system_tools` with one of the following options:
	
	`-updatestatistics description false`
	`-updatestatistics description true`
	`-cancelupdatestats`
	`-getupdatestatsstate`
	Specify `true` to update database statistics only for tables exceeding the change threshold. The change threshold is defined by the `incrementalupdatethresholdpercent` attribute of the `databasestatistics` element in file `database-config.xml`.
	Specify `false` to gather full database statistics.

Now possible to cancel database statistics updating

You can now cancel the database statistics updating process while it is running. This capability comes with changes to how you run database statistics updating.

To cancel the process that updates database statistics, use the following command:

```
system_tools -password password -cancelupdatestats
```

In PolicyCenter 8.0, the database statistics updating process can be paused just as with other work queues. Use the **Work Queue Info** page to pause an in-progress work queue.

Commands for database statistics update statements moved

The `-getdbstatisticsstatements` and `-getincrementaldbstatisticsstatements` commands to get full or incremental statistics updates statements are also now on `system_tools` instead of `maintenance_tools`.

User batch processing renamed

With this release, Guidewire has renamed User batch processing to Team Screens batch processing. The process collects summary counts for team screens.

See also

- "Changes to the team tab" on page 157
- *System Administration Guide*

New and changed in Gosu in 8.0.0

New in Gosu in 8.0.0

New assert statement

To confirm expectations and enforce a programmatic contract, Gosu has an `assert` statement with the same semantics and syntax as in Java. By default, `assert` statements have no effect in PolicyCenter. To enable assertions, you must add the `-ea` flag on the JVM that hosts the application or Studio. For more details, see the *Gosu Reference Guide*.

Local variable now optional in For loop declarations

In previous releases, `for` loop declarations had to declare a local variable:

```
for (x in 1..10) { print( x ) }
```

In some cases, the body of the loop may not use the variable.

```
for (x in 1..10) { print( "hello!") }
```

In version 8.0, if the type to iterate across is a Gosu *interval,* you can optionally omit the variable declaration including the word `in`. For example:

```
for (1..10) { print( "hello!") }
```

There is no upgrade impact of this change.

Final variable initialization separate from declaration

In previous releases, Gosu supported declaring a local variable final using the syntax `final var` and setting a value on the same statement that declares the variable:

```
final var b : int = 10
```

In this release, the previous syntax continues to be supported. There is a new option to declare the variable final but not immediately initialize it with a value. You must set the value eventually in that function for all possible code paths.

For example, you can now write something like this:

```
function foo() {
  final var b : int
  b = 10
}
```

The Gosu compiler verifies all code paths have initialization exactly once. In other words, any if statements or other flow control structures must set the variable and only once.

The following code is correct:

```
function foo(a : boolean) {
  final var b : int
  if(a) {
    b = 0
  } else {
    b = 1
  }
}
```

However, if you remove the else branch it is invalid because as the final variable is initialized only if a is true.

```
function foo(a : boolean) {
  final var b : int // INVALID CODE, UNINITIALIZED IF "a" IS FALSE
  if (a) {
    b = 0
  }
}
```

There is no upgrade impact to this change.

Final class variables

As in previous releases, the final keyword can be used on class variables. The class can set the variable only once, and only from the declared class, not by subclasses.

In this release, the final keyword can now be used on a class variable declaration without immediately initializing the variable. If you do not immediately initialize the variable, all class constructors must initialize it in all possible code paths.

For example, the following syntax is now valid because all constructors initialize it once in each code path:

```
class TestABC {
  final var _name : String

   construct() {
    _name = "hello"
  }
   construct(b : boolean){
    _name = "there"
  }
}
```

However, the following is invalid because one constructor does not initialize the final variable:

```
class TestABC {
  final var _name : String // INVALID CODE, ALL CONSTRUCTORS MUST INITIALIZE THIS IN ALL CODE PATHS

   construct() { // does not initialize the variable
  }
   construct(b : boolean){
    _name = "there"
  }
}
```

There is no upgrade impact to this change.

The new operator is now optionally a statement

In previous releases, the `new` operator is an expression but never a statement. For example, you could type:

```
var s = new String()
```

However, you could not type as a complete statement:

```
new String()
```

In version 8.0, the `new` operator is now a statement. For some types, this may not be useful.

However, if the constructor for the object triggers code that saves a copy of the new object, the return value from `new` may be unnecessary. This change may permit more concise code in some cases.

See also

• *Gosu Reference Guide*

Gosu supports annotations on function parameters

The Gosu language now supports annotations on function parameters, including Gosu block declarations. In some cases you need to explicitly add uses lines to declare which annotation class to use. See the *Gosu Reference Guide*.

There are currently no compile time APIs for changing IDE behavior based on the annotations.

You can now access function parameter annotations using Java reflection APIs.

Named arguments in annotations

Gosu annotations now support the named arguments calling convention:

```
@Param(:FieldName = "str", :FieldDescription = "The String value to parse")
```

For related information about named arguments, see the *Gosu Reference Guide*.

New support for code coverage tools for Gosu code

Code coverage tools analyze the degree of testing of programming code. For Gosu code in Studio, PolicyCenter now supports code coverage tools that use Java class files as input to bytecode analysis. Gosu compiles to Java class files. See the *Gosu Reference Guide*. This feature requires tools that use Java class files, not source code, as input.

The entity touch API is changed and no longer deprecated

PolicyCenter protects entity instances from concurrent access through a version property that exists on all versionable entities. For details, see the *Gosu Reference Guide*.

In rare cases, it may be desirable to force the version number of an entity to increment even if there is no known change to the entity yet. To force PolicyCenter to increment the entity version number, update the modified time, and call preupdate rules on the object, call the `touch` method on the entity instance. The method takes no arguments.

This method existed in earlier versions of PolicyCenter, and was eventually deprecated. In PolicyCenter 8.0, the method is undeprecated. The version of `touch` in this release has slightly different behavior from previous releases that will affect very few customers. In 7.0, the `touch` method set the object property `BeanVersion` temporarily to `null`. In 8.0, the `touch` method immediately increments the `BeanVersion` property.

For more discussion about when to use this method, see the *Gosu Reference Guide*.

Changed in Gosu 8.0.0

Gosu is now case sensitive in most cases

The Gosu language is now case sensitive in most ways. In previous releases, case-sensitivity was encouraged. Now case-sensitivity is required for nearly all types. For example, if a type is declared as `MyClass`, you cannot type it as `myClass` or `myclass`.

WARNING For symbols representing Guidewire entity type data, Gosu continues to be case insensitive. However, it is strongly recommended to write your code as if entity types are case-sensitive.

During upgrade, fix any compilation errors that Studio flags due to this change.

Bundle changes from Gosu and Java

A *bundle* refers to a programmatic abstraction of a database transaction and the set of database rows to update. See the *Gosu Reference Guide*.

In previous releases, PolicyCenter represented a bundle in Gosu as the class `gw.transaction.Bundle`. The class still exists, but it is now in a different package. The new fully-qualified name is `gw.pl.persistence.core.Bundle`. Customers usually do not declare the package directly in their code, but instead use APIs that return a bundle. You may get compilation errors during upgrade if you directly referenced the old package name.

Additionally, the new `Bundle` class is callable from Java code that uses the PolicyCenter 8.0 Java API. For details about this API, see "Java code changes" on page 205.

Gosu map enhancement method changes

Gosu collection classes rely on the collection classes from the Java language. However, Gosu adds enhancement methods on those objects. By using the enhanced Gosu collection classes, with a single line of code you can loop through collection items to perform actions, extract item information, or sort items. In this release, the enhancement methods changed slightly.

The map methods `filterKeys` and `filterValues` no longer exist.

These two methods were replaced by four different methods that are more flexible and named more clearly. The following table documents the new methods. The methods that have `retain` in the name are destructive. The methods with `filter` in the name create a new map and do not modify the original map.

Map method name	Description
`retainWhereKeys(keyFilter)`	Destructively removes all entries whose keys do not satisfy the `keyFilter` expression. Returns `true` if and only if this map changed as a result of the block. The key filter block must take one argument (a key) and return `true` or `false`.
`retainWhereValues(valueFilter)`	Destructively removes all entries whose values do not satisfy the `valueFilter` expression. Return true if this map changed as a result of the call. The value filter block must take one argument (a value) and return `true` or `false`.
`filterByKeys(keyFilter)`	Returns a new map that is a clone of the original map but without entries whose keys do not satisfy the `keyFilter` expression. The key filter block must take one argument (a key) and return `true` or `false`.
`filterByValues(valueFilter)`	Returns a new map that is a clone of the original map but without entries whose values do not satisfy the `valueFilter` expression. The key filter block must take one argument (a value) and return `true` or `false`.

Entity literal syntax is deprecated

Entity literals have been deprecated. Use query builders instead.

Entity literals have the form:

```
entityType(entityId)
```

For example:

```
User("34" /* John Smith */)
Group("3" /* Eastern Regional Center */)
```

For example, if you choose a `Group` or `User` from Studio SmartHelp in previous releases, Studio inserted an entity literal of that type into your code.

Replace these expressions with query builder expressions.

For example:

```
Query.make(Policy).compare(Policy#PublicID, Relop.Equals, myPublicId).select().AtMostOneRow
```

For more information, see the *Gosu Reference Guide*.

Entity methods loadByKey, loadByPublicID, and remove deprecated

The `Bundle` class methods `loadbyKey`, `loadByPublicID`, and `remove` are deprecated in version 8.0.

Instead of using `loadbyKey`, use the new method `loadBean`, which takes a `gw.pl.persistence.core.Key` object. Create a `Key` object with a constructor that takes the entity type and the numeric ID. For example:

```
a = gw.transaction.Transaction.getCurrent().loadBean(new Key(Address, 123))
```

Instead of using `loadbyPublicID`, use the query builder APIs. For example:

```
Query.make(Policy).compare(Policy#PublicID, Relop.Equals, myPublicId).select().AtMostOneRow
```

Instead of using `remove`, use the `Bundle` class method `delete`.

See also

• *Gosu Reference Guide*

Package names changed to reflect public and internal status

Many types not otherwise mentioned changed the package in version 8.0 to better reflect the status of the public or internal nature of the type.

The general rule is:

• Types in the package `com.guidewire.*` are internal. Do not use.
• Types in the `gw.*` package but not in a subpackage that contains the word `internal` are generally for customer use.

> **IMPORTANT** If in doubt whether a class or API is supported in PolicyCenter, contact Guidewire Customer Support.

Some classes changed from `com.guidewire.*` to `gw.*`. Refer to the following table for an example of most commonly used packages that changed.

Old package	New package
com.guidewire	gw.pl.exception
com.guidewire.commons.file	gw.pl.file
com.guidewire.commons.io	gw.pl.io

Old package	New package
com.guidewire.commons.util	gw.pl.util
com.guidewire.logging	gw.pl.logging
com.guidewire.main	gw.pl.simpletool
com.guidewire.modules	gw.pl.modules
com.guidewire.modules.pom	gw.pl.modules.pom
com.guidewire.testharness	gw.pl.testharness
com.guidewire.util	gw.pl.util
com.guidewire.util.csv	gw.pl.util.csv
com.guidewire.util.diff.lcs	gw.pl.util.lcs
com.guidewire.util.webservices	gw.pl.util.webservices
com.guidewire.util.webservices.axis	gw.pl.util.webservices.axis
com.guidewire.util.webservices.login	gw.pl.util.webservices.login
com.guidewire.util.xml	gw.pl.util.xml
com.guidewire.util.xml.helpers	gw.pl.util.xml.helpers
com.guidewire.pl.plugin	gw.pl.plugin
com.guidewire.pl.plugin.addressbook	gw.pl.plugin.addressbook
com.guidewire.pl.plugin.approval	gw.pl.plugin.approval
com.guidewire.pl.plugin.credentials	gw.pl.plugin.credentials
com.guidewire.pl.plugin.custompicker	gw.pl.plugin.custompicker
com.guidewire.pl.plugin.dbauth	gw.pl.plugin.dbauth
com.guidewire.pl.plugin.document	gw.pl.plugin document
com.guidewire.pl.plugin.management	gw.pl.plugin.management
com.guidewire.pl.plugin.messaging	gw.pl.plugin.messaging
com.guidewire.pl.plugin.search	gw.pl.plugin.search
com.guidewire.pl.plugin.security	gw.pl.plugin.security
com.guidewire.pl.plugin.util	gw.pl.plugin.util
com.guidewire.pl.plugin..validation	gw.pl.plugin.validation
com.guidewire.main	gw.pl.main
com.guidewire.external	gw.pl.external

You may get compilation errors due to these changes. In many cases, Studio can help you identify the correct package using the ALT-Enter keyboard shortcut.

For details of which packages changed in version 8.0, consult the API Differences file (apidiffs.txt) in the product.

Gosu concurrency API changes

Gosu has multiple APIs related to concurrency.

Scopes APIs deprecated

In version 7.0, the gw.api.web.Scopes API created maps pre-scoped to scopes such as request and session.

In version 8.0, the `gw.api.web.Scopes` API are now deprecated. Instead, use the `RequestVar` and `SessionVar` classes in the `gw.api.web` package. The `RequestVar` and `SessionVar` classes create a variable with a well-defined lifecycle and attachment point (either the request or session). It is strongly recommended to use `RequestVar` and `SessionVar` rather than the deprecated APIs or using the Java thread local API `java.lang.ThreadLocal<TYPE>`. For more details on these topics, see the *Gosu Reference Guide*.

LazyVar Split into two variants for locking and non-locking

In version 7.0, the `LazyVar` class implements what some people call a *lazy variable*. This means Gosu constructs it only the first time some code uses it. Because the `LazyVar` class uses the Java concurrency libraries, access to the lazy variable is thread-safe.

In version 8.0, the `LazyVar` class is removed. You must change all usages to one of two replacements:

- `LocklessLazyVar` – A lazy variable initialization that does not require thread safety.
- `LockingLazyVar` – A more direct replacement for `LazyVar`. Access to the lazy variable is thread-safe.

See the *Gosu Reference Guide*.

Inequality <> operator is now invalid

In 7.0, you could indicate inequality using either `<>` or `!=` expressions.

In 8.0, `<>` is invalid. Change usages to `!=`.

There is a code inspection option in Studio to detect *equality issues* that finds this issue.

The Gosu command line tool built-in editor removed

The Gosu command line tool in the `PolicyCenter/admin/bin` directory no longer has a built-in Gosu editor. If you run the `gosu.cmd` tool with no arguments, Gosu launches by default into the Gosu interactive shell. See the *Gosu Reference Guide*.

Change in Gosu named parameters usage

In 7.0, a function call (including all method calls) that included named parameters could use a mix of named and unnamed parameters.

In 8.0, if a function call includes a named parameter, all following parameters must be named parameters.

See the *Gosu Reference Guide*.

ArrayList.Size Property now size Method

The `ArrayList` property called `Size` has been removed. Change any usages to the `size` method, to match the native Java implementation.

Static abstract methods no longer allowed

Gosu no longer supports methods that use both the abstract and static modifiers. Abstract methods must be overridden, and static methods cannot be overridden. Thus, having either modifier is mutually exclusive of the other modifier.

chapter 32

New and changed in integration in 8.0.0

New in integration in 8.0.0

New phone number normalizer plugin

The new globalization framework for phone numbers includes a plugin to normalize phone numbers that users enter through the application and that enter the database through data import.

See also

• *Integration Guide*

New MTOM support for results of published WS-I web service

The W3C Message Transmission Optimization Mechanism (MTOM) is a method of efficiently sending binary data to and from web services as attachments outside the normal response body.

The following table compares version 7.0 and version 8.0 MTOM support.

Role of PolicyCenter	Sending or receiving	Version 7.0 support	Version 8.0 support
PolicyCenter consuming a WS-I web service	Data with initial request	--	--
	Data in response	Yes	Yes
PolicyCenter publishing a WS-I web service	Data with initial request	Yes	Yes
	Data in response	--	Yes, configurable

See also

• *Integration Guide*

Multi-threaded inbound integration

There may be cases that require high-performance data throughput for inbound integrations that require special threading or transaction features from the hosting J2EE/JEE application environment. It is difficult to interact with the application server's transactional facilities and write correct, thread-safe, high-performing code. PolicyCenter

includes tools that help you write such inbound integrations. You can focus on your own business logic rather than how to write thread-safe code that works safely in each application server.

The inbound integration API consists of several parts:

- A plugin interface called `InboundIntegrationPlugin` that defines the contract between PolicyCenter and inbound integration high-performance multi-threaded processing of input data.
- A built-in implementation of the plugin that reads text data in local files. Poll a directory in the local file system for new files at a specified interval. Send new files to integration code and process incoming files line by line.
- A built-in implementation of the plugin that reads inbound messages from a JMS message queue.

See also

- *Integration Guide*

New messaging destination option for messages without primary object

In the PolicyCenter messaging system, many messages are associated with a primary object such as a policy. For messages without a primary object, documentation refers to those messages as non-safe-ordered messages.

In version 7.0, there was a messaging destination option called Strict Mode. Strict Mode was either enabled or disabled using a check box in the user interface.

- If Strict Mode was enabled, messages without a primary object were sent in a strict order, and waited for an acknowledgement before sending the next message.
- If Strict Mode was disabled, messages without a primary object were immediately sent in a single thread, and did not wait for an acknowledgement before proceeding to the next message.

In version 8.0, both those modes are available but there is a third choice, which is a multi-threaded sending with no ordering and no waiting for an acknowledgement. The precise order of sending of messages without a primary object is non-deterministic.

The messaging editor offers three choices for the option **Messages Without Primary**:

- **Single thread**
- **Multi thread**
- **Strict Mode**

See also

- *Integration Guide*

Regenerating WSDL and XSD in web service collections (regen-from-wsc)

Web service collection (`.wsc`) files encapsulate the set of resources necessary to connect to a web service on an external system. If you view a web service collection in Studio and click the **Fetch Updates** button, Studio retrieves WSDL and XSD files from the servers that publish those web services. In this release, you can trigger the fetch updates process from a command line tool called `regen-from-wsc`.

See also

- *Integration Guide*

New PolicyEarnedPremiumAPI web service

There is a new web service called `PolicyEarnedPremiumAPI`. It has one method, `calcEarnedPremiumByPolicyNumber`, which calculates the earned premium for a given policy number.

See also

- *Integration Guide*

New plugin interface IFXRatePlugin

The new plugin interface `IFXRatePlugin` handles exchange rate conversion. There is a demo implementation in the default configuration for development. For production use, you must implement your own version of the plugin interface.

See also

- *Configuration Guide*

High availability for free-text search with Apache zookeeper

In version 7.0, PolicyCenter supported high availability using a built-in clustering system.

In version 8.0, that system was replaced with a system that uses the Apache Zookeeper project. PolicyCenter does not include the software for Apache Zookeeper, but it is provided for free at the location:

```
https://zookeeper.apache.org/
```

For Zookeeper configuration, consult the Zookeeper documentation.

To configure PolicyCenter free-text search for use with Zookeeper requires modifications to the following files:

- `solrserver-config.xml` – Configures PolicyCenter to connect with SolrCloud servers in a ZooKeeper ensemble
- `zoo.cfg` – Configures each SolrCloud server for membership in a ZooKeeper ensemble
- `myID` – Configures each SolrCloud server with its ordinal number in the ensemble

See also

- *Configuration Guide*

Changed in integration 8.0.0

RPCE web services deprecated

PolicyCenter has two ways of publishing web services:

WS-I web services	WS-I web services are published by using the SOAP protocol and are compatible with the WS-I standard.
RPC Encoded (RPCE) web services	RPCE web services are an older style of publishing web services. Prior to version 7.0, PolicyCenter supported RPCE web service publishing only.

In version 7.0, Guidewire recommended you publish and consume web services as WS-I web services rather than RPCE. Version 7.0 continued to support RPCE web services, and RPCE web services in the default configuration were not deprecated.

In version 8.0, RPCE web services support and RPCE web services in the default configuration are deprecated. Version 8.0 continues to support RPCE web services for backward compatibility only. Guidewire strongly recommends you begin converting your RPCE code to equivalent WS-I code.

As with all WS-I web services, use `Ctrl+N` to find one or more implementation classes in Studio. If more than one is available, check the package name. For example, if the package contains `pc/800` or `pl/800`, that version was introduced in PolicyCenter 8.0.

- For information on writing code for WS-I web services, see the *Integration Guide*.

WS-I web services that replace deprecated RPCE web services

RPCE web services in the base configuration are deprecated in version 8.0. Corresponding WS-I versions replace them. The replacements use different types for some method arguments and return values. For example, arguments and return values that took entity types in RPCE take corresponding types based on Gosu classes in WS-I. Replace your existing code that interacts with RPCE web services to interact with their WS-I replacements instead.

In some cases, WS-I web services replacements are named slightly differently than their corresponding RPCE web services. For example, WS-I names drop the initial letters "I" from their corresponding RPCE names. In addition, WS-I web services are in different packages than their corresponding RPCE web services, and WS-I package names include the version number of PolicyCenter.

Import tools web service

The WS-I web service `ImportToolsAPI` replaces the following deprecated RPCE web services:

- `IImportToolsAPI`
- `IUserAPI`
- `IGroupAPI`

Replacement and new methods for IImportToolsAPI

The `ImportToolsAPI` web service has replacements for the methods on `IImportToolsAPI`, as well as the following new methods:

- `importXml` – Imports only administrative data from XML data that conforms to an archiving import XSD or `topc_import.xsd`
- `importArchiveXmlData` – Imports only administrative data from XML data in UTF-8 encoding that conforms to an XML format for archiving data
- `importArchiveXmlDataAsByteArray` – Imports only administrative data from XML data in a byte array that conforms to an XML format for archiving data

Replacement methods for IUserAPI and IGroupAPI

Use the `ImportToolsAPI` instead of `IUserAPI` and `IGroupAPI` to add a user or group with methods that import XML or CSV formatted text. For other methods in `IUserAPI` and `IGroupAPI`, there are no equivalent WS-I replacements. Guidewire recommends that you write your own WS-I web service to transfer only the necessary user or group data for each integration point.

See also

- For more information on the Import Tools web service, see the *Integration Guide*.

Login web service

The WS-I web service `LoginAPI` replaces the deprecated RPCE web service `ILoginAPI`. However, the WS-I version of `LoginAPI` differs notably from the RPCE version it replaces.

The deprecated RPCE web service `ILoginAPI` plays a critical role in RPCE for server authentication for subsequent method calls to the web service. If you use the generated Java libraries and its `APILocator` utility class, `APILocator` indirectly uses `ILoginAPI`.

The WS-I web service `LoginAPI` plays no analogous role in WS-I for server authentication. Authentication happens effectively with each WS-I method call. PolicyCenter provides the `LoginAPI` web service to let you test your web service client explicitly for specific authentication credentials. Call the `login` method with a username and a password as type `String`. If authentication fails, the `login` method throws an exception.

Note: You can also use the `LoginAPI` to intentionally leave a user session open for logging purposes.

See also

- *Integration Guide*

Maintenance tools web service

The WS-I web service `MaintenanceToolsAPI` replaces the deprecated RPCE web service `IMaintenanceToolsAPI`. In addition to the change from RPCE to WS-I, some methods were added, changed, or removed.

Note: The `MaintenanceToolsAPI` web service is available only if the system is at the `maintenance` run level or higher.

New methods

In Version 8.0, the following methods on the `MaintenanceToolsAPI` web service replace methods on the deprecated `IMaintenanceToolsAPI` web service:

New methods

The `MaintenanceToolsAPI` web services introduces the following methods related to archiving:

- `startValidateArchiveLinksBatchProcessAll`
- `startValidateArchiveLinksBatchProcessByDateRange`
- `startValidateArchiveLinksBatchProcessByIds`

The `MaintenanceToolsAPI` web service introduces the `stopWorkQueueWorkers` method to stop query workers.

Changed methods

- `getWQueueStatus` – Replaces the deprecated method `getWorkQueueStatus`

Removed methods

- `terminateBatchProcessByName` – Use `requestTerminationBatchProcessByName` instead.
- `terminateBatchProcessByID` – Use `requestTerminationOfBatchProcessByID` instead.

See also

- *Integration Guide*

Messaging tools web service

The WS-I web service `MessagingToolsAPI` replaces the deprecated RPCE web service `IMessagingToolsAPI`.

In Version 8.0, the WS-I web service `MessagingToolsAPI` has the following new methods not previously available with the deprecated `IMessagingToolsAPI` web service.

- `configureDestination` – New method to configure some performance and error handling aspects of a messaging destination on a running server.
- `getConfiguration` – New method to get performance and error handling parameters from a messaging destination on a running server.

Tools related to messaging, such as suspending a destination.

There is also a new `messaging_tools` command line tool with similar functionality. Use the `-config` parameter to get the configuration for a destination. Use the `-restart` parameter to restart the messaging destination with new settings. If you use `-restart`, add additional options: `wait`, `retries`, `initial`, `backoff`, `poll`, `threads`, `chunk`.

See also

- *Integration Guide*
- *System Administration Guide*

Policy search web service for claim system integration

The WS-I web service `CCPolicySearchIntegration` web service in the package
`gw.webservice.pc.pc800.ccintegration` replaces the deprecated RPCE web service of the same name. In
addition to the change from RPCE to WS-I, the `retrievePolicy` method changed to return an object of type
`Envelope` instead of type `CCPolicy`. The type `Envelope` is an example of an XSD type that encapsulates a type from
within PolicyCenter for exchange through web services with claim systems.

See also

- *Integration Guide*

Producer web service

The WS-I web service `ProducerAPI` web service in the package `gw.webservice.pc.pc800.community` replaces the
deprecated RPCE web service of the same name. In addition to the change from RPCE to WS-I, some methods were
added, changed, or removed.

New methods

- `createOrganization` – New method to create an organization
- `getGroupByPublicId` – New method to get a group by public ID
- `addNewGroupToOrganization` – New method to add a new group to an organization
- `updateGroup` – New method to update a group
- `getProducerCodeByPublicId` – New method to get producer code by public ID
- `addNewProducerCodeToOrganization` – New method to add a new producer code to an organization
- `updateProducerCode` – New method to update and commit a producer code based on producer code XML
- `updateOrganization` – New method that replaces deprecated methods `synchronizeAgency`,
  `synchronizeBranch`, and `synchronizeProducerCode`

Changed methods

- `getAgencyOrganization` – Method name changed to `getOrganizationByPublicId`.
- Methods that used the `Organization` entity type now use the `OrganizationDTO` Gosu type, which is a class in
  the package `gw.webservice.pc.pcVERSION.community.datamodel`.

Removed methods

- `getBranch`
- `moveProducerCodeToExistingAgency`
- `moveProducerCodeToSyncedAgency`
- `moveProducerCodeToExistingBranch`
- `moveProducerCodeToSyncedBranch`
- `removeProducerCodeFromExistingAgency`

See also

- *Integration Guide*

Profiling web service

The WS-I web service `ProfilerAPI` web service replaces the deprecated RPCE web service `IProfilerAPI`. In
addition to the change from RPCE to WS-I, the following method signatures change:
- `setEnableProfilerForBatchProcess` – The method argument of type `BatchProcessType` now is `String`.
- `setEnableProfilerForWorkQueue` – The method argument of type `BatchProcessType` now is `String`.

See also

- *Integration Guide*

Systems tools web service

The WS-I web service `SystemToolsAPI` web service replaces the deprecated RPCE web service `ISystemToolsAPI`. In addition to the change from RPCE to WS-I, some methods were added or changed.

New methods

- `getDBCCState` – New method that returns the state of the consistency checks process
- `submitDBCCBatchJob` – New method that submits the consistency checks batch job on the underlying physical database
- `submitUpdateStats` – New method that runs the update stats process on the underlying physical database. Afterwards, use the `isUpdateStatsRunning` method to check on the process status.
- `cancelUpdateStats` – New method that cancels the update stats process if running. Afterwards, use the `getUpdateStatsState` method to check on the process status.
- `getUpdateStatsState` – New method that checks whether the update stats process is currently running.
- `submitAwrReportBatchJob` and `submitAwrReportBatchJobAllOpts` – New methods to submit Oracle AWR Performance Report batch jobs
- `getPerfReportsInfo` – New method to retrieve information about the latest Oracle AWR downloads

Changed methods

- `submitDmvReportBatchJobWithParams` – The now takes the single `includeDatabaseStatistics` argument instead of three arguments
- `getRecentAWRSnapshotInfo` – The method now returns `DatabaseSnapshotInfo[]` instead of `String`
- `updateLoggingLevel` – Logging level now is passed as `LoggingLevel` instead of `String`

See also

- *Integration Guide*

Template web service

The WS-I web service `TemplateToolsAPI` web service replaces the deprecated RPCE web service `ITemplateToolsAPI`. In addition to the change from RPCE to WS-I, the methods were added:
- `validateTemplateInLocale` – New method to validate one template using a specific locale code
- `validateAllTemplatesInLocale` – New method to validate all templates using a specific locale code

See also

- *Integration Guide*

Typelist tools web service

The WS-I web service `TypelistToolsAPI` web service replaces the deprecated RPCE web service `ITypelistToolsAPI`.

See also

- *Integration Guide*

Workflow web service

The WS-I web service `WorkflowAPI` web service replaces the deprecated RPCE web service `IWorkflowAPI`. In addition to the change from RPCE to WS-I, the following method signatures change:

- `invokeTrigger` – The method argument of type `WorkflowTriggerKey` now is `String`.
- `isTriggerAvailable` – The method argument of type `WorkflowTriggerKey` now is `String.`

See also

- *Integration Guide*

Existing WS-I web services changes

Some WS-I web services that existed in version 7.0 have new or changed methods in version 8.0.

Account web service changes

The `findAccounts` method of the `AccountAPI` web service takes an XSD type in version 7.0. The method takes an account search criteria from a Gosu class definition of `AccountSearchInfo` in 8.0. See the implementation class for details.

See also

- *Integration Guide*

Claim to policy system notification web service changes

The `ClaimToPolicySystemNotificationAPI` handles notifications from a claims system. For example, the claims system can notify PolicyCenter of a claim amount that exceeds a threshold set for that type of policy. The method signature the `claimExceedsThreshold` method signature changed in version 8.0. The `grossTotalIncurred` parameter was a `String` 7.0. and now is a `MonetaryAmount`.

See also

- *Integration Guide*

Contact web service changes

The `ContactAPI` web service lets an external system manipulate contact data in PolicyCenter. ContactManager calls this web service.

Transaction IDs no longer used

In version 7.0, some methods took a transaction ID, which represented a unique identifier for an update request. If a request is duplicated later, PolicyCenter identifies it as duplicate by its transaction ID and does not repeat the request.

In 8.0, methods no longer take transaction IDs. Instead, they have the annotation `@WsiCheckDuplicateExternalTransaction`, which automatically enforces unique transaction IDs based on SOAP headers. This change affects the following methods:

- `updateContact`
- `removeContact`
- `mergeContacts`

If the web service client is another Guidewire application, you can set the transaction ID with the method `ContactAPIUtil.setTransactionId`.

New methods for address book client API

The `ABClientAPI` published by ContactManager 8.0 has the following new methods to help manage changes to contact information:

- `pendingUpdateApproved`
- `pendingCreateApproved`
- `pendingUpdateRejected`
- `pendingCreateRejected`

These methods are intended as callbacks from the `ContactAPI` web service that the InsuranceSuite applications PolicyCenter, BillingCenter, and ClaimCenter publish for ContactManager to call. However, each application differs in how its implementation of the `ContactAPI` web service uses the methods call back to the `ABClientAPI` that ContactManager publishes.

PolicyCenter does not generate pending changes for ContactManager, so the callback methods are not called in the default configuration. PolicyCenter throws an exception.

See also

- *Integration Guide*

Policy renewal web service changes

PolicyCenter 8.0 changes the `notifyPaymentReceivedForRenewalOffer` method on `PolicyRenewalAPI` to take a `MonetaryAmount` as an argument instead of a `BigDecimal` argument.

See also

- *Integration Guide*

Java code changes

PolicyCenter 8.0 made significant changes to how you write and deploy Java code, including Java implementations of plugin interfaces. The Java API from previous releases is deprecated in version 8.0.

If you deployed Java code in previous releases:

- Your legacy code that uses the deprecated Java API runs with no changes in PolicyCenter 8.0.
- To regenerate the libraries for the legacy API, continue to use the same `gwpc regen-java-api` command as in previous releases. However, there you must add an additional flag on the tool to generate the deprecated Java libraries in addition to the 8.0 Java API libraries. Also, PolicyCenter generates the libraries and Javadoc for the deprecated API in a slightly different location. See "Java library generation in version 8.0" on page 207.
- Although the older API is deprecated, you can continue to use the deprecated API in this release while you start to upgrade your code to the new API.
- You can use Java code from the deprecated API and code using the new API at the same time for different plugin interfaces. Be sure that for each plugin interface and Java class, you follow deploy requirements for each API style (deprecated API or new API).
- The location on disk of a Java class or library determines whether PolicyCenter expects the deprecated Java API or the version 8.0 Java API.
  - For the 8.0 Java API, deploy classes in the locations defined in the *Integration Guide*. Most notably, you must deploy all classes and JAR files in directories that have `/basic/` in the path. Any other locations are unsupported with the 8.0 Java API.
  - For the deprecated Java API, deploy classes and JARs in a plugin directory as defined in previous releases of PolicyCenter. PolicyCenter loads the class with the legacy class loader and does the external entity conversion and legacy conversion of containers (collections and lists). In general, it is incorrect to put any classes or libraries that use the deprecated Java API into any `plugins/`*`PLUGINDIR`*`/basic/lib` folder or `plugins/`*`PLUGINDIR`*`/basic/classes` subdirectories. The *`PLUGINDIR`* value represents the plugin directory as specified for that plugin implementation in the Studio Plugins editor. See the *Configuration Guide*. The one exception is

if you have classes or libraries that do not use Guidewire entity data and previously were in the `plugins/shared/basic` subdirectory.

- ◦ For a complete reference, see "Where to deploy your Java classes and libraries" on page 209.
- ◦ For more about plugin directories, see the *Configuration Guide*.

Overview of Java API changes in PolicyCenter 8.0

There are many differences between the deprecated Java API and the PolicyCenter 8.0 Java API. The most important difference is that there is no longer a proxy layer between Gosu code and your Java code. In the deprecated API, your Java code would be written against a Java API defined solely within the generated entity libraries.

In the 8.0 Java API, the set of supported types is the set of types that are all of the following:

- types written in Java
- types in the `gw.*` packages but not in the unsupported `gw.internal.*` packages

Types in any other packages are unsupported. For example, never use classes in the `com.guidewire` package, which is internal.

Some supported types have some unsupported fields and methods. The unsupported features are marked with the `@gw.lang.InternalAPI` annotation.

> **IMPORTANT** To identify types and features with the `@InternalAPI` annotation in your external Java IDE, see the *Integration Guide*.

The packages for PolicyCenter entity types appear differently in Java than in Gosu. See "Entity packages and customer extensions from Java in version 8.0" on page 207.

Other important aspects of new Java API are different:

- There is no need for special handling of collections across the proxy boundary between Gosu and Java. In previous releases, by default PolicyCenter performed a shallow clone of collections that were function arguments or return values. In more recent versions of PolicyCenter, there were ways of omitting this behavior, but is no longer ever necessary. In the 8.0 Java API, you can use Java collection classes from Java as you would normally.
- There is no need to map additional Java types to access from your Java code.
- There is no mapping of Gosu package namespaces to proxied types. Types created in Gosu are available only through reflection. See "Calling Gosu classes from Java" on page 209.
- Object properties and methods work directly on Java types. For Gosu types, object properties and methods are available only through reflection. See "Calling Gosu classes from Java" on page 209.
- Static properties and methods work directly on Java types. For Gosu types, static properties are available only through reflection. See "Calling Gosu classes from Java" on page 209.
- In previous releases, Gosu enhancement properties and methods appeared directly on the type. In the new API, Gosu enhancements on Java types are available only through reflection. See "Gosu enhancements on Java classes or Gosu classes" on page 209.
- In previous releases, exceptions that PolicyCenter throws are wrapped in a different exception type such as `com.guidewire.external.GenericCheckedException` or `java.lang.RuntimeException`. In the 8.0 Java API, exception handling in Java is straightforward. Just use `try`, `catch`, and `rethrow` as you normally would in Java.
- There is no stripping of parameterization from types in the new Java API. In other words, suppose a plugin interface uses the generics language feature to declare an argument type as `ArrayList<MyClass>`. In the legacy API, it appeared in Java simply as `ArrayList`. In the new Java API, the retains its fully-parameterized type `ArrayList<MyClass>`.

In general, in the new Java API you can use Java classes, interfaces, and other types simply by referencing them directly in Java. However, there are important changes discussed in later topics:

- "Entity packages and customer extensions from Java in version 8.0" on page 207
- "Java library generation in version 8.0" on page 207
- "Getting a reference to a bundle in Java" on page 208
- "Creating entities in the PolicyCenter 8.0 Java API" on page 208
- "Calling Gosu classes from Java" on page 209
- "Gosu enhancements on Java classes or Gosu classes" on page 209

Do not create Java directly in Studio

It is unsupported to add Java class files in the Studio user interface. However, due to technical reasons, PolicyCenter Studio cannot hide user interface options that can add Java classes to the file hierarchies.

> **IMPORTANT** Do not create Java classes directly in Studio. It is unsupported. If you want to code in Java, you must use a separate IDE for Java development. For example, use a separate instance of IntelliJ IDEA or Eclipse just for your Java development.

Avoiding internal APIs

The Java API allows you to use the same Java types that you can use in Gosu. However, Guidewire specifies some methods and fields on these types for internal use only, and unsupported for customer use. In Gosu, these *internal API* methods and fields are hidden, and use of them triggers a compilation error. In Java, when you are using your own IDE separate from Studio, internal APIs are visible even though unsupported.

Guidewire indicates internal API methods and properties with the annotation `@gw.lang.InternalAPI`.

If you use IntelliJ IDEA as your Java IDE, configure the IDE to flag inappropriate uses of internal APIs as warnings. Your copy of PolicyCenter Studio includes a code inspection you can use. See the *Integration Guide*.

Entity packages and customer extensions from Java in version 8.0

Accessing entity and typelist data from Java changed significantly in this release. For complete documentation on entity and typecode data from Java, see the *Integration Guide*.

In Gosu, you can refer to an entity type using the syntax simply `entity.ENTITYNAME` or simply the entity name because the package `entity` is always in scope. In the PolicyCenter Java API, you can reference a type directly by its fully-qualified name. However, for PolicyCenter entity types, from Java the fully-qualified name of an entity is not simply `entity.ENTITYNAME` or simply the entity name. The syntax `entity.ENTITYNAME` or using the entity name with no package is a shortcut within the Gosu language's type system.

Unlike in previous releases, PolicyCenter exposes each entity type as up to three different interfaces. Which interface name to use depends on whether the property is defined in the base configuration, application-specific extensions, customer extensions, or entirely new entities that you define.

If you only want the base configuration properties, the type name is the same in Java as in Gosu, but the package varies by entity type. Some aspects of the fully-qualified names of some interfaces are configurable.

For typecodes, there are differences for how to access a typecode based on where the typecode was defined or extended. This is similar to the differences mentioned earlier for multiple entity interfaces in the 8.0 Java API.

For more information, refer to the following sections:

- *Integration Guide*

> **WARNING** The Java entity interfaces are supported only in Java code. From Gosu, use the syntax `entity.ENTITYNAME`, or, if unambiguous, simply type `ENTITYNAME`.

Java library generation in version 8.0

As in the legacy Java API, in the new Java API there is a library generation step. After you make changes to the data model to add additional extension properties, with both APIs you must regenerate the libraries to use extension properties.

PolicyCenter 8.0 Java API

For the 8.0 Java API, run the `regen-java-api` command from the `PolicyCenter/bin` directory:

```
gwpc regen-java-api
```

PolicyCenter generates the libraries and Javadoc for the 8.0 Java API at the location:

```
PolicyCenter/java-api/lib
PolicyCenter/java-api/doc
```

See the *Integration Guide*.

Deprecated Java API

To use the deprecated Java API, you must add an additional parameter to the `regen-java-api` command:

```
gwpc regen-java-api -Ddeprecated=true
```

PolicyCenter generates separate deprecated libraries and Javadoc in addition to the regular libraries and Javadoc. The output is in a slightly different location than in previous releases:

```
PolicyCenter/java-api/deprecated/lib
PolicyCenter/java-api/deprecated/doc
```

Note that Java code that uses the deprecated Java API must use the plugin directory locations defined in previous versions of PolicyCenter with the deprecated API. For a complete reference, see "Where to deploy your Java classes and libraries" on page 209. For more about plugin directories, see the *Configuration Guide*.

Getting a reference to a bundle in Java

To use entity instances, in many cases you need a reference to a *bundle*. A bundle is a programmatic abstraction that represents one database transaction. See the *Gosu Reference Guide*. The package for the `Bundle` class changed in this release for both Gosu and Java.

To get the current bundle using the PolicyCenter 8.0 Java API, use the following code:

```
gw.pl.persistence.core.Bundle b = gw.transaction.Transaction.getCurrent();
```

If there is no current bundle, you must create a bundle before creating entity instances or updating entity instances that you get from a database query. See the *Integration Guide*.

Creating entities in the PolicyCenter 8.0 Java API

In previous releases, you would create new entity instances in Java using the `EntityFactory` class. The `EntityFactory` class is now deprecated. Only use the `EntityFactory` with the deprecated Java API.

In the PolicyCenter 8.0 Java API, there are two ways to create a new entity instance.

The recommended API for creating an entity instance is to call the `newInstance` method on the entity type's `TYPE` property. Pass a bundle reference as a method argument. For example:

```
// if you do not need customer extension properties....
Address a1 = Address.TYPE.newInstance(b);

 // if you need customer extension properties, use this syntax...
AddressExt a2 = (AddressExt) Address.TYPE.newInstance(b);
```

For a longer code example and alternative APIs for this task, see the *Integration Guide*.

Calling Gosu classes from Java

In the 8.0 Java API, you can call Gosu classes from Java. However, the syntax and API is very different. You must use *reflection* (introspection of the type system at run time) instead of direct access to types and properties at compile time. See the *Integration Guide*.

Gosu enhancements on Java classes or Gosu classes

You can call Gosu enhancements defined on Java classes or on Gosu classes. However, the syntax and API is very different. You must use *reflection* (introspection of the type system at run time) instead of direct access to types and properties at compile time. See the *Integration Guide*.

Exception handling now straightforward

With the legacy Java API, exceptions that PolicyCenter throws would be wrapped in a different exception type such as `com.guidewire.external.GenericCheckedException` or `java.lang.RuntimeException`.

In the 8.0 Java API, exception handling in Java is straightforward. Just use `try`, `catch`, and `rethrow` as you normally would in Java.

Typecode static properties have the TC_ Prefix, just like from Gosu

In the 8.0 Java API, the static properties on a typelist that represent a typecode have the `TC_` prefix, just like from Gosu. However, to actually work with the typecode, you must call the `get` method on the static property to get the appropriate typecode object. For example:

```
LossType.TC_OPEN.get()
```

For additional information about typecodes in the new API, see the *Integration Guide*.

EntityMock Is deprecated

The `EntityMock` API is part of the legacy Java API and is now deprecated. There is no direct replacement for it, but you can write your own mock code to test against your Java code.

Querying for entities by public ID

In the legacy Java API, there was an API to find entity instances by public ID:

```
myClaim = EntityFactory.getInstance().getEntityByRef(Claim.class, "cc:1234")
```

In the 8.0 Java API, just use the standard query builder APIs. See the *Gosu Reference Guide*.

For minor differences in the Gosu and Java APIs for querying, see the *Integration Guide*.

Where to deploy your Java classes and libraries

Implementing Java plugins

If you are writing Java classes that implement PolicyCenter plugin interfaces, those are known as *Java plugins*.

In the following directory listings, `PLUGINDIR_OR_SHARED` represents that the value can be either:

- a plugin directory name as defined in the plugin registry for that plugin
- the special value `shared`. Use `shared` if other Java class loaders or plugin directories need access to those classes or libraries.

For more about plugin directories, see the *Configuration Guide*.

If the code from version 7.0 uses entity data for anything, use the following locations:

- Put classes in `PolicyCenter/modules/configuration/plugins/PLUGINDIR_OR_SHARED/classes`
- Put libraries in `PolicyCenter/modules/configuration/plugins/PLUGINDIR_OR_SHARED/lib`

From the previous two locations, `PolicyCenter` always performs legacy external entity conversion and container (list or collection) conversion.

> **Note:** If your Java plugin code from version 7.0 uses Guidewire entity data, as in 7.0 you must copy the generated file `gw-entity-cc.jar` to the directory `PolicyCenter/modules/configuration/plugins/shared/lib`.

If the code from version 7.0 does not use entity data for anything, use the following locations:

- Put classes in `PolicyCenter/modules/configuration/plugins/shared/basic/classes`
- Put libraries in `PolicyCenter/modules/configuration/plugins/shared/basic/lib`

From the previous two locations, PolicyCenter never performs legacy external entity conversion or container (list or collection) conversion. You cannot use these locations if you use any entity data with the deprecated API.

If the code uses the PolicyCenter 8.0 Java API, use the following locations:

- Put classes in `PolicyCenter/modules/configuration/plugins/`*PLUGINDIR_OR_SHARED*`/basic/classes`
- Put libraries in `PolicyCenter/modules/configuration/plugins/`*PLUGINDIR_OR_SHARED*`/basic/lib`

In the Java API 8.0, external entity conversion and container conversion is unnecessary.

Java classes and libraries to call from arbitrary Gosu code

In version 8.0, as in previous releases, you can deploy Java classes and libraries for use by arbitrary Gosu code in Studio. In other words, in addition to writing plugin code in Java, you can write other Java classes and call them from Gosu.

In the following directory listings, *GOSU_OR_SHARED* represents that the value can be either `Gosu` or `shared`. Carefully note the capitalization of the two words. Use `shared` if other Java class loaders or plugin directories need access to those classes or libraries.

If the code from version 7.0 uses entity data for anything, use the following locations:

- Put classes in `PolicyCenter/modules/configuration/plugins/`*GOSU_OR_SHARED*`/classes`
- Put libraries in `PolicyCenter/modules/configuration/plugins/`*GOSU_OR_SHARED*`/lib`

From the previous two locations, PolicyCenter always performs legacy external entity conversion and container (list or collection) conversion.

If the code from version 7.0 does not use entity data for anything, use the following locations:

- Put classes in `PolicyCenter/modules/configuration/plugins/Gosu/classes`
- Put libraries in `PolicyCenter/modules/configuration/plugins/Gosu/lib`

  From the previous two locations, PolicyCenter always performs legacy external entity conversion and container (list or collection) conversion.

- Put classes in `PolicyCenter/modules/configuration/plugins/shared/basic/classes`
- Put libraries in `PolicyCenter/modules/configuration/plugins/shared/basic/lib`

  From the previous two locations, PolicyCenter never performs legacy external entity conversion or container (list or collection) conversion. You cannot use these locations if you use any entity data with the deprecated API.

If the code uses the PolicyCenter 8.0 Java API, use the following locations:

- Put classes in `PolicyCenter/modules/configuration/plugins/`*GOSU_OR_SHARED*`/basic/classes`
- Put libraries in `PolicyCenter/modules/configuration/plugins/`*GOSU_OR_SHARED*`/basic/lib`

In the Java API 8.0, external entity conversion and container conversion is unnecessary.

Plugin registry changes in Studio

Due to larger changes in PolicyCenter Studio, the user interface changed for registering plugins. In version 8.0, there are several options for registering code that implements a plugin interface, but the deployment options vary:

- **Gosu** – a Gosu class
- **Java** – a Java class.

  PolicyCenter uses the location of classes and libraries on disk to determine whether to use the legacy Java class loader or the PolicyCenter 8.0 Java class loader.

See also

- "Where to deploy your Java classes and libraries" on page 209
- For the 8.0 Java API, see the *Integration Guide*.
- *Configuration Guide*

Document management integration changes

Document Assistant is a Java applet

In previous releases, the Document Assistant was an ActiveX plugin.

In 8.0, the Document Assistant is a Java applet. As such, it no longer strictly requires Microsoft Internet Explorer. The user web client must have Java installed. For the list of supported browsers, Java versions, and operating systems, visit the Guidewire Community search for knowledge article 1005, "Supported Software Components".

It is no longer necessary to make PCF edits to the file `NewTemplateDocumentDV.pcf` when enabling or disabling the Document Assistant.

Client-side PDF document production removed

In previous releases, you could optionally do PDF document production either client-side or server-side. In conjunction with other changes to document production, in version 8.0 the client-side PDF document production is unsupported.

Server-side PDF document production, continues to work unchanged in this release.

Client-side production on Windows for Microsoft Word and Microsoft Excel work unchanged in this release. However, the client-side production is managed by the new Java applet instead of an ActiveX control. In version 8.0, the Document Assistant is a Java applet that encapsulates client-side scripts. On Windows, the client-side scripts are JScript merge scripts for Windows versions of Microsoft Word and Microsoft Excel.

Document content response types removed: HTML and JSCRIPT

In the previous release there were four document content response types. Each content response type is a value in the `ResponseType` property within `DocumentContentsInfo`.

In version 7.0, the response types were:

- Raw document contents as an input stream. Response type is `DOCUMENT_CONTENTS`
- A web page containing an ActiveX control. Response type is `HTML_PAGE`
- JScript code to run on the user's machine. Response type is `JSCRIPT`.
- A URL that can display the content from a local content store. Response type is `URL`.

In version 8.0, HTML and JScript document production types were removed.

In version 8.0, the only response types are:

- Raw document contents as an input stream. Response type is `DOCUMENT_CONTENTS`
- A URL that can display the content from a local content store. Response type is `URL`.

Removal of JScript response type does not affect typical JScript use

Note that although the JScript response type was removed, this change does not affect typical client-side document production using JScript on Windows that used Document Assistant. In version 8.0, the Document Assistant is now a Java applet and encapsulates the client-side scripts such as JScript merge scripts for Windows applications. Similar to previous releases, you can use the `URL` response type to display the Java applet.

Configuration parameter changes

Because of changes to the Document Assistant, the following `config.xml` parameters changed:

- The parameter `AllowActiveX` was removed and replaced with a new parameter `AllowDocumentAssistant`. The value of the old parameter name is ignored.
- The parameter `UseGuidewireActiveXControlToDisplayDocuments` changed names to `UseDocumentAssistantToDisplayDocuments`. The value of the old parameter is used for the new parameter.
- The parameter `AllowActiveXAutoInstall` was removed.
- The parameter `UseDocumentNameAsFileName` was removed.

Minor change in document management Mailmerge attribute

There is a document template descriptor called `mailmergetype`. It optionally configures of pagination of client-side Microsoft Word production. By default, PolicyCenter uses Microsoft Word *catalog pagination*, which correctly trims the extra blank page at the end. However, catalog pagination forbids template substitution in headers and footers. In contrast, *standard pagination* adds a blank page to the end of the file but enables template substitution in headers and footers.

In 7.0, you set the attribute to catalog for catalog pagination and any other value to use standard pagination.

In 8.0, set this attribute to the value `catalog` to use catalog pagination. To use standard pagination, do not set this attribute at all.

If you set this value to some value other than `catalog`, change your document template descriptor to not set this attribute.

Contact messaging flow changes

The messaging flow changed for contact-related messages to external systems. There are different methods on the `ContactSystemPlugin` interface.

See also

- For more information about the new messaging flow for contacts, see the *Integration Guide*.

PCPolicyPublicID Property added to web services methods

BillingCenter billing instructions include a new important property `PCPolicyPublicID`. This property contains the public ID of the policy in PolicyCenter (or other policy system). This property is used in the integration between PolicyCenter and BillingCenter.

Similarly, PolicyCenter web services have the new property. BillingCenter plugins that send information to a policy system now also have the `PCPolicyPublicID` property.

Line-of-business modularization changes to plugins and classes

In version 7.0, the majority of line-of-business code was in rule sets, plugins, and non-line-of-business PCF files and Gosu classes. In 8.0, the majority of that code moved to the `PolicyLineMethods` implementation classes for each line of business. These classes in the `gw.lob` package have names with the LOB prefix with the pattern `LOBPolicyLineMethods`. This topic describes some of the changes made to plugins and Gosu classes described in the *Integration Guide*.

See also

- "Lines of business modularization" on page 169
- *Configuration Guide*

Changes to the policy period plugin and policy term plugin

Previously, the built-in implementation of the policy period plugin (`IPolicyPeriodPlugin`) and policy term plugin (`PolicyTermPlugin`) contained line-of-business-specific code. This code moved to the `PolicyLineMethods`

implementation classes for each line of business, such as `gw.lob.wc.WCPolicyLineMethods`, the relevant class for workers' compensation.

See also

- *Integration Guide*

Changes to rating plugins

Changes were made to the built-in `PCRatingPlugin` and `SysTableRatingPlugin` plugin implementations that implement the `IRatingPlugin` plugin interface. The code that instantiates the rating engine now resides in the `PolicyLineMethods` implementation classes for each line of business. These classes have names with the LOB prefix with the pattern *LOB*`PolicyLineMethods`. See the method `createRatingEngine`.

Changes to policy period diff plugin

The `PolicyPeriodDiffPlugin` plugin used to rely upon line-of-business abbreviations to return *Lob*`DiffTree.xml` files in the function `getDiffTreeConfig(PolicyPeriod)`. With this release, that function is deprecated and Guidewire recommends that you use `getDiffTreeConfig(Product)` in the `gw.product.ProductConfig` class. Additionally `BADiffTree.xml` was removed and replaced with `CADiffTree.xml` for the commercial auto line.

The `PolicyPeriodDiffPlugin` plugin is the built-in implementation of the `IPolicyPeriodDiffPlugin` interface.

Changes to PCBeanMatcher class

In previous releases, the `PCBeanMatcher` class contained line-of-business functionality in the `doBeansMatch(KeyableBean, KeyableBean)` method for the Personal Auto and Commercial Auto lines. This code was specific to the costs on the lines. This line-of-business code has been replaced with a call to a cost's `CostAdapter.isMatchingBean(KeyableBean)`. By default this method returns `false`, however both `PACostAdapter` and `BACostAdapter` override the default return with the prior functionality.

Changes to cost reinsurance enhancement

In the `gw.plugin.reinsurance.CostReinsuranceEnhancement` class, line-of-business code moved from the getter for the `CoverableName` property to the `CostAdapter` classes for each line of business. For example, the `gw.lob.ba.financials.BACostAdapter` class defines a getter for `CoverableName`.

ClaimCenter typelist generator tool changes

In PolicyCenter 8.0, the tool for exporting PolicyCenter product model data for use in ClaimCenter has been moved from Studio to a command-line tool. The functionality is unchanged.

See also

- *Integration Guide*

PolicyCenter plugin changes for multicurrency

To support multicurrency in PolicyCenter 8.0, the following plugin interfaces changed arguments and return values that represent money (for example, `BigDecimal` types) to the new `MonetaryAmount` type.

Plugin interface	Did interface change?	Implementation classes that changed
`IProrationPlugin`	•	`ProrationPlugin`
`IReinsuranceConfigPlugin`	•	`ReinsuranceConfigPlugin`
`IClaimSearchPlugin`		`GWClaimSearchPlugin`
`IReinsuranceCedingPlugin`		`PCReinsuranceCedingPlugin`

Plugin interface	Did interface change?	Implementation classes that changed
IReinsurancePlugin		PCReinsurancePlugin
ILossHistoryPlugin		LossHistoryPlugin

For plugin interfaces that changed, fix any compilation errors in plugin implementations to use `MonetaryAmount` as required by the new versions of the plugin interfaces.

Upgrade free-text search from 7.0 to 8.0

About this task

If you used free-text search with PolicyCenter 7.0, you must upgrade the Guidewire Solr Extension in addition to upgrading your installation of PolicyCenter. The Guidewire Solr Extension that PolicyCenter 8.0 includes is based on Apache Solr 4. The version included with early releases of PolicyCenter 7.0 is based on Apache Solr 3.

The steps for the upgrade from Apache Solr 3 to Apache Solr 4 are as follows:

Procedure

1. Back up the existing index from your current installation of the Guidewire Solr Extension.
2. Uninstall your current version of the Guidewire Solr Extension.
3. Install the Guidewire Solr Extension included with PolicyCenter 8.0, from the distribution file `PolicyCenter/solr/pc-gwsolr.zip`.

   For complete instructions, see the *Installation Guide*.
4. Use the version of `schema.xml` included with PolicyCenter 8.0. This version is in the `PolicyCenter/modules/config/policy/` directory. If you customized or changed a previous version of `schema.xml`, apply the changes to the included version. Do not reuse a previous version of `schema.xml`.
5. Start the Guidewire Solr Extension.
6. In a browser, open the administrative user interface for the Guidewire Solr Extension by entering the following URL:

   ```
   http://hostName:8983/pc-gwsolr
   ```

7. If you will be normalizing phone numbers, do so at this step, before upgrading the index files. If you normalize phone numbers after upgrading the index files, you will have to upgrade the index files and initiate the batch load process again.
8. Initiate batch load process.

Timing for free-text search upgrade

Guidewire recommends that customers plan ahead for additional fields to the Solr Extension index. Add fields to the index before or during a PolicyCenter upgrade. Do not wait until after an upgrade to do this.

The reason for such advance planning is that the addition of fields to the Solr Extension index requires executing the Solr batch load process. The time and resource cost for running batch load is substantial when a PolicyCenter instance has many policies. Add a field to the Solr Extension index as soon as you determine that the field would be useful and feasible for free-text search.

Guidewire recommends that customers time order the steps for an upgrade conceptually as follows:

1. With a long-term perspective, discern which entity fields would be useful and feasible as search fields for the basic search page. Apply this step with respect to any entity fields other than the default free-text search fields.
2. In the new target system, configure free-text search for these additional entity fields. Team members do not need to configure the basic search page for these fields at this moment. They can configure the basic search page after the upgrade.

3. Perform the upgrade from the current PolicyCenter system to the new system.

4. Run the free-text batch load command.

Configuring free-text search fields before upgrading avoids executing the free-text batch load process a second time after the new target database is full of policies. The preceding guidance applies without regard to whether the old source system uses free-text search.

Populator plugin deprecated

In PolicyCenter 8.0, the populator plugin interface (`PopulatorPlugin`) is deprecated. PolicyCenter no longer calls this plugin in the default configuration. If you implemented or modified the built-in implementation plugin, be aware that PolicyCenter does not call it unless you added your own code to call the plugin.

In PolicyCenter 7.0, the web service implementation classes for `PolicyRenewalAPI`, `SubmissionAPI`, and `ImportPolicyAPI` called the populator plugin. In version 8.0, those web services now directly include lines that duplicate the code in the plugin default implementation class `ExamplePopulatorPlugin`.

The `ExamplePopulatorPlugin` plugin implementation first parsed `String` data that contained XML using the Guidewire XML model (GX model) definition. Next, the plugin implementation called the `populatePolicyPeriod` method on the XML object, which is implemented as a Gosu enhancement method.

If you re-implemented the `PopulatorPlugin` plugin interface or modified the `ExamplePopulatorPlugin` code, you now need to modify the behavior of `PolicyRenewalAPI` and `SubmissionAPI` directly.

If you need similar behavior of the plugin for other code contexts, copy the code from the `ExamplePopulatorPlugin.gs` implementation. Optionally, copy the main populator code that is in the `populatePolicyPeriod` enhancement methods on the XML objects. There are two implementations of this enhancement method, corresponding to either the legacy 7.0 version of the GX Model or the newer 8.0 version. For example:

- The number in the package of the web service implementation class `gw.webservice.pc.pc700.SubmissionAPI` indicates the 7.0 version of the web service.

- The 7.0 version of the web service uses the 7.0 version of the GX Model, which is `gw.webservice.pc.pc700.gxmodel.policyperiodmodel`

- The 7.0 version of the GX Model uses the enhancement method at the path `gw.webservice.pc.pc700.gxmodel.PolicyPeriodModelEnhancement`

Change the `pc700` in the packages to `pc800` if you want to use the 8.0 versions.

You might want to add additional logic to the import process, such as populate and validate line-specific data. You can make that change to the `populatePolicyPeriod` method in the `PolicyPeriodModelEnhancement` file.

Territory code plugin deprecated

The territory code plugin was deprecated. If you need to customize the logic of lookup of territory codes, in Studio edit the Gosu class `gw.lob.common.TerritoryLookupCriteria`. That class has a `search` method that takes the object instance fields and generates a database query based on the fields. The database query returns a set of `Territory` objects that satisfy the criteria.

PolicyCenter product model import into ClaimCenter changes

PolicyCenter 8.0 removes the Product Model editor from Studio and replaces it with a separate web application included with PolicyCenter 8.0. The web application, Product Designer, does not include a menu option to run the ClaimCenter Typelist Generator. Instead, you run the ClaimCenter Typelist Generator from the command line.

See also

- *Integration Guide*

Release Notes Archive

This section contains the release notes for previous versions of PolicyCenter. Use these files to learn what features changed from one release to another.

IMPORTANT This section contains upgrade information originally provided for earlier PolicyCenter releases. It may be superseded by later release notes or other upgrade documentation.

Guidewire PolicyCenter 9.0.5 Release Notes

©2001-2018 Guidewire Software, Inc.

For information about Guidewire trademarks, visit `http://guidewire.com/legal-notices`.

Guidewire Proprietary & Confidential — DO NOT DISTRIBUTE

These release notes contain the following topics:

- "Release information" on page 219
- "Installing this release" on page 220
- "Support" on page 220
- "Major issues and changes" on page 221
- "Improvements and resolved issues" on page 221
- "Known issues and limitations" on page 228

Release information

These release notes apply only to this release of Guidewire PolicyCenter.

> **IMPORTANT** If you skipped upgrading to one or more of the previous releases of PolicyCenter, be sure to read the *Release Notes* for those releases to learn about related changes and fixed issues.

Release number

This release of Guidewire PolicyCenter is 9.0.5.

- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 8.0.1 or later. The latest maintenance release is preferred.
- If you are implementing multiple settlement currencies, BillingCenter 8.0.2 or later is preferred.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 8.0.0 or later. The latest maintenance release is preferred.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 8.0.1 or later. The latest maintenance release is preferred.

Installing this release

Refer to the following:

- The *PolicyCenter Installation Guide* for general installation information
- Prior PolicyCenter release notes for any releases that you have skipped

Upgrade information

IMPORTANT The procedure for upgrading to PolicyCenter 9 from PolicyCenter 8 or previous releases has changed significantly. In PolicyCenter 9, there are new tools available for performing the upgrade. Do not rely on your past knowledge of the upgrade process to begin the upgrade procedure. Before beginning your upgrade, review the topic "Configuration Upgrade Overview" in the *PolicyCenter Upgrade Guide*.

Upgrade to PolicyCenter after running ETL Product Model Loader extension pack (DOC-10285)

Issue

Upgrading to PolicyCenter 8.0.3 or later after running ETL Product Model Loader extension pack may require a manual drop of some tables.

In PolicyCenter 8.0.3, the ETL Product Model Loader Extension Pack functionality moved into the core PolicyCenter product. The extension pack created ETL database tables with `pcx_` as the prefix. PolicyCenter 8.0.3 and later create ETL database tables with `pc_` as the prefix. If you ran this extension pack with a prior version of PolicyCenter, you will have stale tables after upgrading to PolicyCenter 8.0.3 or later. The upgrade does not fail, but the server detects extra tables in the database.

Workaround

To remove these stale tables, manually drop the old `pcx_` tables created by the ETL Product Model Loader extension pack. The ETL Product Model Loader Extension Pack created these tables:

- `pcx_etlclausepattern`
- `pcx_etlcovtermpattern`
- `pcx_etlcovtermoption`
- `pcx_etlcovtermpackage`
- `pcx_etlpackterm`
- `pcx_etlmodifierpattern`
- `pcx_etlratefactorpattern`

You may also need to update your ETL code to reference the new tables.

Then, when upgrading, overwrite or merge ETL product model loader files in your configuration.

Exception when upgrading UWIssue objects in sample data from 7.0.x (PC-34046)

Issue

When upgrading from PolicyCenter 7.0.x, database upgrade throws an `IllegalArgumentException` exception when encountering `UWIssue` objects in the sample data. The exception occurs because the upgrade expects to find typelist values added in PolicyCenter 8.0.x.

Workaround

To workaround this issue, delete `UWIssue` objects in the sample data from the PolicyCenter 7.0.x database or add the typelist values necessary.

Support

For assistance, visit the Guidewire Community.

Guidewire Customers

- https://community.guidewire.com

Guidewire Partners

- https://partner.guidewire.com

Major issues and changes

This section contains issues and major changes that may affect your installation.

For information on new features and major changes, see the topic "New and Changed in PolicyCenter 9.0.5" in the *PolicyCenter New and Changed Guide*.

Base PCF file changes

The link that follows requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 9.0.4 to 9.0.5

To view a report of the changes to the base PCF files, `click here`.

Base resource changes

The link that follows requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 9.0.4 to 9.0.5

There are no changes to the base resources.

Changes in this release provided in Upgrade Diff report

Guidewire provides a report detailing certain differences between the current release and your prior release. This report describes changes in display keys, entities, typelists, and the Gosu API. To obtain your custom Upgrade Diff report, visit the Guidewire Community and search for knowledge article 7898, "How to find reports detailing differences between earlier and later releases of Guidewire core products".

Improvements and resolved issues

This topic describes improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This list is not intended to be comprehensive.

PolicyCenter improvements and resolved issues

ID	Description
PC-16242	Fixed an issue that caused a database consistency check failure in the `pc_groupuser_assign` table after transferring a user to a group from an external organization and deleting their original group assignments.
PC-32712	Fixed an issue on the **Payment** screen where the number of installments was inconsistent with the number of installments returned by BillingCenter. Also fixed the installment amount that was showing the max of all installments instead of the periodic installment amount.
PC-34039	Improved performance of code that handles rate routines. This was particularly noticeable when loading rate books on system startup.

ID	Description
PC-34091	Fixed an issue which caused a database consistency check warning on `FormAssociation` entity instances when adding and removing a form that could occur multiple times on the same coverable (form multiplicity).
PC-34524	Fixed issues with preloading rate tables locking up the server.
PC-34654	The `WorksheetCodeExecutionHelper#postExecute` method was not null safe and could cause a `NullPointerException` causing rating to fail. With this fix, `WorksheetCodeExecutionHelper#postExecute` is now null safe.
PC-34814	Prior to this change, each PolicyCenter node starting up would acquire an exclusive database lock and then validate and update the product model. In some deployment environments, this led to long overall cluster startup times. To resolve this issue, PolicyCenter added a product model configuration fingerprint, which it stores in the database. If the product model configuration has changed, the first node starting up validates the configuration and then makes necessary updates to the database tables. After this process is complete, the first node updates the configuration fingerprint in the database. Subsequent nodes will only acquire the exclusive database lock for a very short time to check that the configuration fingerprint is up to date. If it is, PolicyCenter skips all product model validation and updates on that node.
PC-34860	Added an index on the `CovTermOptLookup` table to improve the performance of product model verification lookup on startup.
PC-35117	Fixed a previously known issue where policy locations were copied incorrectly when rewriting to a new account. This occurred in policies with multiple locations having the same address.
PC-35198	In PolicyCenter 9.0.0, the configuration upgrade tools changed, and the `gwb` command to upgrade the product model, `gwb upgradeProductModelStructure`, was no longer used during the upgrade process. In PolicyCenter 9.0.1, this functionality was completely disabled. This command has now been removed. See the Configuration Upgrade Guide for more information on upgrading the product model.
PC-35329	For improved performance, PolicyCenter now caches `ClosingOrClosedPolicyPeriodStatuses` for the duration of the quote process.
PC-35390	Fixed an issue where viewing, exporting, or validating a rate book resulted in `IllegalStateException: More than one element matches the given condition`. This occurred if the rate book contained a rate routine that had more than one operand of Rate Table type in a rate routine step. For example, this occurred in an IF expression that compared two rate tables.
PC-35396	There was an issue with a user viewing a submission transaction, and then data destruction purging the associated policy, and then the user signing in again. PolicyCenter threw an exception in the log file when the user signed in again after data destruction had purged the policy. This issue has been fixed, and now PolicyCenter displays **???** in place of the transaction number in the **Policy** drop-down menu.
PC-35400	Fixed a previously known issue where data destruction caused the database to generate a reference constraint error when it purged a merged account. This occurred when Reinsurance Management was enabled.
PC-35401	Fixed a previously known issue where PolicyCenter threw a runtime exception "Mismatch between Pinnables in instance graph vs. Pinnable hierarchy" when it attempted to purge an account that had a policy but the policy has been moved to another account. This is part of the data destruction feature.
PC-35402	Fixed an issue which resulted in `IllegalArgumentException: Comparison method violates its general contract!` when exporting a rate book to XML.
PC-35412	Fixed an issue in a renewal which disabled the **Release Lock**, **Quote**, and other buttons on the **View Quote** screen after editing the policy transaction for a second time.
PC-35420	Improved performance of the **Policy Review** step in the Submission wizard by removing unnecessary diff calculations.
PC-35425	Improved performance of `ContingencyEnhancement#QueryForActivities`. This method used to do a full table scan on `Activity` and do the filtering in memory. Now filtering is done in the database.
PC-35430	Instead of throwing an exception, `RangeParam#equals` now returns `false` when comparing values of different types, such as `Integer` and `BigDecimal`.

ID	Description
PC-35452	With this fix, domain graphs are always initialized, but domain graph issues only block server startup if `DisableDomainGraphSupport` is set to `false`.
PC-35454	Fixed an issue in rate routines where the decimal character was localized in computations, causing them to fail.
PC-35458	PolicyCenter now finds and merges duplicate entities for jobs that are not out-of-sequence.
PC-35459	Fixed an issue which caused an exception after re-adding removed coverages then editing the effective date of a policy change.
PC-35466	PolicyCenter will not start if the `config.xml` parameters `ArchiveEnabled` and `DisableDomainGraphSupport` are both `true`, or if `DisableDomainGraphSupport` and `PersonalDataDestructionEnabled` are both `true`.
PC-35477	Fixed an issue where Rating Management sometimes returned incorrect rate factors in rate tables with interpolated parameters. This occurred in large rate tables where the value of `RowID`, an internal Oracle column, could change.
PC-35486	On a production system, you cannot import the XML from an exported rate book using **Utilities→Import Data** because the XML is not fully validated. The updated error message advises the user to use **Rating→Rate Books→Import from XML**.
PC-35487	Fixed an issue where a coverage term did not properly inherit its associated policy term reference type. When the coverage term's reference date was inferred, the coverage term became unavailable.
PC-35510	In `SubmissionEnhancement.gsx`, when cloning the policy period, the underlying implementation of `period.copyBranchIntoNewPolicy` has changed. The `copyBranchIntoNewPolicy` method now preserves the based-on link in entities in the new policy period instead of removing it.
PC-35513	Fixed a performance issue with getting the producer code for a user. The issue occurred in code called by the `ensureProducerIsValid` method in `gw.policy.PolicyPeriodBaseEnhancement.gsx`.
PC-35523	There were a few places where anonymous implementations of `Comparator` were being used to sort objects that implement the `EffDated` interface. These have been changed to use the `EffDatedInternalMethods.EFFECTIVE_DATE_COMPARATOR`. There were also a few places where anonymous implementations of `Comparator` were being used to sort instances of the `RIRisk` class by effective date. They have been changed to use the `RIRiskVersionListInternalMethods.EFFECTIVE_DATE_COMPARATOR`. The `RIRiskVersionListInternalMethods.EFFECTIVE_DATE_COMPARATOR` was only comparing on effective date. This meant that if two instances of `RIRisk` had the same effective date, then their order was non-deterministic. To fix this, this comparator will now compare effective date, expiration date, and finally ID should the previous comparisons be equal.
PC-35527	Fixed an issue in rolling upgrade in which duplicate Public IDs for two different product model pattern types caused the rolling upgrade to fail. The issue was that Guidewire guarantees that public IDs are unique within a pattern type, but, not across different pattern types. This situation could possible cause data corruption. Guidewire added a new field for pattern type for product model rolling update activation. Guidewire also automatically populates this field while performing a full upgrade as well.
PC-35548	For performance reasons the Policy Number version trigger, which runs when upgrading from PolicyCenter 8.0.0 or earlier PolicyCenter releases, now runs at the beginning of a database upgrade with the other database version triggers. This database version trigger updates `PolicyPeriod` to make the `PolicyNumber` column nullable. If `PolicyNumber` equals `Unassigned`, the trigger sets `PolicyNumber` to `NULL`.
PC-35555	To improve the performance of the `GenericClauseSelectionForm` class, added method `FormInferenceContext#getCoverablesForClausePattern(ClausePattern)`. If you modified the `GenericClauseSelectionForm` or `FormInferenceContext` class, consider merging these changes into your implementation..
PC-35580	Improved performance of rate book promotion by removing unnecessary validation checks on included objects such as rate table definitions and rate routine definitions.
PC-35597	The `SideBySideProcess.ParallelQuoteCallable#call` method is now thread-safe.

ID	Description
PC-35618	Fixed an issue where some export work items could not be processed during export of all rate books to XML. The export status for these rate books stopped at "Export in progress" with the error "commitBundle must not be called in a transaction".
PC-35628	Basic policy search, Solr-based free-text search, used to throw a null pointer exception when entering alphabetic characters in the phone field. With this fix, PolicyCenter displays an error message.
PC-35643	Improved database upgrade performance by making sure an expensive SQL query is executed only after version trigger checks determine that it is necessary.
PC-35649	Diff display now uses property name display keys when available for `Clause` entity properties that do not correspond to coverage terms.
PC-35655	Fixed an issue in out-of-sequence transactions where reverting a value back to its original value did not cascade forward in time. For example, if the original color of a car was blue, changed to green, then back to blue, the last change did not revert back to blue in future slices.
PC-35667	Improved performance of importing rate books by removing unnecessary rate routine validations.
PC-35695	Fixed an issue which caused a null pointer exception when trying to communicate with ContactManager. This occurred if the `ContactMessageTransport` was suspended, and there were updates to a contact bean, but then the contact bean was removed. With this fix, the bean still is not sent to ContactManager, but PolicyCenter does not throw a null pointer exception.
PC-35770	Cascaded lookup in rate books, added in a previous release, enables jurisdictions to inherit rate tables from a countrywide rate book. This feature made it difficult to provide the Missing Rate Table warning accurately and with acceptable performance. To avoid reporting false warnings and potentially missing real warnings, this warning has been removed.
PC-35782	If rate table was too large for normalization, the previous behavior was to display an error message without letting the user proceed. Now users can import rate tables regardless of whether normalization succeeds or not. PolicyCenter displays a warning message letting the user know that normalization failed in the background.
PC-35827	To fine tune performance, use the `ProductModelClassCacheConcurrencyLevel` configuration parameter to adjust the concurrency level of the product model class cache. This cache is lazy-loaded when product model pattern objects, such as coverage patterns, are instantiated. See `com.google.common.cache.CacheBuilder#concurrencyLevel` for more information about caches and concurrency levels.
PC-35838	Previously, Policy Renewal Start batch processing attempted to begin renewal processing without checking to see if the policy in question could be renewed. This could result in exceptions being thrown in the worker thread. This now checks whether the policy can be renewed before attempting to initiate the renewal process.
PC-35858	`QuoteProcess` now commits to the database before performing out-of-sequence actions on the policy transaction.
PC-35810	Fixed an issue which caused an exception when editing a rate routine and adding a parameter set that applies to all lines.

Platform improvements and resolved issues

ID	Description
PL-20685	Fixed an issue where a web service call resulted in a `java.lang.LinkageError`. The issue was resolved by removing the JRE `activation.jar` and `saaj.jar` files from the Guidewire-packaged WAR files.
PL-30292	Fixed an issue that caused the **Not Started** column in the **Consistency Check Runs** report to always show 0.
PL-32345	Modified the on-screen labels and added an information message to improve the usability of the Consistency Check popup.
PL-35419	Fixed a message queue issue that occurred after losing and restoring a database connection. The resumed message queue status was shown as `Started`, but the queue did not process messages.

ID	Description
PL-35603	The unwanted test modules that were previously being packaged into the build are removed.
PL-35658	Guidewire has deprecated the `TableImportAPI` methods that perform work synchronously. Guidewire recommends that you use the version of the method that performs work asynchronously instead. In some cases, Guidewire added the asynchronous version of the method if it did not exist. For the asynchronous methods, use the `ProcessID` and poll using the Maintenance Tools API to determine if the work is complete.
PL-36179	Application logging during upgrade now includes upgrade information for troubleshooting purposes.
PL-36280	Fixed an issue where upgrade failed for some databases using a non-English language with following error: "The conversion of a varchar data type to a datetime data type resulted in an out-of-range value".
PL-36504	Fixed an issue that permitted the successful execution of staging table import at the `DAEMONS` server run level, which Guidewire does not support. Now, it is not possible to execute a staging table import in a server run level that is higher than `maintenance (NODAEMONS)`.
PL-36547	The `BatchProcessBase#incrementOperationsFailedReasons` method is now null-safe.
PL-36577	Fixed an issue that caused a large number of `Xml.Error`-type messages when using a localized version of the `xmlcodegen_en.properties` file.
PL-36744	Fixed an issue where a SOAP MTOM (Message Transmission Optimization Mechanism) response that contained multiple sequential `CR-LF` pairs would throw an exception.
PL-36754	Corrected an issue that caused the Load History download to show incorrect or missing information.
PL-36924	In previous releases, a full upgrade of configuration changes only on a single, standalone PolicyCenter server would complete successfully. However, in that case, PolicyCenter did not update the Server Tools **Upgrade and Versions** screen with upgrade information. Guidewire has changed this behavior so that a full upgrade of a standalone server, even if only for configuration changes, now updates the upgrade information on the **Upgrade and Versions** screen.
PL-36961	Modified handling of LOB objects so as to make better use of Oracle temporary tablespace.
PL-37004	Made performance improvements to the database upgrade step that populates entity search denorm fields during the application upgrade.
PL-37029	Obfuscating a `User` entity by using `PersonalDataDestructionAPI` no longer causes the DBCC tool to report that the `UserSetting` object related to that `User` is missing.
PL-37034	Fixed an issue in which Oracle parallelism was not enabled for loader callbacks during the execution of the database `integrityCheckAndLoad` process. Also, updated Javadoc for DML methods for Loader callbacks to remind developers to use the methods in the class hierarchy that respect the parallel setting in the `database-config` `<loader>` and `<callback>` elements.
PL-37048	Removed the obsolete `schemalocations.xml` file, which was replaced with `gwxmlmodule.xml` in version 9.0.0.
PL-37057	Fixed an issue where running the command `gwb genAllWsdl` failed with a `ConcurrentModificationException`.
PL-37072	Fixed an issue that would cause a server error when a startable plugin or external process called `getRunLevel` during server startup.
PL-37122	Addressed security issues by upgrading several third-party libraries.
PL-37135	Fixed an issue in JMS inbound integration with the `ordered` parameter set to `false` where polling an empty thread did not close the JMS session.
PL-37146	Any change to the `useoraclestatspreferences` attribute in `database-config.xml` (from `false` to `true` or from `true` to `false`) takes effect only after an upgrade, either a full upgrade or a rolling (configuration) upgrade. However, if you reset this attribute from `true` to `false`, PolicyCenter throws an exception during the next upgrade and prevents the upgrade from continuing due to locked table statistics in the Oracle database. Review the details of the exception provided in the server log to determine which table statistics need to be unlocked. See "Revert to DBStats Batch Processing for Database Statistics" in the *System Administration Guide* for information on how to unlock the table statistics.

ID	Description
PL-37153	Fixed an issue in which the `DropForeignKeysToProcessHistoryAndDeleteProcessHistoryRecordsTrigger` upgrade trigger deleted `ProcessHistory` data.
PL-37178	Guidewire has updated the ping utility, an unauthenticated web page that you can ping to access information about a PolicyCenter server. See the Guidewire documentation for the meaning of the fields that the ping utility returns.
PL-37186	Added a new public method to `gw.api.profiler.Profiler`: `public static <V> Callable<V> createPotentiallyProfiledCallable(ProfilerTag entryPointTag, String entryPointDetail, Callable<V> block)`
PL-37187	Fixed an issue related to Tomcat version 8.0.37 and higher for which it was possible to see a number of cache-related warning messages in the logs.
PL-37247	Fixed an issue that could cause an exception if bundle contents changed during iteration of the beans in the bundle. The `getBeansByRootType` method now returns a collection that is an immutable copy of the bean values in the bundle.
PL-37256	Previously, during database upgrade, table aliases had a numeric suffix that was incremented globally. As a result, depending on the starting point, two runs of the database upgrade could result in two different strings for the same query. This behavior has been updated so that table aliases are now consistent—running database upgrade on the same initial database does not result in two different strings for the same query.
PL-37263	You can now update the database statistics settings from file `database-config` with a rolling upgrade. This change can also include using `database-config` to switch to using Oracle statistics preferences.
PL-37266	Eliminated a security risk. Gosu servlets no longer accept a session ID if the HTTP connection is non-secure. For network topologies that secure the connection through other means, such as by not allowing external access to the server, the security check can be overridden by setting the system property `gw.servlet.ServletUtils.BypassIsSecure` to `true`.
PL-37284	Fixed an issue in JMS inbound integration that prevented recovery of the message queue if the connection to the application server failed.
PL-37292	Fixed an issue that caused the file inbound integration to stop processing files if the incoming folder contained more than 300 files.
PL-37300	The default configuration's `JavaxEmailMessageTransport` plugin processes an HTML or plain text email as intended. In the plugin's `addBody` method, the email is sent as HTML or plain text based only on the value of the `email` argument's `Html` property, which has a default value of `false`. In the default configuration, the content of the email is of unknown origin. Scanning the content to determine whether it is HTML introduces a security risk. Specifically, the default configuration cannot determine whether the content includes attack-injected HTML. Configuration code that creates the email content itself, and therefore knows the content is secure, can explicitly set the `Html` property to `true` to have the email sent as HTML.
PL-37306	The database schema verifier no longer checks if the table name exceeds the Guidewire maximum table name length. This check occurs in the entity generator/verifier.
PL-37314	Fixed an issue in which duplicate IDs within a PCF could cause a `StackOverflowError` error while executing the `genDataDictionary` command.
PL-37315	Commented out an unnecessary `print` statement in `GwXmlElementEnhancement.gsx`.
PL-37322	Fixed an issue in JMS inbound integration that prevented the message queue from restarting if the destination shut down and restarted.
PL-37324	Improved the performance of foreign key constraint creation during upgrade.
PL-37363	Fixed an issue that caused PolicyCenter to throw a `NullPointerException` exception if method `EmailContact.setContact(contact : Contact)` had a null `Contact` as a parameter.

ID	Description
PL-37365	Changed the default value for `ora-parallel-dml` in the upgrade element in file `database-config` from enable to `enable-all`. This change means, in Oracle, that INSERT, UPDATE, and DELETE statements in version triggers run in parallel by default.
PL-37367	You can now configure the maximum threads per path for communication with Guidewire Solr Extension. Increase the maximum threads per path to improve throughput and concurrent operation.
PL-37369	Fixed an issue that generated an exception in the server log if running `DBStats` or `DBConsistencyCheck` batch processing with no arguments from the command prompt.
PL-37384	Fixed an issue where database upgrade steps were triggered with `versionchecksonly=true`. The `versionchecksonly` setting now works as expected during database upgrade.
PL-37394	Improved the reporting of upgrade-related activity in the application log.
PL-37406	Improved information in the server log during an upgrade. For example, the log now states explicitly when the upgrade completed: `Full upgrade completed successfully: schema and configuration changes have been applied`
PL-37429	Fixed an issue that caused a Null Pointer Exception during the execution of an Oracle AWR report.
PL-37447	Addressed security issues by upgrading the `poi-ooxml` and `fop` JAR libraries. A new and empty `FOPPrint.config` file was added to the `config/resources` directory.
PL-37448	Eliminated a security risk by updating the `ant` and `ant-junit` JAR files.
PL-37461	Implemented performance improvement around server startup.
PL-37471	Eliminated a potential denial-of-service security risk by updating the `commons-fileupload` library. The library is used when uploading files during the document management process.
PL-37475	Fixed an issue with a document template not being able to load a dynamic image.
PL-37486	Fixed an issue where a web service client making an external API call would not release its socket when timed out.
PL-37498	Fixed an issue in a rolling upgrade that caused PolicyCenter to not properly mark deferred tasks as completed. This issue prevented archiving from running.
PL-37530	Fixed a concurrency issue with the `gw.xml.ws.AsyncResponse.get` method.
PL-37561	Changed the message-locking behavior for distributed message transactions. When processing a distributed message transaction, the transaction's message object is not locked. This locking behavior can be configured by setting the new `LockDuringDistributedMessageRequestHandling` configuration parameter. The message-locking behavior for non-distributed message transactions remains unchanged. For non-distributed transactions, the message object is always locked. If the message object is not locked, any associated primary entity will not be locked either, even if locking of the entity is enabled. For details, refer to the main documentation.
PL-37592	Fixed an issue that caused the rolling upgrade process to not accept changes to configuration parameters listed as `localOK`, and, instead, forced a full upgrade.
PL-37600	Fixed an issue that caused duplicate database index definitions for an index defined on an entity delegate extension. This issue caused the index definition to fail.
PL-37607	Fixed a performance issue in the Submission wizard and quoting.
PLWEB-6332	Fixed an issue where a `BooleanRadioInput` that toggled the visibility of other screen elements would cause excessive screen flickering.
PLWEB-6334	Fixed an issue in a time field where entering an invalid value would not be validated and the value would be discarded.
PLWEB-6337	Fixed an issue where clicking on a field when an off-screen field had focus would cause the screen to jump to the field with previous focus.

ID	Description
PLWEB-6338	Fixed an issue where selecting multiple rows in two separate list views and then clicking **Remove** would remove the rows from both list views. Instead, only the rows in the list view containing the clicked button are removed.
PLWEB-6629	Fixed an issue where you could not select the title text in a wizard widget.
PLWEB-6630	Pressing the `AltGr` key no longer behaves as if the `Alt` key is pressed.
PLWEB-6631	Fixed an issue where validation messages were not visible when they appeared in a Workspace panel that was hidden. If the Workspace is hidden when new messages appear, the Workspace now opens automatically.
PLWEB-6643	When running the application from the command line, the ability to dynamically reload PCF files is disabled by default. You can restore it by adding the `-DgosuInit.supportDiscretePackages=true` VM option to the command line call. This functionality is retained by default when running the application from Guidewire Studio by the presence of this option in the Studio server configuration.
PLWEB-6649	Fixed an issue where check box changes on newly expanded `RowTree` rows were ignored when submitting.
PLWEB-6651	When specifying a number as a keyboard shortcut, both the main keyboard number key as well as the corresponding key on the number pad now work as that shortcut.
PLWEB-6652	The `required` attribute of a `Cell` can no longer access the iterated value.
PLWEB-6653	Fixed an issue where a `Link` with an `Arg` was not shown.
PLWEB-6658	Fixed an issue where the `wrap` attribute on a `Cell` was not respected by `FormalCell` or `LinkCell`.
PLWEB-6659	Fixed the second of two labels for policy quote validation results. Previously, the two labels for quote validation results were **On Vehicles** or **On Drivers** as well as **Other**. For the first label, the user interface still displays the option to which you did not navigate, either **On Vehicles** or **On Drivers**. The first label has not changed. However, the second label is now **On Current Page**.
PLWEB-6660	Restored attribute `reflectOnBottom` to the PCF widget `Toolbar`.
PLWEB-6670	Added attributes `actionHandler` and `updateHandler` to PCF widget `TemplatePanel`.
PLWEB-6671	Added new PCF widget `PanelIteratorRemoveLink`. The PCF editor in Guidewire Studio does not yet fully support this new widget, so you might need to manually edit the XML of the PCF file to add `PanelIteratorRemoveLink`.
PLWEB-6685	Fixed an issue where collapsing an input group located at the bottom of a scrolled screen caused the view to shift.
PLWEB-6694	Fixed an issue where a list detail view would not show the correct pagination when selecting a new row.
PLWEB-6842	Fixed an issue with unnecessary locking of PCF files, which caused a performance issue in `PCFServiceImpl`.
PLWEB-6851	Fixed an issue with type system locking that impacted the performance of rendering PCF files.

Known issues and limitations

This section describes known issues with this release of Guidewire PolicyCenter.

PolicyCenter known issues

Provide a way to filter out added and removed subentities in a policy differences tree (PC-7929)

Issue

Need a way to filter out the children of an added or removed entity.

Workaround

Guidewire is aware of this issue.

When using parallel sessions, approvals made by underwriter are not updated for second user simultaneously viewing same work order (PC-12322)

Issue

Approvals made by an underwriter are not updated for a second user simultaneously viewing the same work order. This occurs in the following sequence of events:

1. An agent processes a submission and issues arise
2. The agent calls an underwriter, and remains in the policy
3. The underwriter approves the issue or issues
4. The agent refreshes the **Risk Analysis** screen by going to a different wizard step and returning to the **Risk Analysis** screen

After this sequence of events:

1. The new approvals do not appear on the screen of the agent
2. However, the agent can bind or issue the policy (based on the new approvals by the underwriter)

Workaround

The agent can leave the policy and return to it. The approvals appear as expected. Alternatively, the agent can see the approvals by clicking the **Details** button when issuing the policy.

Must enter VIN when moving from quick quote multi-version or side-by-side to full application (PC-13099)

Issue

In a quick quote, you do not have to enter the VIN or other fields that are required for a full application submission. When converting a quick quote with multi-version or side-by-side quoting to full application, to resolve validation issues, you must enter the VIN for each of the versions. In a job with side-by-side quoting, this can occur with fields that are not base data.

Workaround

Add code that copies this type of data to the other versions.

PolicyCenter does not send renewal billing instruction to BillingCenter (PC-15082)

Issue

On receiving payment notification from BillingCenter, PolicyCenter tries to bind. If the policy is in a draft status, PolicyCenter does not bind and send the Renewal Billing Instruction to BillingCenter. The policy might be in draft status after underwriting edited something on the policy. This is an issue with BillingCenter 3.0.1 and later.

Workaround

Guidewire is aware of this issue.

Renewal processing for renewal conversion assumes BasedOn will be populated (PC-15851)

Issue

In the default configuration, renewal processing assumes the existence of a prior term or a `basedOn` version. Unless customized, this code can be problematic in some convert-on-renewal scenarios.

Workaround

If you are developing convert-on-renewal flows, be aware of the assumptions in the default configuration and modify the code accordingly.

SQL Query performance in free-text search (PC-18794)

Issue

There are open issues related to the performance of SQL query in free-text search.

Workaround

Guidewire is aware of this issue.

Cannot quote a renewal if the expiring policy term has a handled preemption (PC-19416)

Issue

In certain circumstances, if a policy term has a preemption that has been handled, when a renewal job is created, the **Edit Workorder** button is not available on the renewal. The only options are to bind or not take the renewal.

Workaround

Guidewire is aware of this issue.

Copy-data with an out-of-range date (PC-20279)

Issue

When working with the copy data feature, if you specify a specific date to copy data from and that date is out of range for the policy, PolicyCenter presents a warning that no applicable policy version could be found. However, the date selector still indicates the requested date, not the date that PolicyCenter will use for the copy if you choose to proceed.

Workaround

As appropriate, modify the user interface to reset the date selector or to hide the ability to select data to copy.

Gosu classes not intended for use (PC-23009)

Issue

Some Gosu classes are visible in Studio, but are not intended for use. You can distinguish these Gosu classes because they have no visibility annotations (neither `@Export` nor `@ReadOnly`) and they are not in a `gw` package.

Workaround

Do not use these methods in configuration. The methods are unsupported and may change or be withdrawn without notice.

Unbound renewal policy transactions can provide invalid dates (PC-25288)

Issue

If a policy renewal is in the renewal process and the user goes from the renewal job wizard to the policy file summary, the as-of date default in the **Date** field of the **Actions** panel will not be entirely valid, as it will be the `editEffectiveDate` of an unbound period from an incomplete job. The quote, reinsurance, and audit schedule, if enabled, screens display exceptions because appropriate periods do not exist on the specified dates.

Workaround

Edit `JobWizardInfoBar.pcf` to filter out any `PolicyPeriod` objects that cannot provide the `editEffectiveDate`.

Billing integration error when final audit is waived from audit policy transaction (PC-25500)

Issue

When the user waives a final audit from within a running audit policy transaction, the BillingCenter integration sends an incomplete message to BillingCenter. The message does not include the term number of the policy period. BillingCenter needs the term number to identify the BillingCenter policy period that needs the change.

Workaround

Guidewire is aware of this issue.

Cannot change availability tables columns after data added (PC-26836)

Issue

The Product Model Guide provides instructions on how to add columns to an availability lookup table. However, after loading availability data, you cannot add or remove columns.

Workaround

Guidewire is aware of this issue.

Payment screen not updated (PC-30594)

Issue

Updating a payment plan in BillingCenter does not update **Premium Summary**, **Payments**, and **Invoicing** fields on the PolicyCenter **Payment** screen. Even making a selection in the payment plan list view does not update these fields.

Workaround

To refresh these fields, navigate away from and return to the **Payment** screen.

PolicyPeriod.removeFromForms method causes DBCC error (PC-30947)

Issue

The `PolicyPeriod.removeFromForms` method causes a DBCC error.

Workaround

Do not use this method.

List bill with no matching invoice stream for a payment plan generates WSDL fault (PC-31004)

Issue

This issue occurs when a list bill account does not have a compatible invoice stream for one of the payment plans associated with the list bill account. When adding that list bill account to the policy, PolicyCenter throws an exception:

```
WsdlFault: java.lang.IllegalStateException: Cannot lazily create an invoice stream for charge with
owner {policy period id} on list bill account {account number}
```

Workaround

All payment plans must have compatible invoice streams configured in the billing system.

Comparing policy transactions does not work correctly with out-of-sequence transactions (PC-31190)

Issue

When comparing two policy transactions from the Policy Transactions screen, the comparison does not display changes included in the policy revisions being compared that had been made effective on a later date than the edit effective date of the associated transaction (either within that transaction or in a previous transaction). Changes made on earlier dates are included.

Workaround

Guidewire is aware of this issue.

Final audit setting changes if billing method changed in BillingCenter (PC-31360)

Issue

On renewal, final audit is changed from **No** to **Determined by business rule** when the billing method and payment plan have been changed in BillingCenter before starting the renewal. This also occurs with rewrite.

Workaround

Manually reset **Final Audit** to **No**.

Cannot complete unowned activities (PC-31610)

Issue

Cannot complete unowned activities even if you have the correct permissions. This issue is in PolicyCenter.

Workaround

Guidewire is aware of this issue.

Number of payment installments incorrect after reinstatement (PC-32171)

Issue

When integrated with BillingCenter, the number of payment installments may be incorrect after a reinstatement policy transaction. This affects general liability policies with final audit set to true after the policy is canceled. If the policy is reinstated, then the number of installments displayed may be incorrect. The installments in the billing system will still be calculated correctly.
The transaction can be completed without side effects.

Workaround

Guidewire is aware of this issue.

Message events against policy term archived in BillingCenter generates error (PC-33320)

Issue

If a policy period is archived in BillingCenter but not in PolicyCenter, that policy can be transferred to a new account owner in PolicyCenter. However, this causes an error in BillingCenter. The transfer is shown as successful in PolicyCenter, but the message queue includes this error: "The operation cannot be performed on an archived PolicyPeriod. The PolicyPeriod that is the target of the operation is archived." BillingCenter does not support transfer to a new account on archived policies or policies retrieved from archive.

Workaround

Guidewire is aware of this issue.

Can issue policy change against policy period retrieved from archive in BillingCenter (PC-33339)

Issue

If a policy period has been retrieved from the archive in BillingCenter and the policy period in BillingCenter is still closed, then you would expect that PolicyCenter would not allow you to issue a policy change. However,

PolicyCenter allows you to issue a policy change. BillingCenter rejects the policy change message and sends an error to PolicyCenter. The message remain in an unsent state in the message queue.

Workaround

Reopen the policy period in BillingCenter and resend the message.

Waiving final audit or premium report throws BadIdentifierException (PC-33351)

Issue

If you start a final audit and select **Waive** as the close option, PolicyCenter throws a `BadIdentifierException`. The waive final audit message is not sent to BillingCenter. This also affect premium reports.

Workaround

Waive the final audit from the **Audit Schedule** screen.

Availability limitations on schedule coverages (PC-34620)

Issue

Setting availability criteria on scheduled item coverages results in a null pointer exception when viewing policy differences.

Workaround

When configuring schedule coverages:
- Availability is supported on the schedule coverage and its coverage terms (for example, Scheduled Personal Property).
- Availability is supported on the scheduled item coverage pattern options.
- Availability is not supported for the scheduled item coverage and coverage terms (for example, Scheduled Personal Property Item). Set these types of clauses to always available. These clauses inherit their availability from the schedule coverage. That is, the scheduled item coverage is available whenever the schedule coverage is available.

Changes to Availability of existing Product Model Patterns become active immediately in rolling upgrade (PC-35054)

Issue

Changes to availability of existing product model patterns during rolling upgrade of the PolicyCenter cluster become active as soon as the first updated node restarts with the new configuration. For example, suppose that you want to end-date the availability of an existing coverage term pattern and replace it with a newly added coverage term pattern. In that case, you need to be aware that the end-dating becomes active immediately, even though the new coverage term is not available until rolling upgrade fully completes. This can create challenges during the rolling upgrade window as neither coverage term is available during that time. This is true of any product model pattern types that have Availability logic.

Workaround

Consider a two-phase rollout of the change to the product model pattern availability:
1. First, deploy the new product model pattern in a rolling upgrade and complete that upgrade.
2. Then, do one of the following:
   - Perform a second rolling upgrade of the product model and make the old product model pattern unavailable.
   - Use the Server Tools Reload Availability functionality to make the old product model pattern unavailable.

Issues with user interface elements when configured document management system not available (PC-35082)

Issue

Certain user interface controls show inappropriate status or are inexplicably disabled when a configured document management system is not available. The user interface is generally responsive to the presence of a

document management system. However, if that system is unavailable, you may see puzzling user interface state or behavior. This may include usually actionable controls unavailable, actionable controls that inexplicably do not take action, or conflicts between actionable and non-actionable user interface controls.

Workaround

You can check the availability of the document management system through the `gw.document.DocumentsActionsUIHelper.DocumentContentServerAvailable` property. Check this property as appropriate to improve the user experience when the document management system is unavailable.

Can promote an invalid rate book (PC-35087)

Issue

You can promote an invalid rate book to stage, approve it, and activate it. If a policy uses the invalid rate book for rating, PolicyCenter displays an error that the rate book is invalid.

Workaround

Create a new valid version of the invalid rate book that supersedes the existing rate book.

No user interface to set underwriting rule as externally managed (PC-35121)

Issue

In PolicyCenter 9.0.3, PolicyCenter added an `ExternallyManaged` property to the `UWIssueType` entity. Externally managed rules have their rule condition defined outside business rules, often in Gosu code. The underwriting **Rule** screen does not have a button to mark a rule as externally managed.

If you implement a new externally managed underwriting rule in Gosu code, you must define the `UWIssueType` properties in the underwriting **Rule** user interface. This interface does not have a button to mark the rule as externally managed.

Workaround

You can write code that sets the `ExternallyManaged` property to `true` on the `UWIssueType` entity instance for a rule. In the PolicyCenter user interface, you can add a button that accesses this code.

However, be careful about when the button is available. Rules cannot be toggled between `ExternallyManaged` and not `ExternallyManaged`. Make this button available to set `ExternallyManaged` to `true` on a brand new rule that has no information beyond the underwriting issue type fields defined.

Product Designer can insert empty XML tags that prevent rolling upgrade (PC-35123)

Issue

In PolicyCenter 9.0.3, Guidewire added the ability to make certain changes to product model patterns in a rolling upgrade of the servers in the PolicyCenter cluster. To support this change, Guidewire added a new configuration verification of the product model files to ensure that the rolling upgrade contains only valid (permitted) changes to the product model.

If you use Product Designer to modify a product model pattern created outside of Product Designer (for example, created in a release prior to PolicyCenter 8.0), Product Designer inserts empty XML tags for certain elements (`<AvailabilityScript/>`, for example). While these changes are generally legal to make in PolicyCenter, the upgrade configuration verifier flags the changes as illegal and thus blocks the rolling upgrade.

Workaround

Inspect any changes to product model pattern files (the `.xml` files) before you deploy them to production. If you discover new empty XML tags added by Product Designer, do one of the following:

- Remove the empty tags from the affected files before starting the rolling upgrade.
- Push the existing product model pattern XML files into production using a full database upgrade.

Cannot deploy new clauses with generic schedules in rolling upgrade (PC-35125)

Issue

It is not possible to deploy new coverage, condition, and exclusion patterns that include a generic schedule of items in a product model rolling upgrade of the servers in a PolicyCenter cluster. A generic schedule definition consists of two parts, the clauses and the definition of the schedule of items, defined in Gosu code. As Guidewire does not support the modification of existing Gosu code in a rolling upgrade, it is not possible to deploy a new schedule through a rolling upgrade.

Workaround

Deploy the new schedule in a full upgrade.

Need to restart PC server after adding a generic schedule (PC-35128)

Issue

After uploading a change from Product Designer that includes a generic schedule, PolicyCenter throws an `ErrantGosuClassException` exception. This was observed occurring during a submission policy transaction. The PolicyCenter server was in development mode.

Workaround

Restart PolicyCenter.

Out-of-sequence cancellation warning after changing Edit Effective Date (PC-35720)

Issue

An out-of-sequence cancellation warning can appear erroneously when you change the **Edit Effective Date** of a cancellation in the user interface. This happens when you edited a quoted cancellation, then change the edit effective date to a value prior to the previously set cancellation date.

Workaround

You can ignore this message if you were only notified the cancellation was out-of-sequence after changing the effective date of the cancellation. When the cancellation is quoted or bound, it will have the new edit effective date.

Avoid making a policy entity the Instruction target in a rate routine step

Issue

In Guidewire Rating Management, do not use a entity that is part of the policy as the target of the **Instruction** column on the left side of a rate routine step. This requires that PolicyCenter update the policy branch. This is an unexpected use of rate routines. Rate routines are for calculating costs for a policy. They are not expected to make changes to the policy branch.

For example, in a parameter set you have a writable `Vehicle` parameter defined as `entity.VehicleDriver` type. This entity has a `PolicyDriver` property that is a foreign key to a `PolicyDriver` entity instance. In a rate routine that uses this parameter set, avoid specifying `Vehicle.PolicyDriver` as the target of an **Instruction**.

Workaround

Guidewire is aware of this issue.

Product Designer known issues

PolicyCenter does not start in single currency display mode if the product model is configured to use multiple currencies (LPD-255)

Issue

Product Designer allows users to configure multicurrency features without regard for the setting of the `MultiCurrencyDisplayMode` in the PolicyCenter `config.xml` file. If you configure multiple currencies in a

product model object, and the value of the `MultiCurrencyDisplayMode` property is `SINGLE`, PolicyCenter does not start.

For example, if you add clauses that specify multiple coverage currencies and the PolicyCenter `MultiCurrencyDisplayMode` is set to `SINGLE`, the server fails to start and displays messages similar to the following:

```
ERROR ... PolicyLinePattern : ... [field : AvailableCoverageCurrencies] : - Multiple Coverage
Currencies defined for product not allowed in single Currency mode
ERROR ... [ExclusionPattern] : ... [DirectCovTermPattern] : ... [LimitsSet] : - The pattern specifies
limits for different currencies in SingleCurrencyMode
```

Workaround

If you intend to deploy a product model on PolicyCenter servers that are not configured for multiple currency display mode, do not configure multiple currencies in any part of the product model. However, you can test multicurrency products on a staging server configured for multicurrency while the production system is configured for a single currency.

Cannot display system table containing entity with more than one suffix (LPD-385)

Issue

Product Designer cannot properly display a system table that contains an entity with more than one suffix.

You can add additional columns to a system table by defining those columns in entity extension files. However, if you define columns in more than one extension file, Product Designer cannot display those columns. For example, you wish to add columns to the BOPClassCode system table. You create a `BOPClassCode.CA.etx` entity extension file that defines a CA column. You can view and edit this column in Product Designer. If you create a `BOPClassCode.WA.etx` entity extension file the defines a `WA` column, Product Designer cannot display both columns.

For example, Product Designer can properly display a system table containing an entity with the following definitions:

- `metadata/entity/BOPClassCode.eti`
- `extensions/entity/BOPClassCode.CA.etx`

But Product Designer cannot properly display a system table containing an entity with the following definitions:

- `metadata/entity/BOPClassCode.eti extensions/entity/BOPClassCode.etx`
- `extensions/entity/BOPClassCode.CA.etx`
- `extensions/entity/BOPClassCode.AZ.etx`

Workaround

Define additional columns in one entity extension file, and edit the columns in Product Designer. If this is not desirable, create multiple entity extension files and edit the system table in Guidewire Studio. However, the system table, with columns missing, will still be accessible in Product Designer.

Cannot use Product Designer to edit an extended object with no base object (PC-27763)

Issue

If you attempt to edit a part of the product model that is based on an entity extension file (`.etx`), and that extension has no base entity file (`.eti`), Product Designer displays errors and refuses to display the contents of the file.

Although Guidewire best practices recommend that you never create an entity file extension (`.etx` file) when no base entity (`.eti` file) exists, Guidewire Studio did not enforce this recommendation. However, the current version of Product Designer does not allow the use of an entity extension without a corresponding entity base file. If such an entity extension exists, Product Designer displays a **NullPointerException** when you attempt to access the affected objects. You then can use the Back command in the browser to view other pages, but you cannot configure the affected objects.

Workaround

Locate all `.etx` files that do not have corresponding `.eti` files, and rename them with an `.eti` extension.

System table cells display public IDs when referencing another system table (PC-28368)

Issue

If the cells in a particular column of a system table are referencing values from another system table, only the public IDs of those values are shown, not the values, when the system table is edited in Product Designer. In addition, users cannot search for values in these system table cells. The referenced system table, however, does not display the public ids so they cannot be viewed from within Product Designer.

Workaround

View the XML directly for the system table being referenced to determine the public IDs associated with the values that need to be added, edited or deleted. You can then use those public IDs directly.

Invalid audit schedules when audit schedule properties file is changed while user is editing audit schedule (PC-28146)

Issue

If you edit an audit schedule in Product Designer at the time that the associated audit schedule properties file is updated, you may not see the properties file changes reflected in Product Designer. This could result in an invalid audit schedule (that matches the previous audit schedule properties file).

Workaround

Perform the following steps:

- In Product Designer, revert the changes to affected audit schedules.
- To load the audit schedule properties file changes into Product Designer, restart Product Designer, logout, or change the changelist.
- Reopen the affected audit schedules.

An exception might appear when you run Product Designer (PC-28531)

Issue

An `IncompatibleRemoteServiceException` might appear if you run Product Designer 8.0.3 or later after running a previous version of Product Designer.

Workaround

Refresh the browser cache by selecting `Ctrl+F5`.

Studio/Platform known issues

Chrome browser cannot display product documentation in HTML format (DOC-7251)

Issue

If you use the Google Chrome browser, you can view the HTML Guidewire product documentation only if it is served by an HTTP server using the `http://` protocol. The Chrome browser cannot load HTML product documentation from your local disk by using the `file://` protocol.

Workaround

Use a different browser, such as Microsoft Internet Explorer. If you use the HTML documentation only to access the PDF files, you can open them directly in the `pdf` subdirectory of the `doc` directory.

Alignment Property Does Not Work for Inputs in a DetailView (PL-29429)

Issue

Inputs in a detail view do not align to the right or center. They always align to the left, regardless of the value of the `align` property.

Workaround

If you need to right align a series of items such as a set of monetary values, put them into a list view instead of a detail view. You can then set the alignment as needed.

JMS inbound integration implementations must be OSGi or Java plugins (PL-32054)

Issue

Writing JMS inbound integration handlers requires casting an object to the type `javax.jms.Message`. In Gosu, for some app servers this fails with the message: `Caused by:`
`gw.lang.parser.exceptions.ErrantGosuClassException: GosuClass`
`test.GosuJMSMessageReplyHandler has errors, and cannot be used at runtime.`

Workaround

Implement JMS handler code for inbound integration in Java. Guidewire recommends that for inbound integration Java code, you implement your code as an OSGi plugin, not as a standard (non-OSGi) Java plugin.

Compilation errors after removing a delegate (PL-34619)

Issue

When you remove a delegate, some internal code is not generated properly, resulting in compilation errors.

Workaround

In Guidewire Studio, on the **Codegen** menu, click **Generate Metadata Classes**.

Several permissions are defined incorrectly (PL-34946)

Issue

The following permissions are defined incorrectly:
- `perm.Holiday.edit`
- `perm.Holiday.delete`
- `perm.Note.view`

Workaround

To any role that contains the `holidaymanage` permission, add the `buswkmanage` permission to that role. To any role that contains the `noteview` permission, add the `noteedit` permission to that role.

`Alt+Shift+L` to reload PCF files does not work when the server is run from the command line (PLWEB-5819)

Issue

If you run the server from the command line with the command `gwb runServer`, then pressing `Alt+Shift+L` in the application interface does not reload PCF files.

Workaround

Run the server from Guidewire Studio instead.

Server Tools downloads might time out (PLWEB-6184)

Issue

On the **Server Tools** pages, some downloads might take too long and generate a timeout error.

Workaround

Increase the value of the `WebUIAJAXTimeout` configuration parameter.

During rule import, selecting the deployed rule results in incorrect rule version and status (BIZ-912)

Issue

Under certain circumstances, the business rule import process sets the rule version incorrectly. This occurs in the following circumstances:

- The existing and importing rules show as a version conflict.
- The existing or importing rule version is in the deployed state.
- The other rule version has a status other than deployed (draft, staged, approved).
- The deployed version of the rule is selected.

In all of these cases, the rule version is set to Approved with the deployed rule version number. For example, the deployed version is 1 and the version after import is 1+ Approved.

Workaround

Guidewire is aware of this issue.

Batch process freezes if checkInitialConditions returns false (PL-36852)

Issue

If the `BatchProcessBase.checkInitialConditions` method returns false, the application freezes any ongoing work in that batch process. It then continues to retry the `checkInitialConditions` method for a period of 10 minutes until either the check returns `true` or the time limit expires. Although this is the expected behavior, it is possible for the recurring execution of the `checkInitialConditions` method to cause performance problems. This is also a change from the behavior of this method in releases prior to the 9.0.0 release.

Workaround

Add code similar to the following to the batch process implementation class. The example code may need some modification for a custom work queue writer.

```
private var _readyToRun = true

override function checkInitialConditions() : boolean {
  _readyToRun = ... // verify initial conditions here or in delegated method
  return true
}

override function doWork() {
  // verify initial conditions here and short-circuit method if conditions are not met
  if(!_readyToRun) {
    setOperationsFailedReasons( { "No work" } )
    return
  }
  // body of batch process follows
    ...
}
```

Apache ZooKeeper security issue CVE-2017-5637 (PL-37446)

Issue

Apache ZooKeeper has released a security update (CVE-2017-5637) for its server. Guidewire Solr Extension code uses only the client side of the ZooKeeper software, not the server side. As such, the underlying security issue that ZooKeeper has addressed does not affect Guidewire applications directly.

Workaround

If you wish to incorporate the ZooKeeper security update, Guidewire recommends that you update the server-side ZooKeeper software from version 3.4.6 to 3.4.10.

Guidewire PolicyCenter 9.0.4 Release Notes

Release 9.0.4

Overview

These release notes contain the following topics:

Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

IMPORTANT If you skipped one or more upgrade releases to PolicyCenter, be sure to read the Release Notes for those releases to learn about changes and fixed issues.

Version Number

This release of Guidewire PolicyCenter is 9.0.4.
- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 8.0.1 or later. The latest maintenance release is preferred.

  If you are implementing multiple settlement currencies, BillingCenter 8.0.2 or later is preferred.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 8.0.0 or later. The latest maintenance release is preferred.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 8.0.1 or later. The latest maintenance release is preferred.

Installing This Release

Refer to the following:

- The *PolicyCenter Installation Guide* for general installation information
- Prior PolicyCenter release notes for any versions that you have skipped

This topic includes:

- "Upgrade Information" on page 242

Upgrade Information

This topic includes:

- "Significant Changes to Upgrade Procedure" on page 242
- "Exception when upgrading UWIssue objects in sample data from 7.0.x (PC-34046)" on page 242

Significant Changes to Upgrade Procedure

IMPORTANT The procedure for upgrading to PolicyCenter 9 from PolicyCenter 8 or previous releases has changed significantly. In PolicyCenter 9, there are new tools available for performing the upgrade. Do not rely on your past knowledge of the upgrade process to begin the upgrade procedure. Before beginning your upgrade, review the topic "Configuration Upgrade Overview" in the *PolicyCenter Upgrade Guide*.

Exception when upgrading UWIssue objects in sample data from 7.0.x (PC-34046)

When upgrading from PolicyCenter 7.0.x, database upgrade throws an `IllegalArgumentException` exception when encountering `UWIssue` objects in the sample data. The exception occurs because the upgrade expects to find typelist values added in PolicyCenter 8.0.x.

To workaround this issue, delete `UWIssue` objects in the sample data from the PolicyCenter 7.0.x database or add the typelist values necessary.

Support

For assistance, visit the Guidewire Community.

Guidewire customers

```
https://community.guidewire.com
```

Guidewire partners

```
https://partner.guidewire.com
```

Major Issues and Changes

This topic contains major issues and changes that can affect your installation.

This topic includes:

- "New and Changed Features" on page 243
- "Base PCF File Changes" on page 243
- "Base Resource Changes" on page 243
- "Changes in This Release Provided in Upgrade Diff Report" on page 243

New and Changed Features

For information on new features and other changes, see the topic "New and Changed in PolicyCenter 9.0.4" in the *PolicyCenter New and Changed Guide*.

Base PCF File Changes

The following link requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 9.0.3 to 9.0.4

To view a report of the changes to the base PCF files, *click here*.

Base Resource Changes

PolicyCenter release 9.0.3 to 9.0.4

There are no changes to the base resources.

Changes in This Release Provided in Upgrade Diff Report

Guidewire provides a report detailing certain differences between the current release and your prior release. This report describes changes in display keys, entities, typelists, and the Gosu API. To obtain your custom Upgrade Diff Report, visit the Guidewire Community.

Improvements and Resolved Issues

This topic describes improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This list is not intended to be comprehensive.

This topic includes:

- "Application Improvements and Resolved Issues" on page 243
- "Platform Improvements and Resolved Issues" on page 247

Application Improvements and Resolved Issues

ID	Description
Archiving	
US6806	The `PolicyTerm.isArchived` method used to return `true` if one or more of its policy periods were archived, even if not all were archived. With this release, the `isArchived` method has been replaced with the `hasArchivedPolicyPeriod` method. In your code, change occurrences of `isArchived` with `hasArchivedPolicyPeriod`. The `isArchived` method now always returns `false`. `PolicyTermEnhancement.gsx` now has a `checkArchived` property getter that returns the value of `hasArchivedPolicyPeriod`.
Administration	
PC-34432	Added an **All** filter to the **My Accounts** list view on the **Desktop** tab.
PC-35048	External user validation is now done in the `UserExternalValidation` Gosu rule. With this change, validation is always performed. Previously you could skip validation by accepting warnings.

ID	Description
PC-35217	PolicyCenter now displays a warning validation message if you import a rate book through **Administration→Utilities→Import data** when the server is in production mode. This can cause data corruption in an active rate book with rate factors. **Import data** performs XML validation only; it does not validate the rate book. The message advises you to use **Rating→Rate Books→Import from XML** instead because it validates the rate book before importing and does not overwrite rate factors in an active rate book.

Billing / Payments

ID	Description
PC-34768	Fixed an issue where you could not move a policy with a withdrawn policy change transaction between accounts in PolicyCenter. The integration with BillingCenter caused this issue. The transferred policy is now correctly sent to BillingCenter.

Business Rules

ID	Description
PC-35069	During side by side comparison with externally managed rules, only fields relevant to externally managed rules are displayed. This was previously a known issue.
PC-35071	When you import an externally managed rule into PolicyCenter, it is no longer marked as an invalid rule on the **Business Rules→Import/Export Status→Review Import** screen. This was previously a known issue.
PC-35337	Fixed an issue which could have resulted in PolicyCenter overriding changes you made to files in the configuration module, such as changes to PCF files.

Codegen

ID	Description
PC-35164	Fixed an issue where codegen failed to remove the file associated with a clause if you removed that clause from the product model and if that clause was the last one of its type associated with a specific coverable. For example, this occurred when removing the last Condition associated with a Vehicle.

Contacts

ID	Description
PC-35036	Fixed an issue with thread safety in the `ABContactSystemPlugin` plugin.

Data Model / Core Entities

ID	Description
PC-35305	Fixed an issue where PolicyCenter created duplicate answers on `PolicyPeriod`, `PolicyLine`, and `PolicyLocation`. If you extend the `AnswerContainer` delegate to other entities, you can avoid this issue by setting `Owner = true` for entities that implement `AnswerContainer`.

Diffs

ID	Description
PC-35143	The **Policy Review** screen now shows differences even for entities that were removed on the `EditEffectiveDate`.

Forms

ID	Description
PC-34245	In the `FormInferenceEngine#findPotentialForms` method, added a new step that filters form patterns based on jurisdiction. This can reduce the number of patterns sent to the form data creation step and the overall time to do form inference.

Geocoding

ID	Description
PC-35024	Updated the Geocode plugin to support Bing Maps V8 with REST API.

Job Infrastructure

ID	Description
PC-35252	When creating a `RegionZone` entity instance for a given `Country`, the entity now stores the country in the database instead of the default country.

Line of Business - PA

ID	Description
PC-35066	Fixed an issue which caused an `IllegalArgumentException` when starting side-by-side quoting in a settlement currency other than U.S. dollars. With this fix, amounts in side-by-side quotes are displayed in the settlement currency.

Policy Holds

ID	Description
PC-35257	Fixed an issue with adding a hold region to a policy hold in countries such as Japan. In the **Search for Regions** popup, you can now select **Region Type** for all countries.
Performance Improvement	
PC-35079	Improved performance of importing Locations from spreadsheet in Excel format.
PC-35088	Improved performance of the **Rate Table Definition** screen.
PC-35209	Improved performance of the **Roles** screen.
Policy Transactions	
DE2883	Fixed an issue with zero width policy location in rewrite new account policy transactions. This caused an `IllegalArgumentException` for `PolicyLocation` when purging a policy that was rewritten to a new account.
PC-35127	Fixed an issue which caused a null pointer exception in an out-of-sequence policy change.
PC-35147	Fixed an issue in a renewal which disabled the **Release Lock**, **Quote**, and other buttons on the **View Quote** screen after editing the policy transaction for a second time.
Product Model	
PC-35067	Guidewire now disables the **Reload Availability** button on the Server Tools **Product Model Info** screen during rolling upgrade. This fixes a previously known issue.
PC-35072	You can now perform additive changes to the types of Offering Selections listed below during a Rolling Upgrade: • Policy Term Selection • Question Set Selection • Policy Line Selection • Product Modifier Selection • Modifier Selection • Coverage Selection • Exclusion Selection • Condition Selection • CovTerm Selection • CovTermOpt Selection • CovTermPack Selection This fixes a previously known issue. Guidewire does not allow, however, the modification or removal of existing Offering Selections during a Rolling Upgrade.
PC-35200	Fixed an issue where existing product model patterns were not available after database upgrade in a clustered environment.
PC-35358	Because of changes to product model pattern locking in PolicyCenter 9.0.3, upgrade now makes changes to `PMLockedField` entities in the database.
PC-35394	Now, a full PolicyCenter upgrade without a product version change (for example, 9.0.4 to 9.0.4) automatically triggers activation of product model patterns added during the upgrade. This does not happen with a rolling upgrade, however. You must still manually activate the new patterns after the rolling upgrade.
Quote purging	
PC-35076	In quote purging, PolicyCenter does not purge policy transactions (jobs) or policy periods that have contingencies. You can now configure whether to purge: • A policy transaction that has a contingency attached to any of its policy periods • A policy period that has a contingency attached to it To purge policy transactions or policy periods with contingencies, make the following changes in `PCPurgePlugin.gs`: • In the `skipPolicyPeriodForPurge` method, modify the code related to policy periods with contingencies. • In the `prepareForPurge` method, delete or null out the foreign key from the contingency to the policy period.

ID	Description
	Note In the base configuration, there is one situation where PolicyCenter will purge a policy transaction with contingencies. If purging removes all policy transactions associated with a policy, PolicyCenter also removes the `Policy` object. As a result of removing the `Policy`, PolicyCenter also removes the contingencies. This situation occurs when a `Policy` only exists because of an unbound submission.
PC-35264	Fixed an issue with quote purging where purge executed before prune caused an `IllegalStateException` while pruning.
US17726	In the `PurgeAPI` web service, methods containing the word `Purge` are deprecated. In the replacement method, the word `Purge` is replaced by `Destroy`. For example, the `isDoNotPurgePolicyPeriod` method is now `isDoNotDestroyPolicyPeriod`. A version trigger renames `DoNotPurge` database columns to `DoNotDestroy`.
Rating	
PC-34695	Fixed a previously known issue where a value provider for coverage term options did not return any values. This issue only occurred for coverages and coverage terms created in 9.0.0 or later with different values for `CodeIdentifier` and `PublicId`.
PC-35085	Fixed a previously known issue where any user, internal or external, could delete rate tables, rate books, rate table definitions, rate routines, and parameter sets without having the necessary permissions. With this fix, users must have proper permission (`perm.System.ratebookedit`) to add, edit, and delete rate books, rate routines, rate table definitions, and parameter sets.
PC-35086	When promoting rate books to stage, if there are errors related to a rate table, the rate table name is now displayed in the validation results instead of the rate table code.
PC-35103	Fixed an issue where the **Argument Sources→Source of Argument Objects** in the **Rate Table Definition** screen was editable when `ArgumentSourceSet` was used in the rate book and the rate book included rate routines that were not in draft state.
PC-35116	Fixed a memory leak which occurred when rating many policy transactions with rating worksheets enabled.
PC-35141	Added a denormalized `TaxAndSurchargesRPT` field to `PolicyPeriod`. This field improves performance of PCF files such as the **Billing** and **Summary** screens that display taxes and surcharges for policies with many costs. When you quote a policy period, the `TaxSurchargesRPT` field is populated and used through the lifecycle of the quote. A getter for the field replaces less efficient calls to methods that calculate taxes and surcharges for a policy period. There is no upgrade trigger to populate the field for existing policy periods, because a getter accesses and populates the field if necessary. The getter, `getTaxAndSurchargesRPT`, returns the value of the field if it is already populated. If the value needs to be calculated, the getter calculates and returns the value. The `RateAmountType` typelist now includes the pre-existing `TaxSurcharge` typecode for taxes and surcharges. There is a new type filter, `TaxesAndSurcharges`, on the `RateAmountType` typelist. This type filter includes the `TaxSurcharge` typecode. The `getTaxAndSurchargesRPT` getter uses the typelist to get costs for taxes and surcharges. Calls to `policyPeriod.AllCosts.TaxSurcharges.AmountSum(policyPeriod.PreferredSettlementCurrency)` are replaced with the calls to the `getTaxAndSurchargesRPT` getter. The getter has the same logic of the call it replaces, and it matches pre-existing behavior. This behavior includes never returning `null`, since the method will return a `MonetaryAmount` of `0` even if there are no costs.
PC-35378	Fixed an issue which resulted in `IllegalArgumentException: Comparison method violates its general contract!` when exporting a rate book to XML.
PC-35409	Fixed an issue which caused the `RateBookCoreEnhancement.RateTablesMissingInCalcRoutines` method to throw this exception: `IllegalStateException: More than one element matches the given condition`
Revisioning	
PC-35107	Fixed an issue in which adding an Additional Interest to an entity in an out-of-sequence change that was already added to the same entity in a future dated change resulted in losing data about the Additional Interest.
PC-35340	Fixed an issue with out-of-sequence changes where sliced `DiffAdd` objects were not applied correctly to preemption branches and renewals.

ID	Description
PC-35348	
Risk Assessment with Spotlight	
PC-34884	Accessing Spotlight from PolicyCenter no longer requires second authentication. Previously, after invoking Spotlight, the user was asked to authenticate again with PolicyCenter credentials. Likewise, returning to PolicyCenter from within Spotlight no longer requires authenticating a second time.
User Tools (Activities, Notes, etc)	
PC-35022	Fixed an issue on the **Activity** tab where **View Notes** displayed notes not associated with the activity.

Platform Improvements and Resolved Issues

The following are the primary improvements and issues corrected in this release:

ID	Description
Database	
PL-36421	Fixed an issue in which log messages from database startup and upgrade contained a mixture of hard-coded English and the localized language.
Database Instrumentation	
PL-36363	Fixed an intermittent issue with Oracle AWR reports caused by the use of an incorrect datatype.
PL-36679	Correctly ordered the column headings in the Message Queue Depth report in the AWR download on the **Server Tools** page.
Database Support - General	
PL-34712	If using PolicyCenter to update database statistics on Oracle or SQL Server, the configuration now allows an option to specify `force` as a table action. PolicyCenter updates statistics if running incremental statistics regardless of the percentage threshold.
PL-34965	Removed the `databasedegree` and `samplingpercentage` attributes from the `histogramstatistics` element in `database-config.xml`. It has no meaning or effect at the column (histogram) level.
Database Support - Oracle	
PL-36624	Fixed an issue that caused an Oracle error when using the browser **Find** to search localized text in a long note.
Database Upgrade	
PL-36161	The database upgrade step of migrating a SQL Server database from using four-byte integers and the `datetime` datatype to eight-byte big integers and the `datetime2` datatype has been rewritten to run much faster. Note: This conversion will not be available in 10.0.x releases and must be done while still on a 9.0.x release.
PL-36291	Fixed an issue where upgrading a SQL Server database from 7.0 to 9.0 caused a data truncation error.
Document Management	
PL-36623	Fixed an issue where Note links to a document were broken if the `IDocumentMetadataSource` plugin was enabled.
PL-36738	Fixed an issue with the rendering of documents linked in the body of a note. When retrieving such documents, the `IDocumentContentSource` plugin might ignore the `TargetHiddenFrame` property of the `DocumentContentsInfo` object. The `TargetHiddenFrame` property specifies whether the document web page is opened in a hidden browser frame.
Email	
PL-36644	Fixed a `NullPointerException` issue when processing an email in the `MessageTransport` plugin. The issue occurred whenever the recipient's email address was invalid.

ID	Description
Entities/Metadata	
PL-36423	You can now enable histograms on base typekey columns by using the new `createhistogram` attribute of the `<typekey-override>` element.
PL-36818	Fixed an issue that occurred after merging the product and data models using the Guidewire Upgrade Tools to upgrade to version 9.0.x. An exception was thrown when the Studio Codegen → `Generate Everything` menu item was selected.
PL-36838	The description of the `Contact` entity in the **Data Dictionary** now displays correctly.
Gosu	
GOSU-31	Studio and the Gosu compiler now enforce the restriction that methods, including property accessor methods, cannot be both final and static.
GOSU-32	Fixed an issue where a `BigDecimal` to `MonetaryAmount` comparison caused an error.
GOSU-33	Fixed an issue in auto-casting to structural types.
GOSU-34	Fixed a Gosu parser issue that caused a compilation error.
GOSU-38	Fixed an issue that prevented the Studio debugger from evaluating some expressions.
GOSU-40	Fixed an issue in compiling Gosu code that used the compound assignment operator and dimensions.
GOSU-42	Fixed an issue that created incorrect byte-code for enhancement methods on Gosu interfaces.
PL-36712	Fixed an issue that prevented a protected inner enumeration being visible to a subclass of the enclosing class.
PL-36799	Fixed an issue that prevented a protected inner class being visible to a subclass of the enclosing class.
Infrastructure	
PL-36755	Fixed an issue where the "PolicyCenter ready" message appeared before the user could log in.
Inbound Integration	
PL-36630	Fixed an issue where file-based Inbound Integration did not process files that existed in the incoming directory at server startup.
Integration	
PL-36643	Fixed an issue where the `suite-config.xml` file's gw.pc.env parameter would be ignored.
PL-36699	Fixed an issue with the `gwb exportWsdl` command where nested XSD files were not included in the generated JAR file.
Messaging	
PL-36546	Fixed an issue where the message queue status is shown as `Started` even though the message queue did not start.
PL-36645	Fixed an issue where paging was not enabled to scroll through the entries on the **Administration**→**Monitoring**→**Message Queues**→**Destination** screen.
PL-36696	To improve troubleshooting operations, all failures to suspend or resume a message destination are logged.
Metadata Code Generator	
PL-36652	Fixed an error that occurred when a typelist had the same name as an entity subtype.
PL-36748	Fixed an issue where attempting to remove an index using the `<remove-index>` element would result in an error that the index "is not perf-only".
PCF - Widget - Address AutoFill	
PLWEB-6335	Fixed an issue where an invalid postal code would sometimes not be highlighted.
PCF - Widget - Menu	

ID	Description
PLWEB-6318	Fixed an issue with `ListView` header menus not opening when using a Microsoft Surface device.
PCF - Widget - Modal Cell	
PLWEB-6336	Fixed an issue where text would not wrap in a modal cell in a list view.
Persistence	
PL-36585	Fixed an issue where incorrect Java classes would be generated when an entity implemented a structure.
PL-36813	Fixed an issue where running `gwb genDataMapping` would not populate the Description column in the resulting `DataMap.csv` file.
Rolling Upgrade	
PL-36333	Added an `autoupgrade` attribute to the `<database>` element in file `database-config.xml`. This attribute takes two values: • `full`: Full takes precedence and initiates a full upgrade assuming all other necessary conditions are met. • `manual`: Requires that you set either the database upgrade type (in Server Tools **Upgrade and Versions screen**) or the date system property.
Search	
PL-34987	Updated HttpClient library to 4.5.3.
Web - UI/Runtime	
PLWEB-6317	Fixed an issue that prevented an `ExitPoint` from working within a worksheet.
PLWEB-6360	Fixed an issue where the error message "Please check server log" was shown, but there was no information added to the log.
Web Services - WSI (New)	
PL-36659	Fixed an issue where a web service call might be directed to an incorrect web service provider. The issue could occur if the definition of a `<webservice-collection>` included an `<override-url>` element that did not specify an `<env>` element. In such cases, the incorrect `<url>` element might be selected.

Known Issues and Limitations

This topic describes known issues with this release of Guidewire PolicyCenter.

This topic includes:

> **Note:** For maintenance releases, Guidewire often defers fixing configuration issues that require merging files during the upgrade. Workarounds to many of these issues are listed in the following topics. The goal of this policy is to make upgrades as straightforward as possible.

PolicyCenter Known Issues

Provide a way to filter out added and removed subentities in a policy differences tree (PC-7929)

Issue – Need a way to filter out the children of an added or removed entity in a policy differences tree.

Workaround – Guidewire is aware of this issue.

When using parallel sessions, approvals made by underwriter are not updated for second user simultaneously viewing same work order (PC-12322)

Issue – Approvals made by an underwriter are not updated for a second user simultaneously viewing the same work order. This occurs in the following sequence of events:

- An agent processes a submission and issues arise
- The agent calls an underwriter, and remains in the policy
- The underwriter approves the issue or issues
- The agent refreshes the **Risk Analysis** screen by going to a different wizard step and returning to the **Risk Analysis** screen

After this sequence of events:

- The new approvals do not appear on the agent's screen
- However, the agent can bind or issue the policy (based on the new approvals by the underwriter)

Workaround – The agent can leave the policy and return to it. The approvals appear as expected. Alternatively, the agent can see the approvals by clicking the **Details** button when issuing the policy.

Must enter VIN when moving from quick quote multi-version or side-by-side to full application (PC-13099)

Issue – In a quick quote, you do not have to enter the VIN or other fields that are required for a full application submission. When converting a quick quote with multi-version or side-by-side quoting to full application, to resolve validation issues, you must enter the VIN for each of the versions. In a job with side-by-side quoting, this issue can occur with fields that are not base data.

Workaround – Add code that copies this type of data to the other versions.

PolicyCenter does not send renewal billing instruction to BillingCenter (PC-15082)

Issue – On receiving payment notification from BillingCenter, PolicyCenter tries to bind. If the policy has a renewal transaction in draft status, PolicyCenter does not bind and send the Renewal Billing Instruction to BillingCenter. The policy might be in draft status after an underwriter edited something on the policy. This is an issue with BillingCenter 3.0.1 and later.

Workaround – Guidewire is aware of this issue.

Renewal processing for renewal conversion assumes BasedOn is populated (PC-15851)

Issue – The `startNewRenewal` method in `gw.webservice.pc.pc700.job.PolicyRenewalAPI` assumes the existence of a prior term or a `basedOn` version. Unless customized, this method can cause problems in certain convert-on-renewal scenarios. This method has been deprecated.

Workaround – If you are developing convert-on-renewal flows, use the `gw.webservice.pc.pc700.job.startConversionRenewal` method.

SQL Query performance in free-text search (PC-18794)

Issue – There are open issues related to the performance of SQL query in free-text search. This occurs the first time that you load the PolicyCenter data into the Solr index database.

Workaround – Guidewire is aware of this issue.

Cannot quote a renewal if the expiring policy term has a handled preemption (PC-19416)

Issue – In Commercial Package Policy in the base configuration, if a policy term has a preemption that has been handled, when a renewal job is created, the **Edit Workorder** button is not always available on the renewal. When this occurs, the only options are to bind or not take the renewal.

Workaround – Guidewire is aware of this issue.

Copy-data with an out-of-range date (PC-20279)

Issue – When working with the copy data feature, if you specify a specific date to copy data from and that date is out of range for the policy, PolicyCenter presents a warning that no applicable policy version could be found. However, the date selector still indicates the requested date, not the date that PolicyCenter will use for the copy if you choose to proceed.

Workaround – As appropriate, modify the user interface to reset the date selector or to hide the ability to select data to copy.

Gosu classes not intended for use (PC-23009)

Issue – Some Gosu classes are visible in Studio, but are not intended for use. You can distinguish these Gosu classes because they have no visibility annotations (neither `@Export` nor `@ReadOnly`) and they are not in a `gw` package.

Workaround – Do not use these methods in configuration. The methods are unsupported and may change or be withdrawn without notice.

Unbound renewal policy transactions can provide invalid dates (PC-25288)

Issue – If a policy renewal is in the renewal process and the user goes from the renewal job wizard to the policy file summary, the as-of date default in the **Date** field of the **Actions** panel will not be entirely valid, as it will be the `editEffectiveDate` of an unbound period from an incomplete job.The quote, reinsurance, and audit schedule, if enabled, screens display exceptions because appropriate periods do not exist on the specified dates.

Workaround – Edit `JobWizardInfoBar.pcf` to filter out any `PolicyPeriod` objects that cannot provide the `editEffectiveDate`.

Billing integration error when final audit is waived from Audit job (PC-25500)

Issue – When the user waives a final audit from within a running Audit job, the BillingCenter integration sends an incomplete message to BillingCenter. The message does not include the term number of the policy period. BillingCenter needs the term number to identify the BillingCenter policy period that needs the change.

Workaround – Guidewire is aware of this issue.

Cannot change availability tables columns after data added (PC-26836)

Issue – The Product Model Guide provides instructions on how to add columns to an availability lookup table. However, after loading availability data, you cannot add or remove columns.

Workaround – Guidewire is aware of this issue.

Cannot use Product Designer to edit an extended object with no base object (PC-27763)

Issue – If you attempt to edit a part of the product model that is based on an entity extension file (`.etx`), and that extension has no base entity file (`.eti`), Product Designer displays errors and refuses to display the contents of the file.

Although Guidewire best practices recommend that you never create an entity file extension (`.etx` file) when no base entity (`.eti` file) exists, Guidewire Studio did not enforce this recommendation. However, the current version of Product Designer does not allow the use of an entity extension without a corresponding entity base file. If such an entity extension exists, Product Designer displays a **NullPointerException** when you attempt to access the affected objects. You then can use the Back command in the browser to view other pages, but you cannot configure the affected objects.

Workaround – Locate all `.etx` files that do not have corresponding `.eti` files, and rename them with an `.eti` extension.

Payment screen not updated (PC-30594)

Issue – Updating a payment plan in BillingCenter does not update **Premium Summary**, **Payments**, and **Invoicing** fields on the PolicyCenter **Payment** screen. Even making a selection in the payment plan list view does not update these fields.

Workaround – To refresh these fields, navigate away from and return to the **Payment** screen.

PolicyPeriod.removeFromForms method causes DBCC error (PC-30947)

Issue – The `PolicyPeriod.removeFromForms` method causes a DBCC error.
Workaround – Guidewire is aware of this issue. Do not use this method.

List bill with no matching invoice stream for a payment plan generates WSDL fault (PC-31004)

Issue – This issue occurs when a list bill account does not have a compatible invoice stream for one of the payment plans associated with the list bill account. When adding that list bill account to the policy, PolicyCenter throws an exception:

```
WsdlFault: java.lang.IllegalStateException: Cannot lazily create an invoice stream for charge with
owner {policy period id} on list bill account {account number}
```

Workaround – All payment plans must have compatible invoice streams configured in the billing system.

Comparing policy transactions does not work correctly with out-of-sequence transactions (PC-31190)

Issue – When comparing two policy transactions from the Policy Transactions screen, any changes included in the policy revisions being compared that had been made effective on a later date than the edit effective date of the associated transaction (either within that transaction or in a previous transaction) are not displayed in the comparison. Changes made on earlier dates are included.
Workaround – Guidewire is aware of this issue.

Final audit setting changes if billing method changed in BillingCenter (PC-31360)

Issue – On renewal, final audit is changed from **No** to **Determined by business rule** when the billing method and payment plan have been changed in BillingCenter before starting the renewal. This also occurs with rewrite.
Workaround – Manually reset **Final Audit** to **No**.

Cannot complete unowned activities (PC-31610)

Issue – Cannot complete unowned activities even if you have the correct permissions. This issue is in PolicyCenter.
Workaround – Guidewire is aware of this issue.

Number of payment installments incorrect after reinstatement (PC-32171)

Issue – When integrated with BillingCenter, the number of payment installments may be incorrect after a reinstatement policy transaction. This affects general liability policies with final audit set to true after the policy is canceled. If the policy is reinstated, then the number of installments displayed may be incorrect. The installments in the billing system will still be calculated correctly.
Workaround – Guidewire is aware of this issue. The transaction can be completed without side effects.

Message events against policy term archived in BillingCenter generates error (PC-33320)

Issue – If a policy period is archived in BillingCenter but not in PolicyCenter, that policy can be transferred to a new account owner in PolicyCenter. However, this causes an error in BillingCenter. The transfer is shown as successful in PolicyCenter, but the message queue includes this error: "The operation cannot be performed on an archived PolicyPeriod. The PolicyPeriod that is the target of the operation is archived." BillingCenter does not support transfer to a new account on archived policies or policies retrieved from archive.
Workaround – Guidewire is aware of this issue.

Can issue policy change against policy period retrieved from archive in BillingCenter (PC-33339)

Issue – If a policy period has been retrieved from the archive in BillingCenter and the policy period in BillingCenter is still closed, then you should not be allowed to issue a policy change in PolicyCenter. However, PolicyCenter

allows you to issue a policy change. BillingCenter rejects the policy change message and sends an error to PolicyCenter. The message remain in an unsent state in the message queue.

Workaround – Reopen the policy period in BillingCenter and resend the message. This workaround requires that BC-15815 be fixed.

Waiving final audit or premium report throws BadIdentifierException (PC-33351)

Issue – If you start a final audit and select **Waive** as the close option, PolicyCenter throws a `BadIdentifierException`. The waive final audit message is not sent to BillingCenter. This also affect premium reports.

Workaround – Waive the final audit from the **Audit Schedule** screen.

Availability limitations on schedule coverages (PC-34620)

Issue – Setting availability criteria on scheduled item coverages results in a null pointer exception when viewing policy differences.

Workaround – When configuring schedule coverages:

- Availability is supported on the schedule coverage and its coverage terms (for example, Scheduled Personal Property).
- Availability is supported on the scheduled item coverage pattern options.
- Availability is not supported for the scheduled item coverage and coverage terms (for example, Scheduled Personal Property Item). Set these types of clauses to always available. These clauses inherit their availability from the schedule coverage. That is, the scheduled item coverage is available whenever the schedule coverage is available.

Changes to Availability of existing Product Model Patterns become active immediately in rolling upgrade (PC-35054)

Issue – Changes to availability of existing product model patterns during rolling upgrade of the PolicyCenter cluster become active as soon as the first updated node restarts with the new configuration. For example, suppose that you want to end-date the availability of an existing coverage term pattern and replace it with a newly added coverage term pattern. In that case, you need to be aware that the end-dating becomes active immediately, even though the new coverage term is not available until rolling upgrade fully completes. This can create challenges during the rolling upgrade window as neither coverage term is available during that time. This is true of any product model pattern types that have Availability logic.

Workaround – Consider a two-phase rollout of the change to the product model pattern availability:

1. First, deploy the new product model pattern in a rolling upgrade and complete that upgrade.
2. Then, do one of the following:
   - Perform a second rolling upgrade of the product model and make the old product model pattern unavailable.
   - Use the Server Tools Reload Availability functionality to make the old product model pattern unavailable.

Issues with user interface elements when configured document management system not available (PC-35082)

Issue – Certain user interface controls show inappropriate status or are inexplicably disabled when a configured document management system is not available. The user interface is generally responsive to the presence of a document management system. However, if that system is unavailable, you may see puzzling user interface state or behavior. This may include usually actionable controls unavailable, actionable controls that inexplicably do not take action, or conflicts between actionable and non-actionable user interface controls.

Workaround – You can check the availability of the document management system through the `gw.document.DocumentsActionsUIHelper.DocumentContentServerAvailable` property. Check this property as appropriate to improve the user experience when the document management system is unavailable.

Can promote an invalid rate book (PC-35087)

Issue – You can promote an invalid rate book to stage, approve it, and activate it. If a policy uses the invalid rate book for rating, PolicyCenter displays an error that the rate book is invalid.

Workaround – Create a new valid version of the invalid rate book that supersedes the existing rate book.

Rewrite new account of policy with locations having same address generates only one Account location (PC-35003, PC-35117)

Issue – Policies may have multiple locations that all have the same address. For example, commercial property policies may have locations with the same address, but these locations may be reinsured differently. In this case, if the policy is canceled and then rewritten to a new account, not all locations from the source/cancelled policy are generated correctly for the rewritten policy and its account. PolicyCenter generates only one Account location when the addresses are the same. Multiple policy locations are generated, but they all link to the same Account location.

Workaround – Guidewire is aware of this issue. Additionally, if the Copy Data feature is implemented, it can be used to copy the correct data from the original policy to the rewritten policy.

No user interface to set underwriting rule as externally managed (PC-35121)

Issue – In PolicyCenter 9.0.3, PolicyCenter added an `ExternallyManaged` property to the `UWIssueType` entity. Externally managed rules have their rule condition defined outside business rules, often in Gosu code. The underwriting **Rule** screen does not have a button to mark a rule as externally managed.

If you implement a new externally managed underwriting rule in Gosu code, you must define the `UWIssueType` properties in the underwriting **Rule** user interface. This interface does not have a button to mark the rule as externally managed.

Workaround – You can write code that sets the `ExternallyManaged` property to `true` on the `UWIssueType` entity instance for a rule. In the PolicyCenter user interface, you can add a button that accesses this code.

However, be careful about when the button is available. Rules cannot be toggled between `ExternallyManaged` and not `ExternallyManaged`. Make this button available to set `ExternallyManaged` to `true` on a brand new rule that has no information beyond the underwriting issue type fields defined.

Product Designer can insert empty XML tags that prevent rolling upgrade (PC-35123)

Issue – In PolicyCenter 9.0.3, Guidewire added the ability to make certain changes to product model patterns in a rolling upgrade of the servers in the PolicyCenter cluster. To support this change, Guidewire added a new configuration verification of the product model files to ensure that the rolling upgrade contains only valid (permitted) changes to the product model.

If you use Product Designer to modify a product model pattern created outside of Product Designer (for example, created in a release prior to PolicyCenter 8.0), Product Designer inserts empty XML tags for certain elements (`<AvailabilityScript/>`, for example). While these changes are generally legal to make in PolicyCenter, the upgrade configuration verifier flags the changes as illegal and thus blocks the rolling upgrade.

Workaround – Inspect any changes to product model pattern files (the `.xml` files) before you deploy them to production. If you discover new empty XML tags added by Product Designer, do one of the following:

• Remove the empty tags from the affected files before starting the rolling upgrade.

• Push the existing product model pattern XML files into production using a full database upgrade.

Cannot deploy new clauses with generic schedules in rolling upgrade (PC-35125)

Issue – It is not possible to deploy new coverage, condition, and exclusion patterns that include a generic schedule of items in a product model rolling upgrade of the servers in a PolicyCenter cluster. A generic schedule definition consists of two parts, the clauses and the definition of the schedule of items, defined in Gosu code. As Guidewire does not support the modification of existing Gosu code in a rolling upgrade, it is not possible to deploy a new schedule through a rolling upgrade.

Workaround – Deploy the new schedule in a full upgrade.

Need to restart PC server after adding a generic schedule (PC-35128)

Issue – After uploading a change from Product Designer that includes a generic schedule, PolicyCenter throws an `ErrantGosuClassException` exception. This was observed occurring during a submission policy transaction. The PolicyCenter server was in development mode.

Workaround – Restart PolicyCenter.

Reference constraint error when purging a merged account (PC-35400)

Issue – Data destruction causes the database to generate a reference constraint error when it purges a merged account. This occurs if:

- Reinsurance Management is enabled
- The merged accounts have `LocationRisk` objects

The foreign key from `LocationRisk` to `AccountLocation` is not updated when the accounts are merged.

Workaround – After the merge, go through all `LocationRisk` objects and call `setDenormalizedFK` method on `LocationRisk` to update the foreign key on `locationRisk`.

Purging account whose policy has been moved to another account generates exception (PC-35401)

Issue – PolicyCenter throws a runtime exception `Mismatch between Pinnables in instance graph vs. Pinnable hierarchy` when it attempts to purge an account that had a policy but the policy has been moved to another account. This is part of the data destruction feature. This occurs because the `JobGroup` has not been updated to point to the destination account.

Workaround – For all jobs on the source policy that are part of a `JobGroup`, add the job to a new `JobGroup` or add it to the current `JobGroup` on the destination `Account`. To affect all jobs on the policy, make this change in the `addToGroup` method in `submissionEnhancement` and `RenewalEnhancement`.

Upgrade to PolicyCenter after running ETL Product Model Loader extension pack (DOC-10285)

Issue – Upgrading to PolicyCenter 8.0.3 or later after running ETL Product Model Loader extension pack may require a manual drop of some tables.

In PolicyCenter 8.0.3, the ETL Product Model Loader Extension Pack functionality moved into the core PolicyCenter product. The extension pack created ETL database tables with `pcx_` as the prefix. PolicyCenter 8.0.3 and later create ETL database tables with `pc_` as the prefix. If you ran this extension pack with a prior version of PolicyCenter, you will have stale tables after upgrading to PolicyCenter 8.0.3 or later. The upgrade does not fail, but the server detects extra tables in the database.

Workaround – To remove these stale tables, manually drop the old `pcx_` tables created by the ETL Product Model Loader extension pack. The ETL Product Model Loader Extension Pack created these tables:

- `pcx_etlclausepattern`
- `pcx_etlcovtermpattern`
- `pcx_etlcovtermoption`
- `pcx_etlcovtermpackage`
- `pcx_etlpackterm`
- `pcx_etlmodifierpattern`
- `pcx_etlratefactorpattern`

You may also need to update your ETL code to reference the new tables.

Product Designer Known Issues

PolicyCenter does not start in single currency display mode if the product model is configured to use multiple currencies (PC-27829)

Issue – Product Designer allows users to configure multicurrency features without regard for the setting of the `MultiCurrencyDisplayMode` in the PolicyCenter `config.xml` file. If you configure multiple currencies in a product model object, and the value of the `MultiCurrencyDisplayMode` property is `SINGLE`, PolicyCenter does not start.

For example, if you add clauses that specify multiple coverage currencies and the PolicyCenter `MultiCurrencyDisplayMode` is set to `SINGLE`, the server fails to start and displays messages similar to the following:

```
ERROR ... PolicyLinePattern : ... [field : AvailableCoverageCurrencies] : - Multiple Coverage
Currencies defined for product not allowed in single Currency mode
ERROR ... [ExclusionPattern] : ... [DirectCovTermPattern] : ... [LimitsSet] : - The pattern specifies
limits for different currencies in SingleCurrencyMode
```

Workaround – If you intend to deploy a product model on PolicyCenter servers that are not configured for multiple currency display mode, do not configure multiple currencies in any part of the product model. However, you can test multicurrency products on a staging server configured for multicurrency while the production system is configured for a single currency.

System table cells display public IDs when referencing another system table (PC-28368)

Issue – If the cells in a particular column of a system table are referencing values from another system table, only the public IDs of those values are shown, not the values, when the system table is edited in Product Designer. In addition, users cannot search for values in these system table cells. The referenced system table, however, does not display the public ids so they cannot be viewed from within Product Designer.

Workaround – View the XML directly for the system table being referenced to determine the public IDs associated with the values that need to be added, edited or deleted. You can then use those public IDs directly.

An exception might appear when you run Product Designer (PC-28531)

Issue – An `IncompatibleRemoteServiceException` might appear if you run Product Designer 8.0.3 or later after running a previous version of Product Designer.

Workaround – Refresh the browser cache by selecting `Ctrl+F5`.

Invalid audit schedules when audit schedule properties file is changed while user is editing audit schedule (PC-28146)

Issue – If you edit an audit schedule in Product Designer at the time that the associated audit schedule properties file is updated, you may not see the properties file changes reflected in Product Designer. This could result in an invalid audit schedule (that matches the previous audit schedule properties file).

Workaround – Perform the following steps:
- In Product Designer, revert the changes to affected audit schedules.
- To load the audit schedule properties file changes into Product Designer, restart Product Designer, logout, or change the changelist.
- Reopen the affected audit schedules.

Studio/Platform Issues

Chrome browser cannot display product documentation in HTML format (DOC-7251)

Issue – If you use the Google Chrome browser, you can view the HTML Guidewire product documentation only if it is served by an HTTP server using the `http://` protocol. The Chrome browser cannot load HTML product documentation from your local disk by using the `file://` protocol.

Workaround – Use a different browser, such as Microsoft Internet Explorer. If you use the HTML documentation only to access the PDF files, you can open them directly in the `pdf` subdirectory of the `doc` directory.

Alignment Property Does Not Work for Inputs in a DetailView (PL-29429)

Issue – Inputs in a detail view do not align to the right or center. They always align to the left, regardless of the value of the `align` property.

Workaround – If you need to right align a series of items such as a set of monetary values, put them into a list view instead of a detail view. You can then set the alignment as needed.

JMS inbound integration implementations must be OSGi or Java plugins (PL-32054)

Issue – Writing JMS inbound integration handlers requires casting an object to the type `javax.jms.Message`. In Gosu, for some app servers this fails with the message: "Caused by: gw.lang.parser.exceptions.ErrantGosuClassException: GosuClass test.GosuJMSMessageReplyHandler has errors, and cannot be used at runtime."

Workaround – Implement JMS handler code for inbound integration in Java. Guidewire recommends that for inbound integration Java code, implement your code as an OSGi plugin, not a standard (non-OSGi) Java plugin.

Compilation errors after removing a delegate (PL-34619)

Issue – When you remove a delegate, some internal code is not generated properly, resulting in compilation errors.

Workaround – In Guidewire Studio, on the **Codegen** menu, click **Generate Metadata Classes**.

Several permissions are defined incorrectly (PL-34946)

Issue – The following permissions are defined incorrectly:

- `perm.Holiday.edit`
- `perm.Holiday.delete`
- `perm.Note.view`

Workaround – To any role that contains the `holidaymanage` permission, add the `buswkmanage` permission to that role. To any role that contains the `noteview` permission, add the `noteedit` permission to that role.

Alt+Shift+L to reload PCF files does not work when the server is run from the command line (PLWEB-5819)

Issue – If you run the server from the command line with the command `gwb runServer`, then pressing `Alt+Shift+L` in the application interface does not reload PCF files.

Workaround – Run the server from Guidewire Studio instead.

Server Tools downloads may time out (PLWEB-6184)

Issue – On the **Server Tools** pages, some downloads may take too long and generate a timeout error.

Workaround – Increase the value of the `WebUIAJAXTimeout` configuration parameter.

During rule import, selecting the deployed rule results in incorrect rule version and status (BIZ-912)

Issue – Under certain circumstances, the business rule import process sets the rule version incorrectly. This occurs in the following circumstances:

- The existing and importing rules show as a version conflict.
- The existing or importing rule version is in the deployed state.
- The other rule version has a status other than deployed (draft, staged, approved).
- The deployed version of the rule is selected.

In all of these cases, the rule version is set to Approved with the deployed rule version number. For example, the deployed version is 1 and the version after import is 1+ Approved.

Workaround – Guidewire is aware of this issue.

DBCC scan reports obfuscated users (BC-16712)

Issue – In the base configuration, personal data destruction of a user results in the obfuscation of the `User` entity and the removal of the linked `UserSettings` entity from `User`. A subsequent database consistency check reports a constraint violation for such users because it is unable to find the `UserSettings` entity.

The issue is reported in BC-16712, but it applies to all Guidewire InsuranceSuite applications.

Workaround – If you do not enable personal data destruction, this constraint violation will not occur. If you do enable personal data destruction, you can ignore these constraint violations reported by the database consistency checking tool DBCC.

Guidewire PolicyCenter 9.0.3 Release Notes

Release 9.0.3

Overview

These release notes contain the following topics:

Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

> **IMPORTANT** If you skipped one or more upgrade releases to PolicyCenter, be sure to read the Release Notes for those releases to learn about changes and fixed issues.

Version Number

This release of Guidewire PolicyCenter is 9.0.3.
- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 8.0.1 or later. The latest maintenance release is preferred.

  If you are implementing multiple settlement currencies, BillingCenter 8.0.2 or later is preferred.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 8.0.0 or later. The latest maintenance release is preferred.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 8.0.1 or later. The latest maintenance release is preferred.

Installing This Release

Refer to the following:

- The *PolicyCenter Installation Guide* for general installation information
- Prior PolicyCenter release notes for any versions that you have skipped

Upgrade Information

Significant Changes to Upgrade Procedure

IMPORTANT The procedure for upgrading to PolicyCenter 9 from PolicyCenter 8 or previous releases has changed significantly. In PolicyCenter 9, there are new tools available for performing the upgrade. Do not rely on your past knowledge of the upgrade process to begin the upgrade procedure. Before beginning your upgrade, review the topic "Configuration Upgrade Overview" in the *PolicyCenter Upgrade Guide*.

Exception when upgrading UWIssue objects in sample data from 7.0.x (PC-34046)

When upgrading from PolicyCenter 7.0.x, database upgrade throws an `IllegalArgumentException` exception when encountering `UWIssue` objects in the sample data. The exception occurs because the upgrade expects to find typelist values added in PolicyCenter 8.0.x.

To workaround this issue, delete `UWIssue` objects in the sample data from the PolicyCenter 7.0.x database or add the typelist values necessary.

Support

For assistance, visit the Guidewire Community.

Guidewire Customers

- `https://community.guidewire.com`

Guidewire Partners

- `https://partner.guidewire.com`

Major Issues and Changes

This topic contains major issues and changes that can affect your installation.

New and Changed Features

For information on new features and other changes, see the topic "New and Changed in PolicyCenter 9.0.3" in the *PolicyCenter New and Changed Guide*.

New Studio Options for Managing Application Upgrades (IDE-3811, IDE-3960)

In the Studio **Settings** dialog, on the **Gosu Compiler** screen, there are new options for helping you manage application upgrades more efficiently. You can disable local Gosu compilation, exclude compilation of tests, and treat PCF code generation errors as warnings. See the *PolicyCenter Configuration Guide*.

Improve Build Performance by Preprocessing XML Schema Files (PL-36353)

This release includes significant improvements to build times. In addition to the many built-in improvements, build time can be further reduced by preprocessing XML schema files and certain other resources. The preprocessed resources are stored in a JAR file, which is used in subsequent builds without the need to be reprocessed.

For details on integrating this feature in the build process, visit the Guidewire Community and search for knowledge article 7356, "Generating an XML Schema Jar".

Base PCF File Changes

The following link requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 9.0.2 to 9.0.3

To view a report of the changes to the base PCF files, `click here`.

Base Resource Changes

The following link requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 9.0.2 to 9.0.3

To view a report of the changes to the base resources, `click here`.

Changes in This Release Provided in Upgrade Diff Report

Guidewire provides a report detailing certain differences between the current release and your prior release. This report describes changes in display keys, entities, typelists, and the Gosu API. To obtain your custom Upgrade Diff Report, visit the Guidewire Community.

Improvements and Resolved Issues

This topic describes improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This list is not intended to be comprehensive.

Application Improvements and Resolved Issues

ID	Description
Activities	
DE1592 DE1823	Fixed issues when creating a new activity on the **Activity** worksheet. The **Assign To** search did not show all groups the user had access to. The root group was shown in **Assign To** when creating an activity.
Administration	
PC-34712	Fixed an issue where you could create an external user on the carrier organization. Added user interface validation to prevent you from selecting the carrier organization when the **External User** radio button is set to **Yes**. If the carrier organization is set and **External User** is **Yes**, then the update is rejected. Although the user interface does not permit it, added validation that an internal user cannot belong to organization other than their organization.
PC-35008	Added checks to ensure that a user can only be added to a group once, and a group can only be added to a user once.
Business Rules	

ID	Description
PC-34652	For consistency with ClaimCenter, you can now access utility functions by typing `Util` in formula expressions in business rules. This is equivalent to `util`.
PC-34942	Fixed a previously known issue where externally managed rules were shown as invalid in the **Underwriting Rules** screen. Externally managed rules are now identified as such in the **State** column. Externally managed rules have their rule condition defined outside business rules, often in Gosu code.
PC-34943	For externally managed underwriting rules, the Underwriting Rule screen only displays fields related to underwriting issue type. Externally managed rules have their rule condition defined outside business rules, often in Gosu code.
Contacts	
PC-26497	Fixed an issue in the PolicyCenter to ContactManager integration which resulted in a `PLDataTypeException` and suspended the message queue. With this fix, an activity description that is longer than the size of the description field is truncated.
Contingencies	
PC-33731	Fixed a previously known issue where the **Contingency** screen showed hidden documents. With this fix, hidden documents are not shown by default. Added a **Show Included** button to view hidden documents.
PC-34412	Fixed a previously known issue on a contingency, where deleting a document whose contents were replaced using upload gave the error "Object you are trying to update was changed by another user."
Data Model / Core Entities	
PC-34961	To improve the speed searching for policy transactions (referred to as jobs in the data model), added `JobNumberSearch` index to the `Job` entity. However, this change increases upgrade time. If quicker upgrade is more important than quicker policy transaction search, then you can remove the `JobNumberSearch` index from `Job.etx`.
General	
PC-32472	In `SpecialtyType.ttx`, removed trailing space from description for `chiropractic` typecode.
PC-33721	When the `IDocumentMetatSource` plugin is enabled, documents uploaded to a contingency are now visible in the **Contingency→Documents** list.
PC-33734	Fixed an issue where the user was able to edit or delete a final document even without the `DOCEDIT` and `DOCDELETE` permissions. This applies when `PLConfigParameters.FinalDocumentsNotEditable` is `true` and the `IDocumentMetadataSource` and `IDocumentContentSource` plugins are enabled.
PC-33738	With `IDocumentContentSource` and `IDocumentMetadataSource` enabled, fixed an issue where two submission letters were created instead of one.
PC-34374	Made changes to the commercial auto submission "Clone Selected" button so that when clicked, it makes copy of each selected vehicle and the vehicle list is updated on UI
PC-34683	Added `ExternalUserAccess` parameter to `config.xml` to indicate which groups and producer codes are available to external users. Values are: • `FULLYRESTRICTED` – external users can access groups and producer codes in the same organization • `PROTECTINTERNAL` – external users can access all external groups and producer codes • `ALLOWINTERNAL` – external users can access all groups and producer codes, both internal and external Default is `FULLYRESTRICTED`.
PC-35005	Fixed an issue where `AssignableImpl#assignGroupByRoundRobin` failed to assign to a group if the current group does not have sub-groups
I18N / G11n / Lang Packs	
PC-34093	In the businessowners line, replaced abbreviations and phrases in mixed case strings in coverage pattern display keys with standard English phrases. For example, `RetroactiveDate` is now `Retroactive Date`, and `CondoMiscPropDed` is now `Condo Miscellaneous Property Deductible`.
PC-34962	Fixed an issue which caused a `ClassNotFound` exception when getting localized `PolicyContactRole` display name.

ID	Description
Integration	
PC-34286	`RIRisk`, `RIProgram`, `ProgramTreaty`, and `RIAgreement` entity instances now all generate events if they change. Also, some foreign keys are now marked as `ignoreforevents="true"` so that typically, only one entity is reported as changed. For example, if you change `RIAttachmentInclusion`, only `RIRisk` is reported as changed, not `RIRisk`, `RIAgreement`, and `PolicyPeriod`. These additional events enable more granularity in Gosu rules when a reinsurance entity instance updates. These changes improve performance when these entities persist to the database.
PC-34848	Fixed an issue in the `PersonalVehicle` entity definition, `StatedValue` field. The type of `StatedValue` is now defined as `gw.pl.currency.MonetaryAmount` in `PersonalVehicleModel.gx`, matching the definition of `StatedValue` in the `PersonalVehicle` entity.
Job Infrastructure	
PC-33826	PolicyCenter now checks user permissions when retrieving documents attached to a contingency.
PC-34866	PolicyCenter now checks user permissions when retrieving notes attached to contingency.
PC-34880	Fixed an issue where policy period validation results could be not initialized when side by side process started. This issue caused a concurrent data exception.
Performance Improvement	
PC-34134	Improved performance during the quote process and other code that calls `PolicyPeriod#getAllOOSDates` method. With this fix, the out-of-sequence date cache is only cleared if the newly added date is not in the cache.
PC-34896	The signatures of the `RateRoutineValidation#validateCalcStepInScopeUsage` and `RateRoutineValidation:validateInScopeUsage` methods now take a nested map of `CalcRoutineParamName` for the `InScopeUsage` map. Refactored rate routine validation logic for improved performance.
PC-35115	Fixed a memory leak which occurred when rating many policy transactions with rating worksheets enabled.
Policy File	
PC-34102	You can now transfer policies with attached final documents between accounts, provided you have appropriate permissions for performing the transfer.
PC-34554	When a policy renewal is in draft and the user goes from the renewal job wizard to the policy file summary, the as-of date is now set to effective date of the base policy period.
Product Model	
PC-34827	Added checking for coverage term availability on offerings for conditions and exclusions.
PC-34849	Product model code generator now scans a specific subset of folders holding the generated source files instead of the entire output folder.
Quote purging	
PC-35084	Fixed an issue where quote purging did not purge some policy periods eligible for purging.
Rating	
PC-30084	The rate routine editor has been improved so that this previously known issue does not occur. This issue was that the rate routine editor did not display a message when the rate table definition parameter was modified.
Reinsurance	
PC-34851	Fixed a rounding issue in reinsurance where a scale of 2 for calculating percentages produced incorrect results for large values. With this fix, a scale of 6 is used for calculating percentages.
Search	

ID	Description
PC-25541	Fixed an issue on the **Notes** screen where **Related To** returned notes from accounts other than the account of the current submission. On the **Notes** search screen, removed **<none>** from the **Related To** field. This field is initialized to the first of possible choices.
UI Configuration	
PC-34536	Fixed an issue in side-by-side quoting which caused an `IllegalStateException` after clicking **View All** in the **Risk Analysis - UW issues** screen.

Platform Improvements and Resolved Issues

The following are the primary improvements and issues corrected in this release:

ID	Description
AppServer Support - JBoss	
PL-36302	Added support for JBoss Enterprise Application Platform version 7.0.2.
Build Infrastructure	
PL-36368	After adding a new GX model, WSDL, or XSD file, the subsequent compilation process has been optimized. The performance gain is achieved by significantly reducing the number of files processed during the compilation.
Database Support - Oracle	
PL-36274	Added the following option to the list of options to use in generating a set of performance reports from the Server Tools Oracle AWR Information screen: **Include native Oracle report**. This option ensures that PolicyCenter generates an Oracle Standard AWR report automatically as you generate a PolicyCenter AWR report, if the database user has been granted the required privileges. See "Oracle AWR" in the *System Administration Guide* for details.
Document Management	
PL-36424	When a document is retrieved from the document management system, the resulting `Document` object must include the retrieved metadata. Implementations of the Document Metadata Source plugin method `retrieveDocument` must explicitly restore the document's original internal values by calling the `DocumentsUtilBase.initOriginalValues` method. The original internal values are evaluated by subsequent permission checks. Permission handlers evaluate properties to determine whether the user has the appropriate permission to access the document. For example, an `Edit` handler checks the `Document` object's `Status` property and prohibits editing if the status is final.
GX Tools	
PL-36434	The `gwb verifyResources` command now verifies GX model files, workflow XML files, and WS-I web service annotations, as well as PCF files, Gosu types, and XML schemas.
Gradle Build Infrastructure	
PL-36401	You can now adjust memory settings for various command-line processes in the `gradle.properties` file. See "Tune Command Line Tool Memory Settings" in the *Installation Guide*.
PL-36511	Fixed an issue where generating a WAR file would produce an invalid file if any of the files in the configuration had names containing non-ASCII characters.
PL-36550	The `gwb studio` command automatically runs code generation first if it has not yet been run.
Inbound Integration	
PL-34046	Fixed an issue that resulted in an error when changing the server run level from `maintenance` to `multiuser`.
Integration	

ID	Description
PL-34510	In the `ImportResults.pcf` file, the value of the `canVisit` property has been changed from `perm.User.exportadmindata` to the correct `perm.User.importadmindata`.
Other	
PLWEB-6287	Replaced several hard-coded strings in JavaScript files with display keys.
PCF - Actions - Auto Complete	
PLWEB-6286	Fixed an issue where you could not type into a TaxID or SSN text box after navigating from a field that used auto-completion.
PCF - Layout - List View - Iterator	
PLWEB-6261	Fixed an issue where exporting CSV data from a `LinkIterator` widget would produce empty values.
Rolling Upgrade	
PL-36406	Fixed an issue that caused rolling upgrade checksum verification to take longer than necessary on server start-up.
Web - PCF Compiler	
PLWEB-6263	Fixed an issue where the PCF compiler would not show an error when column level attributes were assigned row level values.
Web - PCF Compiler - Studio	
PLWEB-6269	Fixed an issue where PCF files would not reload when there was a compilation error in any Gosu code that was loaded as part of the page.

Known Issues and Limitations

This topic describes known issues with this release of Guidewire PolicyCenter.

PolicyCenter Known Issues

Provide a way to filter out added and removed subentities in a policy differences tree (PC-7929)

Issue – Need a way to filter out the children of an added or removed entity in a policy differences tree.

Workaround – Guidewire is aware of this issue.

When using parallel sessions, approvals made by underwriter are not updated for second user simultaneously viewing same work order (PC-12322)

Issue – Approvals made by an underwriter are not updated for a second user simultaneously viewing the same work order. This occurs in the following sequence of events:

- An agent processes a submission and issues arise
- The agent calls an underwriter, and remains in the policy
- The underwriter approves the issue or issues
- The agent refreshes the **Risk Analysis** screen by going to a different wizard step and returning to the **Risk Analysis** screen

After this sequence of events:

- The new approvals do not appear on the agent's screen
- However, the agent can bind or issue the policy (based on the new approvals by the underwriter)

Workaround – The agent can leave the policy and return to it. The approvals appear as expected. Alternatively, the agent can see the approvals by clicking the **Details** button when issuing the policy.

Must enter VIN when moving from quick quote multi-version or side-by-side to full application (PC-13099)

Issue – In a quick quote, you do not have to enter the VIN or other fields that are required for a full application submission. When converting a quick quote with multi-version or side-by-side quoting to full application, to resolve validation issues, you must enter the VIN for each of the versions. In a job with side-by-side quoting, this issue can occur with fields that are not base data.

Workaround – Add code that copies this type of data to the other versions.

PolicyCenter does not send renewal billing instruction to BillingCenter (PC-15082)

Issue – On receiving payment notification from BillingCenter, PolicyCenter tries to bind. If the policy has a renewal transaction in draft status, PolicyCenter does not bind and send the Renewal Billing Instruction to BillingCenter. The policy might be in draft status after an underwriter edited something on the policy. This is an issue with BillingCenter 3.0.1 and later.

Workaround – Guidewire is aware of this issue.

Renewal processing for renewal conversion assumes BasedOn is populated (PC-15851)

Issue – The `startNewRenewal` method in `gw.webservice.pc.pc700.job.PolicyRenewalAPI` assumes the existence of a prior term or a `basedOn` version. Unless customized, this method can cause problems in certain convert-on-renewal scenarios. This method has been deprecated.

Workaround – If you are developing convert-on-renewal flows, use the `gw.webservice.pc.pc700.job.startConversionRenewal` method.

SQL Query performance in free-text search (PC-18794)

Issue – There are open issues related to the performance of SQL query in free-text search. This occurs the first time that you load the PolicyCenter data into the Solr index database.

Workaround – Guidewire is aware of this issue.

Cannot quote a renewal if the expiring policy term has a handled preemption (PC-19416)

Issue – In Commercial Package Policy in the base configuration, if a policy term has a preemption that has been handled, when a renewal job is created, the **Edit Workorder** button is not always available on the renewal. When this occurs, the only options are to bind or not take the renewal.

Workaround – Guidewire is aware of this issue.

Copy-data with an out-of-range date (PC-20279)

Issue – When working with the copy data feature, if you specify a specific date to copy data from and that date is out of range for the policy, PolicyCenter presents a warning that no applicable policy version could be found. However, the date selector still indicates the requested date, not the date that PolicyCenter will use for the copy if you choose to proceed.

Workaround – As appropriate, modify the user interface to reset the date selector or to hide the ability to select data to copy.

Gosu classes not intended for use (PC-23009)

Issue – Some Gosu classes are visible in Studio, but are not intended for use. You can distinguish these Gosu classes because they have no visibility annotations (neither `@Export` nor `@ReadOnly`) and they are not in a `gw` package.

Workaround – Do not use these methods in configuration. The methods are unsupported and may change or be withdrawn without notice.

Unbound renewal policy transactions can provide invalid dates (PC-25288)

Issue – If a policy renewal is in the renewal process and the user goes from the renewal job wizard to the policy file summary, the as-of date default in the **Date** field of the **Actions** panel will not be entirely valid, as it will be the

`editEffectiveDate` of an unbound period from an incomplete job.The quote, reinsurance, and audit schedule, if enabled, screens display exceptions because appropriate periods do not exist on the specified dates.

Workaround – Edit `JobWizardInfoBar.pcf` to filter out any `PolicyPeriod` objects that cannot provide the `editEffectiveDate`.

Billing integration error when final audit is waived from Audit job (PC-25500)

Issue – When the user waives a final audit from within a running Audit job, the BillingCenter integration sends an incomplete message to BillingCenter. The message does not include the term number of the policy period. BillingCenter needs the term number to identify the BillingCenter policy period that needs the change.

Workaround – Guidewire is aware of this issue.

Cannot change availability tables columns after data added (PC-26836)

Issue – The Product Model Guide provides instructions on how to add columns to an availability lookup table. However, after loading availability data, you cannot add or remove columns.

Workaround – Guidewire is aware of this issue.

Cannot use Product Designer to edit an extended object with no base object (PC-27763)

Issue – If you attempt to edit a part of the product model that is based on an entity extension file (`.etx`), and that extension has no base entity file (`.eti`), Product Designer displays errors and refuses to display the contents of the file.

Although Guidewire best practices recommend that you never create an entity file extension (`.etx` file) when no base entity (`.eti` file) exists, Guidewire Studio did not enforce this recommendation. However, the current version of Product Designer does not allow the use of an entity extension without a corresponding entity base file. If such an entity extension exists, Product Designer displays a **NullPointerException** when you attempt to access the affected objects. You then can use the Back command in the browser to view other pages, but you cannot configure the affected objects.

Workaround – Locate all `.etx` files that do not have corresponding `.eti` files, and rename them with an `.eti` extension.

Payment screen not updated (PC-30594)

Issue – Updating a payment plan in BillingCenter does not update **Premium Summary**, **Payments**, and **Invoicing** fields on the PolicyCenter **Payment** screen. Even making a selection in the payment plan list view does not update these fields.

Workaround – To refresh these fields, navigate away from and return to the **Payment** screen.

PolicyPeriod.removeFromForms method causes DBCC error (PC-30947)

Issue – The `PolicyPeriod.removeFromForms` method causes a DBCC error.

Workaround – Guidewire is aware of this issue. Do not use this method.

List bill with no matching invoice stream for a payment plan generates WSDL fault (PC-31004)

Issue – This issue occurs when a list bill account does not have a compatible invoice stream for one of the payment plans associated with the list bill account. When adding that list bill account to the policy, PolicyCenter throws an exception:

```
WsdlFault: java.lang.IllegalStateException: Cannot lazily create an invoice stream for charge with
owner {policy period id} on list bill account {account number}
```

Workaround – All payment plans must have compatible invoice streams configured in the billing system.

Comparing policy transactions does not work correctly with out-of-sequence transactions (PC-31190)

Issue – When comparing two policy transactions from the Policy Transactions screen, any changes included in the policy revisions being compared that had been made effective on a later date than the edit effective date of the

associated transaction (either within that transaction or in a previous transaction) are not displayed in the comparison. Changes made on earlier dates are included.

Workaround – Guidewire is aware of this issue.

Final audit setting changes if billing method changed in BillingCenter (PC-31360)

Issue – On renewal, final audit is changed from **No** to **Determined by business rule** when the billing method and payment plan have been changed in BillingCenter before starting the renewal. This also occurs with rewrite.

Workaround – Manually reset **Final Audit** to **No**.

Cannot complete unowned activities (PC-31610)

Issue – Cannot complete unowned activities even if you have the correct permissions. This issue is in PolicyCenter.

Workaround – Guidewire is aware of this issue.

Number of payment installments incorrect after reinstatement (PC-32171)

Issue – When integrated with BillingCenter, the number of payment installments may be incorrect after a reinstatement policy transaction. This affects general liability policies with final audit set to true after the policy is canceled. If the policy is reinstated, then the number of installments displayed may be incorrect. The installments in the billing system will still be calculated correctly.

Workaround – Guidewire is aware of this issue. The transaction can be completed without side effects.

Message events against policy term archived in BillingCenter generates error (PC-33320)

Issue – If a policy period is archived in BillingCenter but not in PolicyCenter, that policy can be transferred to a new account owner in PolicyCenter. However, this causes an error in BillingCenter. The transfer is shown as successful in PolicyCenter, but the message queue includes this error: "The operation cannot be performed on an archived PolicyPeriod. The PolicyPeriod that is the target of the operation is archived." BillingCenter does not support transfer to a new account on archived policies or policies retrieved from archive.

Workaround – Guidewire is aware of this issue.

Can issue policy change against policy period retrieved from archive in BillingCenter (PC-33339)

Issue – If a policy period has been retrieved from the archive in BillingCenter and the policy period in BillingCenter is still closed, then you should not be allowed to issue a policy change in PolicyCenter. However, PolicyCenter allows you to issue a policy change. BillingCenter rejects the policy change message and sends an error to PolicyCenter. The message remain in an unsent state in the message queue.

Workaround – Reopen the policy period in BillingCenter and resend the message. This workaround requires that BC-15815 be fixed.

Waiving final audit or premium report throws BadIdentifierException (PC-33351)

Issue – If you start a final audit and select **Waive** as the close option, PolicyCenter throws a `BadIdentifierException`. The waive final audit message is not sent to BillingCenter. This also affect premium reports.

Workaround – Waive the final audit from the **Audit Schedule** screen.

Availability limitations on schedule coverages (PC-34620)

Issue – Setting availability criteria on scheduled item coverages results in a null pointer exception when viewing policy differences.

Workaround – When configuring schedule coverages:

- Availability is supported on the schedule coverage and its coverage terms (for example, Scheduled Personal Property).
- Availability is supported on the scheduled item coverage pattern options.
- Availability is not supported for the scheduled item coverage and coverage terms (for example, Scheduled Personal Property Item). Set these types of clauses to always available. These clauses inherit their availability from the schedule coverage. That is, the scheduled item coverage is available whenever the schedule coverage is available.

Coverage Term Option Value Providers not working in Rating (PC-34695)

Issue – If a value provider of coverage term options is created for a rate table column, it will not return any values. This problem only manifests for coverages and coverage terms that were created in 9.0.0 or later.

Workaround – Do not use a value provider for the associated column and enter values directly into the column.

Changes to Availability of existing Product Model Patterns become active immediately in rolling upgrade (PC-35054)

Issue – Changes to availability of existing product model patterns during rolling upgrade of the PolicyCenter cluster become active as soon as the first updated node restarts with the new configuration. For example, suppose that you want to end-date the availability of an existing coverage term pattern and replace it with a newly added coverage term pattern. In that case, you need to be aware that the end-dating becomes active immediately, even though the new coverage term is not available until rolling upgrade fully completes. This can create challenges during the rolling upgrade window as neither coverage term is available during that time. This is true of any product model pattern types that have Availability logic.

Workaround – Consider a two-phase rollout of the change to the product model pattern availability:

1. First, deploy the new product model pattern in a rolling upgrade and complete that upgrade.
2. Then, do one of the following:
   - Perform a second rolling upgrade of the product model and make the old product model pattern unavailable.
   - Use the Server Tools Reload Availability functionality to make the old product model pattern unavailable.

Do not use the Reload Availability functionality during a Rolling Upgrade (PC-35067)

Issue – PolicyCenter provides the ability to reload product model availability data through the Server Tools **Product Model Info** screen. Do not activate this functionality during a rolling upgrade of a PolicyCenter cluster as it can lead to potential conflicts with changes being made as part of the rolling upgrade.

Workaround – Wait until after the rolling upgrade completes to use the **Reload Availability** functionality.

In side-by-side comparison of business rules, PolicyCenter displays fields not available to externally managed rules (PC-35069)

Issue – When comparing externally managed business rules on the **Compare Rules** screen, PolicyCenter displays fields that are not used by externally managed rules. Only fields related to the underwriting issue type are used by externally managed rules, but all Underwriting **Rule** fields are displayed (for example, the **Rule Condition**).

Workaround – Guidewire is aware of this issue.

Externally managed underwriting rules appear to be invalid in Review Import screen (PC-35071)

Issue – When you import an externally managed rule into PolicyCenter, it is marked as an invalid rule rather than an externally managed rule. On the **Business Rules** -> **Import/Export Status** -> **Review Import** screen, the rule is marked as invalid. Unlike the **Underwriting Rules** screen, this screen lacks a **State** column to indicate externally managed rules.

Workaround – Guidewire is aware of this issue.

Product Model rolling upgrade does not support changes to existing Offerings (PC-35072)

Issue – PolicyCenter does not allow the addition of a selection for a new Product Model pattern (for example, a new coverage or coverage term option) to an existing Offering during a Product Model rolling upgrade. Guidewire restricts this functionality due to potential data loss.

Workaround – Guidewire is aware of this issue.

Issues with user interface elements when configured document management system not available (PC-35082)

Issue – Certain user interface controls show inappropriate status or are inexplicably disabled when a configured document management system is not available. The user interface is generally responsive to the presence of a document management system. However, if that system is unavailable, you may see puzzling user interface state or behavior. This may include usually actionable controls unavailable, actionable controls that inexplicably do not take action, or conflicts between actionable and non-actionable user interface controls.

Workaround – You can check the availability of the document management system through the `gw.document.DocumentsActionsUIHelper.DocumentContentServerAvailable` property. Check this property as appropriate to improve the user experience when the document management system is unavailable.

User without required permissions can delete rating objects (PC-35085)

Issue – Any user, internal or external, can delete rate tables, rate books, rate table definitions, rate routines, and parameter sets without having the necessary permissions.

Workaround – Change the **Delete** button for each of these rating entities to only be available to users with the Edit rate books and tables permission.

Can promote an invalid rate book (PC-35087)

Issue – You can promote an invalid rate book to stage, approve it, and activate it. If a policy uses the invalid rate book for rating, PolicyCenter displays an error that the rate book is invalid.

Workaround – Create a new valid version of the invalid rate book that supersedes the existing rate book.

Rewrite new account of policy with locations having same address generates only one Account location (PC-35003, PC-35117)

Issue – Policies may have multiple locations that all have the same address. For example, commercial property policies may have locations with the same address, but these locations may be reinsured differently. In this case, if the policy is canceled and then rewritten to a new account, not all locations from the source/cancelled policy are generated correctly for the rewritten policy and its account. PolicyCenter generates only one Account location when the addresses are the same. Multiple policy locations are generated, but they all link to the same Account location.

Workaround – Guidewire is aware of this issue. Additionally, if the Copy Data feature is implemented, it can be used to copy the correct data from the original policy to the rewritten policy.

No user interface to set underwriting rule as externally managed (PC-35121)

Issue – In PolicyCenter 9.0.3, PolicyCenter added an `ExternallyManaged` property to the `UWIssueType` entity. Externally managed rules have their rule condition defined outside business rules, often in Gosu code. The underwriting **Rule** screen does not have a button to mark a rule as externally managed.

If you implement a new externally managed underwriting rule in Gosu code, you must define the `UWIssueType` properties in the underwriting **Rule** user interface. This interface does not have a button to mark the rule as externally managed.

Workaround – You can write code that sets the `ExternallyManaged` property to `true` on the `UWIssueType` entity instance for a rule. In the PolicyCenter user interface, you can add a button that accesses this code.

However, be careful about when the button is available. Rules cannot be toggled between `ExternallyManaged` and not `ExternallyManaged`. Make this button available to set `ExternallyManaged` to `true` on a brand new rule that has no information beyond the underwriting issue type fields defined.

Product Designer can insert empty XML tags that prevent rolling upgrade (PC-35123)

Issue – In PolicyCenter 9.0.3, Guidewire added the ability to make certain changes to product model patterns in a rolling upgrade of the servers in the PolicyCenter cluster. To support this change, Guidewire added a new configuration verification of the product model files to ensure that the rolling upgrade contains only valid (permitted) changes to the product model.

If you use Product Designer to modify a product model pattern created outside of Product Designer (for example, created in a release prior to PolicyCenter 8.0), Product Designer inserts empty XML tags for certain elements (`<AvailabilityScript/>`, for example). While these changes are generally legal to make in PolicyCenter, the upgrade configuration verifier flags the changes as illegal and thus blocks the rolling upgrade.

Workaround – Inspect any changes to product model pattern files (the `.xml` files) before you deploy them to production. If you discover new empty XML tags added by Product Designer, do one of the following:

- Remove the empty tags from the affected files before starting the rolling upgrade.
- Push the existing product model pattern XML files into production using a full database upgrade.

Cannot deploy new clauses with generic schedules in rolling upgrade (PC-35125)

Issue – It is not possible to deploy new coverage, condition, and exclusion patterns that include a generic schedule of items in a product model rolling upgrade of the servers in a PolicyCenter cluster. A generic schedule definition consists of two parts, the clauses and the definition of the schedule of items, defined in Gosu code. As Guidewire does not support the modification of existing Gosu code in a rolling upgrade, it is not possible to deploy a new schedule through a rolling upgrade.

Workaround – Deploy the new schedule in a full upgrade.

Need to restart PC server after adding a generic schedule (PC-35128)

Issue – After uploading a change from Product Designer that includes a generic schedule, PolicyCenter throws an `ErrantGosuClassException` exception. This was observed occurring during a submission policy transaction. The PolicyCenter server was in development mode.

Workaround – Restart PolicyCenter.

Upgrade to PolicyCenter after running ETL Product Model Loader extension pack (DOC-10285)

Issue – Upgrading to PolicyCenter 8.0.3 or later after running ETL Product Model Loader extension pack may require a manual drop of some tables.

In PolicyCenter 8.0.3, the ETL Product Model Loader Extension Pack functionality moved into the core PolicyCenter product. The extension pack created ETL database tables with `pcx_` as the prefix. PolicyCenter 8.0.3 and later create ETL database tables with `pc_` as the prefix. If you ran this extension pack with a prior version of PolicyCenter, you will have stale tables after upgrading to PolicyCenter 8.0.3 or later. The upgrade does not fail, but the server detects extra tables in the database.

Workaround – To remove these stale tables, manually drop the old `pcx_` tables created by the ETL Product Model Loader extension pack. The ETL Product Model Loader Extension Pack created these tables:

- `pcx_etlclausepattern`
- `pcx_etlcovtermpattern`
- `pcx_etlcovtermoption`
- `pcx_etlcovtermpackage`
- `pcx_etlpackterm`
- `pcx_etlmodifierpattern`
- `pcx_etlratefactorpattern`

You may also need to update your ETL code to reference the new tables.

Product Designer Known Issues

PolicyCenter does not start in single currency display mode if the product model is configured to use multiple currencies (PC-27829)

Issue – Product Designer allows users to configure multicurrency features without regard for the setting of the `MultiCurrencyDisplayMode` in the PolicyCenter `config.xml` file. If you configure multiple currencies in a product model object, and the value of the `MultiCurrencyDisplayMode` property is `SINGLE`, PolicyCenter does not start.

For example, if you add clauses that specify multiple coverage currencies and the PolicyCenter `MultiCurrencyDisplayMode` is set to `SINGLE`, the server fails to start and displays messages similar to the following:

```
ERROR ... PolicyLinePattern : ... [field : AvailableCoverageCurrencies] : - Multiple Coverage
Currencies defined for product not allowed in single Currency mode
ERROR ... [ExclusionPattern] : ... [DirectCovTermPattern] : ... [LimitsSet] : - The pattern specifies
limits for different currencies in SingleCurrencyMode
```

Workaround – If you intend to deploy a product model on PolicyCenter servers that are not configured for multiple currency display mode, do not configure multiple currencies in any part of the product model. However, you can test multicurrency products on a staging server configured for multicurrency while the production system is configured for a single currency.

System table cells display public IDs when referencing another system table (PC-28368)

Issue – If the cells in a particular column of a system table are referencing values from another system table, only the public IDs of those values are shown, not the values, when the system table is edited in Product Designer. In addition, users cannot search for values in these system table cells. The referenced system table, however, does not display the public ids so they cannot be viewed from within Product Designer.

Workaround – View the XML directly for the system table being referenced to determine the public IDs associated with the values that need to be added, edited or deleted. You can then use those public IDs directly.

An exception might appear when you run Product Designer (PC-28531)

Issue – An `IncompatibleRemoteServiceException` might appear if you run Product Designer 8.0.3 or later after running a previous version of Product Designer.

Workaround – Refresh the browser cache by selecting `Ctrl+F5`.

Invalid audit schedules when audit schedule properties file is changed while user is editing audit schedule (PC-28146)

Issue – If you edit an audit schedule in Product Designer at the time that the associated audit schedule properties file is updated, you may not see the properties file changes reflected in Product Designer. This could result in an invalid audit schedule (that matches the previous audit schedule properties file).

Workaround – Perform the following steps:

- In Product Designer, revert the changes to affected audit schedules.
- To load the audit schedule properties file changes into Product Designer, restart Product Designer, logout, or change the changelist.
- Reopen the affected audit schedules.

Studio/Platform Issues

Chrome browser cannot display product documentation in HTML format (DOC-7251)

Issue – If you use the Google Chrome browser, you can view the HTML Guidewire product documentation only if it is served by an HTTP server using the `http://` protocol. The Chrome browser cannot load HTML product documentation from your local disk by using the `file://` protocol.

Workaround – Use a different browser, such as Microsoft Internet Explorer. If you use the HTML documentation only to access the PDF files, you can open them directly in the `pdf` subdirectory of the `doc` directory.

Alignment Property Does Not Work for Inputs in a DetailView (PL-29429)

Issue – Inputs in a detail view do not align to the right or center. They always align to the left, regardless of the value of the `align` property.

Workaround – If you need to right align a series of items such as a set of monetary values, put them into a list view instead of a detail view. You can then set the alignment as needed.

JMS inbound integration implementations must be OSGi or Java plugins (PL-32054)

Issue – Writing JMS inbound integration handlers requires casting an object to the type `javax.jms.Message`. In Gosu, for some app servers this fails with the message: "Caused by: gw.lang.parser.exceptions.ErrantGosuClassException: GosuClass test.GosuJMSMessageReplyHandler has errors, and cannot be used at runtime."

Workaround – Implement JMS handler code for inbound integration in Java. Guidewire recommends that for inbound integration Java code, implement your code as an OSGi plugin, not a standard (non-OSGi) Java plugin.

Compilation errors after removing a delegate (PL-34619)

Issue – When you remove a delegate, some internal code is not generated properly, resulting in compilation errors.

Workaround – In Guidewire Studio, on the **Codegen** menu, click **Generate Metadata Classes**.

Several permissions are defined incorrectly (PL-34946)

Issue – The following permissions are defined incorrectly:

- `perm.Holiday.edit`
- `perm.Holiday.delete`
- `perm.Note.view`

Workaround – To any role that contains the `holidaymanage` permission, add the `buswkmanage` permission to that role. To any role that contains the `noteview` permission, add the `noteedit` permission to that role.

Alt+Shift+L to reload PCF files does not work when the server is run from the command line (PLWEB-5819)

Issue – If you run the server from the command line with the command `gwb runServer`, then pressing `Alt+Shift+L` in the application interface does not reload PCF files.

Workaround – Run the server from Guidewire Studio instead.

Server Tools downloads may time out (PLWEB-6184)

Issue – On the **Server Tools** pages, some downloads may take too long and generate a timeout error.

Workaround – Increase the value of the `WebUIAJAXTimeout` configuration parameter.

During rule import, selecting the deployed rule results in incorrect rule version and status (BIZ-912)

Issue – Under certain circumstances, the business rule import process sets the rule version incorrectly. This occurs in the following circumstances:

- The existing and importing rules show as a version conflict.
- The existing or importing rule version is in the deployed state.
- The other rule version has a status other than deployed (draft, staged, approved).
- The deployed version of the rule is selected.

In all of these cases, the rule version is set to Approved with the deployed rule version number. For example, the deployed version is 1 and the version after import is 1+ Approved.

Workaround – Guidewire is aware of this issue.

Guidewire PolicyCenter 9.0.2 Release Notes

Release 9.0.2

Overview

These release notes contain the following topics:

- "Release Information" on page 275
- "Installing This Release" on page 276
- "Support" on page 276
- "Major Issues and Changes" on page 276
- "Improvements and Resolved Issues" on page 277
- "Known Issues and Limitations" on page 280

Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

> **IMPORTANT** If you skipped one or more upgrade releases to PolicyCenter, be sure to read the Release Notes for those releases to learn about changes and fixed issues.

Version Number

This release of Guidewire PolicyCenter is 9.0.2.

- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 8.0.1 or later. The latest maintenance release is preferred.

  If you are implementing multiple settlement currencies, BillingCenter 8.0.2 or later is preferred.

- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 8.0.0 or later. The latest maintenance release is preferred.

- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 8.0.1 or later. The latest maintenance release is preferred.

Installing This Release

Refer to the following:

- The *PolicyCenter Installation Guide* for general installation information
- Prior PolicyCenter release notes for any versions that you have skipped

Upgrade Information

IMPORTANT The procedure for upgrading to PolicyCenter 9 from PolicyCenter 8 or previous releases has changed significantly. In PolicyCenter 9, there are new tools available for performing the upgrade. Do not rely on your past knowledge of the upgrade process to begin the upgrade procedure. Before beginning your upgrade, review the topic "Configuration Upgrade Overview" in the *PolicyCenter Upgrade Guide*.

Exception when upgrading UWIssue objects in sample data from 7.0.x (PC-34046)

When upgrading from PolicyCenter 7.0.x, database upgrade throws an `illegationArgumentException` exception when encountering `UWIssue` objects in the sample data. The exception occurs because the upgrade expects to find typelist values added in PolicyCenter 8.0.x.

To workaround this issue, delete `UWIssue` objects in the sample data from the PolicyCenter 7.0.x database or add the typelist values necessary.

Support

For assistance, visit the Guidewire Community.

Guidewire Customers

- `https://community.guidewire.com`

Guidewire Partners

- `https://partner.guidewire.com`

Major Issues and Changes

This topic contains major issues and changes that can affect your installation.

New and Changed Features

For information on new features and other changes, see the topic "New and Changed in PolicyCenter 9.0.1" in the *PolicyCenter New and Changed Guide*.

Free Text Search is Supported on WebLogic

Free-text search is now supported on the WebLogic application server.

Base PCF File Changes

The following link requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 9.0.1 to 9.0.2

To view a report of the changes to the base PCF files, *click here*.

Base Resource Changes

The following link requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 9.0.1 to 9.0.2

To view a report of the changes to the base resources, *click here*.

Changes in This Release Provided in Upgrade Diff Report

Guidewire provides a report detailing certain differences between the current release and your prior release. This report describes changes in display keys, entities, typelists, and the Gosu API. To obtain your custom Upgrade Diff Report, visit the Guidewire Community.

Improvements and Resolved Issues

This topic describes improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This list is not intended to be comprehensive.

Application improvements and resolved issues

ID	Description
Account	
PC-34675	The `Account#getIssuedPolicies` method now queries directly instead of using `PolicySearchCriteria`.
Administration	
PC-34696	Fixed a previously known issue where policy form patterns were not updated when importing from **Administration→Utilities→Import Data** if there were records needing resolution and you selected **Case-by-case resolution** in **Existing Record Resolution**.
Archiving / Purging	
PC-34008	Fixed an issue where policies with outstanding contingencies were marked for archiving. Added an `OutstandingStatuses` typefilter to `ContingencyStatus.tti`. This filter lists contingency statuses considered outstanding during archiving. With this fix, only policies with no outstanding contingency can become archiving candidates.
Billing / Payments	
PC-34769	In `PolicyPeriodValidation.gs`, removed the unused method `isBasedOnPeriodArchivedInBillingSystem`. If you need a similar method, see `checkThatBasedOnPeriodNotArchivedInBillingSystem` in `JobProcess.gs`.
Data Model / Core Entities	
PC-26435	The `LossHistoryType` typelist is not final anymore.
General	
PC-34335	Fixed navigation back to **Financial Transactions** from **All Transactions**, **Transactions by Job**, and **Transactions by Period**.
PC-34664	The `Activity.RelatedTo` virtual property is now deprecated. This property always returned the empty string.
PC-34687	On the **Account File Document** screen, the **Delete** button inside the **Actions** column now deletes as expected.

ID	Description
PC-34730	Fixed an issue with the way documents were queried from the database. For users with matching producer code, the date range was not taken into account. Now the producer code must match and the document's last modification date must fall with the date range.
Job Infrastructure	
PC-33275	If a note is attached to a contingency, it now appears in the list of notes on the **Policy File→Notes** screen.
PC-34738	Added a clearing cache mechanism in `JobWizard` when resuming work on a `PolicyPeriod` that has changed outside the wizard. This fix addresses an issue that caused PolicyCenter to throw a `ConcurrentDataChangeException`.
Jobs	
PC-34723	With this fix, applying changes for a preempted policy change no longer modifies the effective dates of its policy lines.
LOB - PA	
PC-32284	Fixed a previously known issue where personal auto coverages were not synchronized correctly.
Rating	
PC-34080	Fixed an issue with Rating Management impact testing where quoting a test period failed if no rate book was specified. With this fix, quoting uses the specified generic rate book for impact testing. If not specified, impact testing uses the rate book in the cache.
PC-34671	Fixed an issue which occurred when creating a new copy of a rate book using **Create New Edition** if the rate book contained a rate routine referencing a rate table not included in the rate book. Previously, PolicyCenter threw an exception: **The following rate tables do not exist in this book**. With this fix, PolicyCenter displays a warning when user adds rate routines with missing tables.
Revisioning	
PC-34744	Changed the matching logic of `AbstractEffDatedPropertiesMatcher` to compare the `FixedId` value rather than the memory address when filtering out matches. With this fix, the `mergeDuplicateAdds` method no longer incorrectly deletes objects that appeared in more than one slice.
UI Configuration	
PC-34514	Fixed an issue where the `ViewQuote` step in the `QuickSubmission` wizard was hard-coded for personal auto. The `PAQuickSubmission_QuoteScreen.pcf` has been replaced by a generic quick quote screen, `QuickSubmission_QuoteScreen.pcf`. You can now customize the quick quote wizard for other lines of business.

Platform improvements and resolved issues

The following are the primary improvements and issues corrected in this release:

ID	Description
Archiving	
PL-36139	Microsoft SQL Server 2014 introduced a new optimizer cardinality estimator. The use of this optimizer creates sub-optimal plans during archiving, which can eventually lead to deadlocks on SQL Server. Guidewire disables this optimizer by adding a database hint at the end of the archiving UPDATE statement. For a SQL Server database to use this hint, you must grant the `sysadmin` role to the database schema owner.
Build Infrastructure	
PL-34507	You can now export all WSI web service WSDL and related resources to a single JAR file. The new command `gwb exportWsdl` generates the JAR file to the `modules\configuration\build` directory.
PL-36217	Updated the `gwb zipChangedConfig` build command so that generated files are not included in the changed configuration zip file.

ID	Description
Database Support - General	
PL-35933	Guidewire has removed the unused `numfreqvals` attribute on the `<histogramstatistics>` element in file `database-config.xml`.
Database Upgrade	
PL-36178	During upgrade of an Oracle database, if the option `defer-create-nonessential-indexes="true"` was used with the option `degree-parallel-ddl` set to some value greater than 1, an Oracle exception would occur. This problem has been fixed.
Entities/Metadata	
PL-33429	The method `setFieldValue` is deprecated and reserved for Guidewire internal use only.
Management Plugin	
PL-36018	Fixed an issue that blocked publishing several server MBeans if the JMXManagementPlugin was enabled.
Other	
PLWEB-6203	Fixed an issue where putting the server into maintenance mode would not show a message in the UI stating that the server is undergoing maintenance.
PCF - Layout - List View	
PLWEB-6231	Fixed an issue where the cursor focus on a page would skip cells in row iterator widgets.
PCF - Layout - List View - Iterator	
PLWEB-6189	Fixed an issue where the configuration parameter `ListViewPageSizeDefault` did not have an effect.
Web - Performance	
PL-35768	Fixed an integer overflow error in `BCPReplicate.java`.
Web - PCF Compiler - Studio	
PLWEB-6269	Fixed an issue where PCFs would not reload when there was a compilation error in any Gosu code that was loaded as part of the page.
Web - UI/Runtime	
PLWEB-6146	Fixed an issue where the `SelectOnEnter` property on a card would throw an exception.
PLWEB-6252	Fixed an issue where user interface shortcut keys would not work.
Web Services - WSI (New)	
PL-36114	The XML schema files `wsdl.xsd` and `soap11.xsd` were updated to correctly conform to the WS-I Basic Profile 1.1 specification.
PL-36127	If a web service includes the `@WsiCheckDuplicateExternalTransaction` annotation then its `transaction_id` header will be exposed.
PL-36251	Fixed an issue in WSDL generation for services with multiple `@Throws` annotations for the same exception but different reasons. This issue caused an exception in some cases for WSDL consumers. The WSDL now merges related faults into one fault with all explanations concatenated in one comment. The fix prevents an exception that some WSDL consumers experienced.
Work Queues	
PL-36226	Fixed an issue in which class `BulkInsertWorkitemBase` did not use system time in creating work items.
Workflow	
PL-6352	Resource validation now flags additional workflow definition errors.
XMLElement (and XSD types)	

ID	Description
PL-36071	In Studio, you no longer need to explicitly run **Codegen → Generate Xml Classes**. Studio now includes XML classes when it generates internal code automatically.
PL-36146	An XSD file can define an `<enumeration>` element with a `value` attribute that includes Polish characters.

Known Issues and Limitations

This topic describes known issues with this release of Guidewire PolicyCenter.

PolicyCenter Known Issues

Underwriting rules not managed by the Business Rules User Interface appear to be broken when they are not (PC-33501)

Issue – For an underwriting rule that is not maintained directly through the Business Rules user interface for Underwriting Rules, such as a Gosu-driven rule, the user interface does not show correct information about the rule. It shows a red exclamation icon in the Error column, even if the rule has no errors in it. It indicates that the rule has not been enabled, even when it is. If a user views the rule, a prominent warning is shown with a link that says: "Rule is not enabled. Enable it." Users should not try to correct errors in the rule or enable it in the Underwriting Rules interface.
Workaround – Guidewire is aware of this issue.

Coverage Term Option Value Providers not working in Rating (PC-34695)

Issue – If a value provider of coverage term options is created for a rate table column, it will not return any values. This problem only manifests for coverages and coverage terms that were created in 9.0.0 or later.
Workaround – Do not use a value provider for the associated column and enter values directly into the column.

Provide a way to filter out added and removed subentities in a policy differences tree (PC-7929)

Issue – Need a way to filter out the children of an added or removed entity in a policy differences tree.
Workaround – Guidewire is aware of this issue.

When using parallel sessions, approvals made by underwriter are not updated for second user simultaneously viewing same work order (PC-12322)

Issue – Approvals made by an underwriter are not updated for a second user simultaneously viewing the same work order. This occurs in the following sequence of events:
- An agent processes a submission and issues arise
- The agent calls an underwriter, and remains in the policy
- The underwriter approves the issue or issues
- The agent refreshes the **Risk Analysis** screen by going to a different wizard step and returning to the **Risk Analysis** screen

After this sequence of events:
- The new approvals do not appear on the agent's screen
- However, the agent can bind or issue the policy (based on the new approvals by the underwriter)

Workaround – The agent can leave the policy and return to it. The approvals appear as expected. Alternatively, the agent can see the approvals by clicking the **Details** button when issuing the policy.

Must enter VIN when moving from quick quote multi-version or side-by-side to full application (PC-13099)

Issue – In a quick quote, you do not have to enter the VIN or other fields that are required for a full application submission. When converting a quick quote with multi-version or side-by-side quoting to full application, to resolve

validation issues, you must enter the VIN for each of the versions. In a job with side-by-side quoting, this issue can occur with fields that are not base data.

Workaround – Add code that copies this type of data to the other versions.

PolicyCenter does not send renewal billing instruction to BillingCenter (PC-15082)

Issue – On receiving payment notification from BillingCenter, PolicyCenter tries to bind. If the policy has a renewal transaction in draft status, PolicyCenter does not bind and send the Renewal Billing Instruction to BillingCenter. The policy might be in draft status after an underwriter edited something on the policy. This is an issue with BillingCenter 3.0.1 and later.

Workaround – Guidewire is aware of this issue.

Renewal processing for renewal conversion assumes BasedOn is populated (PC-15851)

Issue – The `startNewRenewal` method in `gw.webservice.pc.pc700.job.PolicyRenwalAPI` assumes the existence of a prior term or a `basedOn` version. Unless customized, this method can cause problems in certain convert-on-renewal scenarios. This method has been deprecated.

Workaround – If you are developing convert-on-renewal flows, use the `gw.webservice.pc.pc700.job.startConversionRenewal` method.

SQL Query performance in free-text search (PC-18794)

Issue – There are open issues related to the performance of SQL query in free-text search. This occurs the first time that you load the PolicyCenter data into the Solr index database.

Workaround – Guidewire is aware of this issue.

Cannot quote a renewal if the expiring policy term has a handled preemption (PC-19416)

Issue – In Commercial Package Policy in the base configuration, if a policy term has a preemption that has been handled, when a renewal job is created, the **Edit Workorder** button is not always available on the renewal. When this occurs, the only options are to bind or not take the renewal.

Workaround – Guidewire is aware of this issue.

Copy-data with an out-of-range date (PC-20279)

Issue – When working with the copy data feature, if you specify a specific date to copy data from and that date is out of range for the policy, PolicyCenter presents a warning that no applicable policy version could be found. However, the date selector still indicates the requested date, not the date that PolicyCenter will use for the copy if you choose to proceed.

Workaround – As appropriate, modify the user interface to reset the date selector or to hide the ability to select data to copy.

Gosu classes not intended for use (PC-23009)

Issue – Some Gosu classes are visible in Studio, but are not intended for use. You can distinguish these Gosu classes because they have no visibility annotations (neither `@Export` nor `@ReadOnly`) and they are not in a `gw` package.

Workaround – Do not use these methods in configuration. The methods are unsupported and may change or be withdrawn without notice.

Unbound renewal policy transactions can provide invalid dates (PC-25288)

Issue – If a policy renewal is in the renewal process and the user goes from the renewal job wizard to the policy file summary, the as-of date default in the **Date** field of the **Actions** panel will not be entirely valid, as it will be the `editEffectiveDate` of an unbound period from an incomplete job.The quote, reinsurance, and audit schedule, if enabled, screens display exceptions because appropriate periods do not exist on the specified dates.

Workaround – Edit `JobWizardInfoBar.pcf` to filter out any `PolicyPeriod` objects that cannot provide the `editEffectiveDate`.

Billing integration error when final audit is waived from Audit job (PC-25500)

Issue – When the user waives a final audit from within a running Audit job, the BillingCenter integration sends an incomplete message to BillingCenter. The message does not include the term number of the policy period. BillingCenter needs the term number to identify the BillingCenter policy period that needs the change.
Workaround – Guidewire is aware of this issue.

Cannot change availability tables columns after data added (PC-26836)

Issue – The Product Model Guide provides instructions on how to add columns to an availability lookup table. However, after loading availability data, you cannot add or remove columns.
Workaround – Guidewire is aware of this issue.

Rate routine editor message needs notification of parameter change (PC-30084)

Issue – The rate routine editor does not display message if rate table definition parameter is modified.

A rate table definition is referenced in a rate routine. You modify a parameter in the rate table definition. When you return to the rate routine, the rate routine editor does not display a message warning that it is unable to reconcile the definition of rate table.

If you select a rate table, the rate routine editor does not warn you that parameters have been modified in the rate table. You are not given the option to accept this change or cancel.
Workaround – Guidewire is aware of this issue.

Payment screen not updated (PC-30594)

Issue – Updating a payment plan in BillingCenter does not update **Premium Summary**, **Payments**, and **Invoicing** fields on the PolicyCenter **Payment** screen. Even making a selection in the payment plan list view does not update these fields.
Workaround – To refresh these fields, navigate away from and return to the **Payment** screen.

PolicyPeriod.removeFromForms method causes DBCC error (PC-30947)

Issue – The `PolicyPeriod.removeFromForms` method causes a DBCC error.
Workaround – Guidewire is aware of this issue. Do not use this method.

List bill with no matching invoice stream for a payment plan generates WSDL fault (PC-31004)

Issue – This issue occurs when a list bill account does not have a compatible invoice stream for one of the payment plans associated with the list bill account. When adding that list bill account to the policy, PolicyCenter throws an exception:

```
WsdlFault: java.lang.IllegalStateException: Cannot lazily create an invoice stream for charge with
owner {policy period id} on list bill account {account number}
```

Workaround – All payment plans must have compatible invoice streams configured in the billing system.

Comparing policy transactions does not work correctly with out-of-sequence transactions (PC-31190)

Issue – When comparing two policy transactions from the Policy Transactions screen, any changes included in the policy revisions being compared that had been made effective on a later date than the edit effective date of the associated transaction (either within that transaction or in a previous transaction) are not displayed in the comparison. Changes made on earlier dates are included.
Workaround – Guidewire is aware of this issue.

Final audit setting changes if billing method changed in BillingCenter (PC-31360)

Issue – On renewal, final audit is changed from **No** to **Determined by business rule** when the billing method and payment plan have been changed in BillingCenter before starting the renewal. This also occurs with rewrite.
Workaround – Manually reset **Final Audit** to **No**.

Cannot complete unowned activities (PC-31610)

Issue – Cannot complete unowned activities even if you have the correct permissions. This issue is in PolicyCenter.
Workaround – Guidewire is aware of this issue.

Number of payment installments incorrect after reinstatement (PC-32171)

Issue – When integrated with BillingCenter, the number of payment installments may be incorrect after a reinstatement policy transaction. This affects general liability policies with final audit set to true after the policy is canceled. If the policy is reinstated, then the number of installments displayed may be incorrect. The installments in the billing system will still be calculated correctly.
Workaround – Guidewire is aware of this issue. The transaction can be completed without side effects.

Message events against policy term archived in BillingCenter generates error (PC-33320)

Issue – If a policy period is archived in BillingCenter but not in PolicyCenter, that policy can be transferred to a new account owner in PolicyCenter. However, this causes an error in BillingCenter. The transfer is shown as successful in PolicyCenter, but the message queue includes this error: "The operation cannot be performed on an archived PolicyPeriod. The PolicyPeriod that is the target of the operation is archived." BillingCenter does not support transfer to a new account on archived policies or policies retrieved from archive.
Workaround – Guidewire is aware of this issue.

Can issue policy change against policy period retrieved from archive in BillingCenter (PC-33339)

Issue – If a policy period has been retrieved from the archive in BillingCenter and the policy period in BillingCenter is still closed, then you should not be allowed to issue a policy change in PolicyCenter. However, PolicyCenter allows you to issue a policy change. BillingCenter rejects the policy change message and sends an error to PolicyCenter. The message remain in an unsent state in the message queue.
Workaround – Reopen the policy period in BillingCenter and resend the message. This workaround requires that BC-15815 be fixed.

Waiving final audit or premium report throws BadIdentifierException (PC-33351)

Issue – If you start a final audit and select **Waive** as the close option, PolicyCenter throws a `BadIdentifierException`. The waive final audit message is not sent to BillingCenter. This also affect premium reports.
Workaround – Waive the final audit from the **Audit Schedule** screen.

Error when deleting a contingency document (PC-34412)

Issue – On a contingency, deleting a document whose contents were replaced using upload gives the error "Object you are trying to update was changed by another user."
Workaround – Guidewire is aware of this issue.

Hidden documents associated with a contingency appear on the Contingency screen (PC-33731)

Issue – In general, hidden documents do not appear in lists unless you make a selection to include them. However, documents associated with a contingency are not filtered and are visible to any user with permission to view the **Contingency** screen.
Workaround – If desired, add filtering of hidden documents to the PCF file for the **Contingency** screen.

Upgrade to PolicyCenter after running ETL Product Model Loader extension pack (DOC-10285)

Issue – Upgrading to PolicyCenter 8.0.3 or later after running ETL Product Model Loader extension pack may require a manual drop of some tables.
In PolicyCenter 8.0.3, the ETL Product Model Loader Extension Pack functionality moved into the core PolicyCenter product. The extension pack created ETL database tables with `pcx_` as the prefix. PolicyCenter 8.0.3 and later create ETL database tables with `pc_` as the prefix. If you ran this extension pack with a prior version of

PolicyCenter, you will have stale tables after upgrading to PolicyCenter 8.0.3 or later. The upgrade does not fail, but the server detects extra tables in the database.

Workaround – To remove these stale tables, manually drop the old `pcx_` tables created by the ETL Product Model Loader extension pack. The ETL Product Model Loader Extension Pack created these tables:

- `pcx_etlclausepattern`
- `pcx_etlcovtermpattern`
- `pcx_etlcovtermoption`
- `pcx_etlcovtermpackage`
- `pcx_etlpackterm`
- `pcx_etlmodifierpattern`
- `pcx_etlratefactorpattern`

You may also need to update your ETL code to reference the new tables.

Product Designer Known Issues

Cannot use Product Designer to edit an extended object with no base object (PC-27763)

Issue – If you attempt to edit a part of the product model that is based on an entity extension file (`.etx`), and that extension has no base entity file (`.eti`), Product Designer displays errors and refuses to display the contents of the file.

Although Guidewire best practices recommend that you never create an entity file extension (`.etx` file) when no base entity (`.eti` file) exists, Guidewire Studio did not enforce this recommendation. However, the current version of Product Designer does not allow the use of an entity extension without a corresponding entity base file. If such an entity extension exists, Product Designer displays a **NullPointerException** when you attempt to access the affected objects. You then can use the Back command in the browser to view other pages, but you cannot configure the affected objects.

Workaround – Locate all `.etx` files that do not have corresponding `.eti` files, and rename them with an `.eti` extension.

PolicyCenter does not start in single currency display mode if the product model is configured to use multiple currencies (PC-27829)

Issue – Product Designer allows users to configure multicurrency features without regard for the setting of the `MultiCurrencyDisplayMode` in the PolicyCenter `config.xml` file. If you configure multiple currencies in a product model object, and the value of the `MultiCurrencyDisplayMode` property is `SINGLE`, PolicyCenter does not start.

For example, if you add clauses that specify multiple coverage currencies and the PolicyCenter `MultiCurrencyDisplayMode` is set to `SINGLE`, the server fails to start and displays messages similar to the following:

```
ERROR ... PolicyLinePattern : ... [field : AvailableCoverageCurrencies] : - Multiple Coverage
Currencies defined for product not allowed in single Currency mode
ERROR ... [ExclusionPattern] : ... [DirectCovTermPattern] : ... [LimitsSet] : - The pattern specifies
limits for different currencies in SingleCurrencyMode
```

Workaround – If you intend to deploy a product model on PolicyCenter servers that are not configured for multiple currency display mode, do not configure multiple currencies in any part of the product model. However, you can test multicurrency products on a staging server configured for multicurrency while the production system is configured for a single currency.

System table cells display public IDs when referencing another system table (PC-28368)

Issue – If the cells in a particular column of a system table are referencing values from another system table, only the public IDs of those values are shown, not the values, when the system table is edited in Product Designer. In addition, users cannot search for values in these system table cells. The referenced system table, however, does not display the public ids so they cannot be viewed from within Product Designer.

Workaround – View the XML directly for the system table being referenced to determine the public IDs associated with the values that need to be added, edited or deleted. You can then use those public IDs directly.

An exception might appear when you run Product Designer (PC-28531)

Issue – An `IncompatibleRemoteServiceException` might appear if you run Product Designer 8.0.3 or later after running a previous version of Product Designer.

Workaround – Refresh the browser cache by selecting `Ctrl+F5`.

Invalid audit schedules when audit schedule properties file is changed while user is editing audit schedule (PC-28146)

Issue – If you edit an audit schedule in Product Designer at the time that the associated audit schedule properties file is updated, you may not see the properties file changes reflected in Product Designer. This could result in an invalid audit schedule (that matches the previous audit schedule properties file).

Workaround – Perform the following steps:

- In Product Designer, revert the changes to affected audit schedules.
- To load the audit schedule properties file changes into Product Designer, restart Product Designer, logout, or change the changelist.
- Reopen the affected audit schedules.

Availability limitations on schedule coverages (PC-34620)

Issue – Setting availability criteria on scheduled item coverages results in a null pointer exception when viewing policy differences.

Workaround – When configuring schedule coverages:

- Availability is supported on the schedule coverage and its coverage terms (for example, Scheduled Personal Property).
- Availability is supported on the scheduled item coverage pattern options.
- Availability is not supported for the scheduled item coverage and coverage terms (for example, Scheduled Personal Property Item). Set these types of clauses to always available. These clauses inherit their availability from the schedule coverage. That is, the scheduled item coverage is available whenever the schedule coverage is available.

Studio/Platform Issues

Chrome browser cannot display product documentation in HTML format (DOC-7251)

Issue – If you use the Google Chrome browser, you can view the HTML Guidewire product documentation only if it is served by an HTTP server using the `http://` protocol. The Chrome browser cannot load HTML product documentation from your local disk by using the `file://` protocol.

Workaround – Use a different browser, such as Microsoft Internet Explorer. If you use the HTML documentation only to access the PDF files, you can open them directly in the `pdf` subdirectory of the `doc` directory.

Alignment Property Does Not Work for Inputs in a DetailView (PL-29429)

Issue – Inputs in a detail view do not align to the right or center. They always align to the left, regardless of the value of the `align` property.

Workaround – If you need to right align a series of items such as a set of monetary values, put them into a list view instead of a detail view. You can then set the alignment as needed.

JMS inbound integration implementations must be OSGi or Java plugins (PL-32054)

Issue – Writing JMS inbound integration handlers requires casting an object to the type `javax.jms.Message`. In Gosu, for some app servers this fails with the message: "Caused by: gw.lang.parser.exceptions.ErrantGosuClassException: GosuClass test.GosuJMSMessageReplyHandler has errors, and cannot be used at runtime."

Workaround – Implement JMS handler code for inbound integration in Java. Guidewire recommends that for inbound integration Java code, implement your code as an OSGi plugin, not a standard (non-OSGi) Java plugin.

Compilation errors after removing a delegate (PL-34619)

Issue – When you remove a delegate, some internal code is not generated properly, resulting in compilation errors.
Workaround – In Guidewire Studio, on the **Codegen** menu, click **Generate Metadata Classes**.

Several permissions are defined incorrectly (PL-34946)

Issue – The following permissions are defined incorrectly:
- `perm.Holiday.edit`
- `perm.Holiday.delete`
- `perm.Note.view`

Workaround – To any role that contains the `holidaymanage` permission, add the `buswkmanage` permission to that role. To any role that contains the `noteview` permission, add the `noteedit` permission to that role.

Alt+Shift+L to reload PCF files does not work when the server is run from the command line (PLWEB-5819)

Issue – If you run the server from the command line with the command `gwb runServer`, then pressing `Alt+Shift+L` in the application interface does not reload PCF files.
Workaround – Run the server from Guidewire Studio instead.

Server Tools downloads may time out (PLWEB-6184)

Issue – On the **Server Tools** pages, some downloads may take too long and generate a timeout error.
Workaround – Increase the value of the `WebUIAJAXTimeout` configuration parameter.

Compiling project from Studio causes rolling upgrade to fail (PL-36030)

Issue – After compiling your project from Studio, rolling upgrade reports a configuration mismatch, and that a full upgrade is required.
Workaround – Edit the file `PolicyCenter/modules/script/gw-build.gradle`. In that file, locate the following code block:

```
webapp {
  deployTemplates = project.rootProject.file('modules/script/deploy-templates')
  warIncludes['modules/configuration/plugins'] = file('plugins')
}
```

Replace the above code block with the following:

```
webapp {
  deployTemplates = project.rootProject.file('modules/script/deploy-templates')
    file('plugins')?.listFiles().each { File outerFile ->
      outerFile?.listFiles().each { File innerFile ->
        if(innerFile.name != 'idea-gclasses') {
          warIncludes['modules/configuration/plugins/' + outerFile.name + '/' + innerFile.name] =
innerFile
        }
      }
    }
}
```

During rule import, selecting the deployed rule results in incorrect rule version and status (BIZ-912)

Issue – Under certain circumstances, the business rule import process sets the rule version incorrectly. This occurs in the following circumstances:

1. The existing and importing rules show as a version conflict.
2. The existing or importing rule version is in the deployed state.
3. The other rule version has a status other than deployed (draft, staged, approved).
4. The deployed version of the rule is selected.

In all of these cases, the rule version is set to Approved with the deployed rule version number. For example, the deployed version is 1 and the version after import is 1+ Approved.

Workaround – Guidewire is aware of this issue.

"Could not index class" errors appear in logs with JBoss 7 (PL-36231)

Issue – When you deploy the application with JBoss 7, the log includes "Could not index class" errors and associated stack traces.

Workaround – You can safely ignore these messages.

Guidewire PolicyCenter 9.0.1 Release Notes

Release 9.0.1

Overview

These release notes contain the following topics:
- "Release Information" on page 289
- "Installing This Release" on page 290
- "Support" on page 290
- "Major Issues and Changes" on page 290
- "Improvements and Resolved Issues" on page 291
- "Known Issues and Limitations" on page 296

Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

> **IMPORTANT** If you skipped one or more upgrade releases to PolicyCenter, be sure to read the Release Notes for those releases to learn about changes and fixed issues.

Version Number

This release of Guidewire PolicyCenter is 9.0.1.
- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 8.0.1 or later. The latest maintenance release is preferred.

  If you are implementing multiple settlement currencies, BillingCenter 8.0.2 or later is preferred.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 8.0.0 or later. The latest maintenance release is preferred.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 8.0.1 or later. The latest maintenance release is preferred.

Installing This Release

Refer to the following:

- The *PolicyCenter Installation Guide* for general installation information
- Prior PolicyCenter release notes for any versions that you have skipped

Upgrade Information

IMPORTANT The procedure for upgrading to PolicyCenter 9 from PolicyCenter 8 or previous releases has changed significantly. In PolicyCenter 9, there are new tools available for performing the upgrade. Do not rely on your past knowledge of the upgrade process to begin the upgrade procedure. Before beginning your upgrade, review the topic "Configuration Upgrade Overview" in the *PolicyCenter Upgrade Guide*.

UWIssueType Version Trigger (PC-33820)

When upgrading from PolicyCenter 7.0, a version trigger upgrades the `ValueFormatter` on existing `UWIssueType` entity instances from `Currency` to `MonetaryAmount`.

In Gosu code, change the code for `UWIssueType` objects to use `MonetaryAmount` instead of `Currency`.

Exception when upgrading UWIssue objects in sample data from 7.0.x (PC-34046)

When upgrading from PolicyCenter 7.0.x, database upgrade throws an `illegationArgumentException` exception when encountering `UWIssue` objects in the sample data. The exception occurs because the upgrade expects to find typelist values added in PolicyCenter 8.0.x.

To workaround this issue, delete `UWIssue` objects in the sample data from the PolicyCenter 7.0.x database or add the typelist values necessary.

Support

For assistance, visit the Guidewire Community.

Guidewire Customers

- `https://community.guidewire.com`

Guidewire Partners

- `https://partner.guidewire.com`

Major Issues and Changes

This topic contains major issues and changes that can affect your installation.

New and Changed Features

For information on new features and other changes, see the topic "New and Changed in PolicyCenter 9.0.1" in the *PolicyCenter New and Changed Guide*.

Oracle Support

```
The Oracle database is now supported. For version information, visit the Guidewire Community
and search for knowledge article 1005, Supported Software Components.
```

Base PCF File Changes

The following link requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 9.0.0 to 9.0.1

To view a report of the changes to the base PCF files, `click here`.

Base Resource Changes

The following link requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 9.0.0 to 9.0.1

To view a report of the changes to the base resources, `click here`.

Changes in This Release Provided in Upgrade Diff Report

```
Guidewire provides a report detailing certain differences between the current release and your
prior release. This report describes changes in display keys, entities, typelists, and the
Gosu API. To obtain your custom Upgrade Diff Report, visit the Guidewire Community.
```

Improvements and Resolved Issues

This topic describes improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This list is not intended to be comprehensive.

Application improvements and resolved issues

ID	Description
Administration	
PC-34011	Fixed a performance issue with Team Screens batch processing.
Billing / Payments	
PC-32698	Added an error message that displays when triggering incompatible BillingCenter plugin functionality.
PC-33371	Fixed a previously known issue which caused an error when revising final audit on a policy period archived in BillingCenter. This issue was fixed in PolicyCenter 9.0.0.
Contacts	
PC-34051	For contact change messages sent to ContactManager, PolicyCenter now populates the `External_UpdateUser` parameter in the ContactAPI `XmlBackedInstance` with username rather than display name.
PC-34094	When creating a new primary named insured in a future dated policy change, the address is now correctly populated on the page.
Data Model / Core Entities	
PC-34014	Fixed an issue where workers' compensation class codes containing letters caused an error during quote.
PC-34047	Overlap checking using `isOverlap` method no longer fails if neighboring intervals occur in arbitrary order in the XML file. This issue prevented PolicyCenter from starting.
Diffs	

ID	Description
PC-33889	Fixed an issue with multi-version policy transactions where comparing branches at their respective slices switched the two versions.
Financials	
PC-33424	Transaction calculator emits $0 offset transactions if their end dates are before or the same as corresponding onset transactions.
General	
PC-34033	Asynchronous document content storage is correctly enabled in this release. In earlier releases, if errors occurred in initial synchronous storage, documents were discarded rather than sent asynchronously. The asynchronous document content store transport (documentStoreTransport.gwp) is now enabled in the Plugins registry. The asynchronous document store message destination (ID 324) is now enabled in the Messaging registry. The configuration now conforms to Sync-first configuration as described in the *Integration Guide* in "Asynchronous Document Content Storage".
Localization	
PC-34024	Fixed unlocalized account type in move policy error message.
PC-34026	Fixed unlocalized date format in **Financial Transactions→Policy Transactions by Job** screen
PC-34096	The **Remove** button on the **New Affinity Group→Products** tab is now localizable.
PC-34097	In **Risk Analysis**, on the **Risk Approval Details** popup, answer values to qualification queries are now localizable.
PC-34483	In the Japanese language pack, localized the EffectiveDate in Type(Date) column in the PolicyFile_Transactions_All screen so it is consistent with other dates under the Japanese Imperial calendar.
Job Infrastructure	
PC-33939	DBCC check for all closed assignments no longer takes into account retired entities.
PC-34353	In a policy change, the operations are now reordered to get the appropriate policy period. Then the dates are adjusted for that period.
Jobs	
PC-33607	The **Audit Details** screen now displays prior final audit payroll for audit revisions.
LOB - CPP	
PC-33999	The **Units-Each** class code basis type no longer prorates the basis. Class code 0804 has a basis type units-each.
Performance Improvement	
PC-33842	The attribute allowInitialValueForUpgrade is now set to true for PreferredCoverageCurrency in the PolicyLine entity.
PC-33891	Improved performance when adding a group within an existing organization with a large number of groups.
PC-34107	Improved performance of product model synchronization by adding an API that enables additional caching during product model synchronization. The following Gosu classes now use the new API: gw.job.JobProcess • gw.validation.PCValidationContext • gw.web.productmodel.ProductModelSyncIssuesHandler If you have modified these classes, merge the changes into your code. If you have implemented your own methods for product model synchronization or validation, consider using the new API in your code, and refer to these updates for examples. The new API, available in Gosu through class gw.api.productmodel.AvailabilityContextScope, must only be used in situations like product model synchronization, where the base policy data accessed by lookuptables.xml cannot change.

ID	Description

	When using this API, always use a Gosu using clause to wrap product model synchronization code. For example, `ProductModelSyncIssuesHandler#syncWithIssues` now sets up a new `AvailabilityContextScope` before calling its addIssues block:
	
	`private static function syncWithIssues(helper : JobWizardHelper, addIssues(issues : ArrayList<ProductModelSyncIssueWrapper>) ) {`
	`var issuesList = new ArrayList<ProductModelSyncIssueWrapper>()`
	`using (new AvailabilityContextScope ()) {`
	`addIssues(issuesList)`
	`}`
	`if(helper != null){`
	`filterAndDisplayIssuesAfterSync( issuesList, helper )`
	`}`
	`}`

PC-34118	Improved performance of policy search by name.
Product Model	
PC-33447	Coverable subtypes can now be used with Coverages defined on their base type.
PC-34534	Starting in PolicyCenter 9.0.0, product model patterns have both a `PublicID` and a `CodeIdentifier`. This change enhances the ETL product model tables such that they contain columns and values for both `PublicID` and `CodeIdentifier`. Downstream systems that need to consume product model information and need a human-readable identifier for product model patterns should use the `CodeIdentifier` column on the ETL tables.
Revisioning	
PC-33718	If several canceled terms overlap in effective time for a single policy, retrieving this policy as of a date within the overlap window now returns the canceled period having the highest `PeriodID`.
Security	
PC-33698	Social security numbers now use the `PrivacyInput` widget.
UI Configuration	
PC-33510	With this fix, the `AccountFile_Billing` screen does not start in edit mode anymore. The input widget with ID `BillingAccounts` has been replaced by a similar `ToolbarRangeInput`, so it can be changed when not in edit mode.
PC-34013	This fix corrects the **Policies Affected** diagram on the **Impact Result** screen.
Underwriting	
PC-33728	PolicyCenter no longer throws an error when deleting an updated contingency document.

Product Designer improvements and resolved issues

ID	Description
Underwriting issue types	
PC-33414	Fixed a previously known issue where the underwriting issue types must be kept in synchronization between Product Designer and PolicyCenter. If the `uw_issue_types.xml` file was not synchronized then it could cause an error to occur when you attempted to link underwriting issues with question sets in Product Designer. This issue was fixed in PolicyCenter 9.0.0.

Platform improvements and resolved issues

The following are the primary improvements and issues corrected in this release:

ID	Description
Business Rules	
BIZ-822	Fixed an issue where editing the display values for the existing typekeys in `Parentheses.tti` and `RuleBooleanOperator.tti` would change the translation of the Rule Condition into Gosu.
Database Instrumentation	
PL-35669	The Server Tools **Database Statistics** download report now contains the full SQL command used to generate the table statistics. The report displays the command in the `DBMS_STATS.GATHER_TABLE_STATS` column for each listed table.
PL-35708	Fixed an issue that reported the elapsed time for consistency checks in the application log as 0.000 seconds.
Database Support - Oracle	
PL-34858	Removed the `<indexstatistics>` element from file `database-config.xml`.
PL-34960	Removed the **Update Statistics Statements for Indexes** table report from the Server Tools **Database Statistics** download report.
PL-34962	Fixed an issue that caused a Null Pointer Error during download of the (Server Tools) Database Catalog Statistics Information report. Attribute `numbuckets` is now a required attribute on the `<histogramstatistics>` element in `database-config.xml`.
PL-35491	The Oracle database is now supported. For version information, visit the Guidewire Community and search for knowledge article 1005, *Supported Software Components*.
Database Upgrade	
PL-34087	Previously, during an Oracle database upgrade, if a table with locked statistics caused upgrade to fail, the error was not found until the end of the database upgrade process. Database upgrade now checks for this problem at the beginning of the upgrade process, allowing a fix to be applied immediately. **Note:** See the *Upgrade Guide* for information on disabling statistics update for tables with locked statistics.
PL-34740	Database upgrade now checks for multiple foreign key constraints on a column when verifying that the physical database and the data model match. Previously, the schema verifier assumed that there was at most one foreign key constraint on a column. Now, if the schema verifier finds multiple foreign key constraints on a column, the unexpected foreign keys are removed and a warning about the mismatch is logged.
Document Management	
PL-34624	Fixed an issue with printing notes that have attached documents.
Entities/Metadata	
PL-35825	Fixed an error that would occur when existing subtypes had their typecode retired.
PL-36060	Fixed an issue where generating the Data Dictionary would fail if there were certain Gosu errors in PCF files.
GX Tools	
PL-35531	Fixed an issue with GX Model serialization of typekey types when the Incremental option is set to true.
Logging	
PLWEB-4353	When an exception occurs while rendering the UI, the error log now records the source location of the exception.
Messaging	

ID	Description
PL-34569	There is now support for optionally distributing payload transformation across many servers in the cluster. The new distributed payload transformation work runs as a process across potentially multiple servers in the cluster, potentially long before the actual send time of the message. You can enable and configure this feature separately for each messaging destination. As part of this new feature, there is a new `Message` property called Bound, which is a `boolean` flag that specifies whether the message request processing is complete. You can optionally set it after message creation if you know that the message does not need processing. After the server performs the before send processing by calling the appropriate `beforeSend` method, the server sets `Message.Bound` to true. To correctly handle message resending, your `beforeSend` method must check `Message.Bound` and behave accordingly because the message may have the Bound property already set to `true`. You must ensure that the `beforeSend` request processing is idempotent, which means that it could be called multiple times and have the same effect. For important related information, see "Distributed Message Payload Transformations Before Sending" in the 9.0.1 section of the *New and Changed Guide*.
PL-35458	Fixed an issue where messaging leases are not released when changing to Maintenance run level.

Metadata Code Generator

PL-35307	Retired typekey values now render in Java and Gosu in strikethrough text.

PCF - Layout - List View

PLWEB-6177	Fixed an issue where clicking a check box in a list view would select more than one row.

PCF - Widget - Choice

PLWEB-6079	Fixed an issue where radio buttons were cut off on the right side in Internet Explorer.

PCF - Widget - Range

PLWEB-6169	Fixed an issue that caused `MultiSelect` range inputs to display incorrectly in certain circumstances.

PCF - Widget - Text

PLWEB-6186	Fixed an issue where a line might wrap in the middle of a word.

Search

PL-32634	Added Solr integration to the **Management Beans** page.
PL-34749	Fixed an issue that could cause an HTTP 400 error when submitting a Solr index request.
PL-35884	Fixed an issue that prevented the server locating the Solr distribution archive when server is launched from the command line and embedded Solr is enabled.

Web - PCF Compiler

PLWEB-6088	Fixed the PCF Format Reference file so that it does not change between releases when there have been no changes to the PCF schema.

Web - Styling

PLWEB-6049	Fixed an issue where the drop-down arrow for a disabled Tab bar menu was not disabled.

Web - UI/Runtime

PLWEB-6165	Fixed an error that would occur when pressing `Enter` inside a combo box.

Web Services - WSI (New)

PL-35045	For consuming WSDL for external web services, the product now additionally supports SOAP 1.2 schema.

XMLElement (and XSD types)

PL-35528	For all XML code generation (XSD, GX model, WSDL), console output is now in the language specified as the default locale.

Known Issues and Limitations

This topic describes known issues with this release of Guidewire PolicyCenter.

PolicyCenter Known Issues

Underwriting rules not managed by the Business Rules User Interface appear to be broken when they are not (PC-33501)

Issue – For an underwriting rule that is not maintained directly through the Business Rules user interface for Underwriting Rules, such as a Gosu-driven rule, the user interface does not show correct information about the rule. It shows a red exclamation icon in the Error column, even if the rule has no errors in it. It indicates that the rule has not been enabled, even when it is. If a user views the rule, a prominent warning is shown with a link that says: "Rule is not enabled. Enable it." Users should not try to correct errors in the rule or enable it in the Underwriting Rules interface.
Workaround – Guidewire is aware of this issue.

Coverage Term Option Value Providers not working in Rating (PC-34695)

Issue – If a value provider of coverage term options is created for a rate table column, it will not return any values. This problem only manifests for coverages and coverage terms that were created in 9.0.0 or later.
Workaround – Do not use a value provider for the associated column and enter values directly into the column.

Provide a way to filter out added and removed subentities in a policy differences tree (PC-7929)

Issue – Need a way to filter out the children of an added or removed entity in a policy differences tree.
Workaround – Guidewire is aware of this issue.

When using parallel sessions, approvals made by underwriter are not updated for second user simultaneously viewing same work order (PC-12322)

Issue – Approvals made by an underwriter are not updated for a second user simultaneously viewing the same work order. This occurs in the following sequence of events:
- An agent processes a submission and issues arise
- The agent calls an underwriter, and remains in the policy
- The underwriter approves the issue or issues
- The agent refreshes the **Risk Analysis** screen by going to a different wizard step and returning to the **Risk Analysis** screen

After this sequence of events:
- The new approvals do not appear on the agent's screen
- However, the agent can bind or issue the policy (based on the new approvals by the underwriter)

Workaround – The agent can leave the policy and return to it. The approvals appear as expected. Alternatively, the agent can see the approvals by clicking the **Details** button when issuing the policy.

Must enter VIN when moving from quick quote multi-version or side-by-side to full application (PC-13099)

Issue – In a quick quote, you do not have to enter the VIN or other fields that are required for a full application submission. When converting a quick quote with multi-version or side-by-side quoting to full application, to resolve validation issues, you must enter the VIN for each of the versions. In a job with side-by-side quoting, this issue can occur with fields that are not base data.
Workaround – Add code that copies this type of data to the other versions.

PolicyCenter does not send renewal billing instruction to BillingCenter (PC-15082)

Issue – On receiving payment notification from BillingCenter, PolicyCenter tries to bind. If the policy has a renewal transaction in draft status, PolicyCenter does not bind and send the Renewal Billing Instruction to BillingCenter. The

policy might be in draft status after an underwriter edited something on the policy. This is an issue with BillingCenter 3.0.1 and later.

Workaround – Guidewire is aware of this issue.

Renewal processing for renewal conversion assumes BasedOn is populated (PC-15851)

Issue – The `startNewRenewal` method in `gw.webservice.pc.pc700.job.PolicyRenwalAPI` assumes the existence of a prior term or a `basedOn` version. Unless customized, this method can cause problems in certain convert-on-renewal scenarios. This method has been deprecated.

Workaround – If you are developing convert-on-renewal flows, use the `gw.webservice.pc.pc700.job.startConversionRenewal` method.

SQL Query performance in free-text search (PC-18794)

Issue – There are open issues related to the performance of SQL query in free-text search. This occurs the first time that you load the PolicyCenter data into the Solr index database.

Workaround – Guidewire is aware of this issue.

Cannot quote a renewal if the expiring policy term has a handled preemption (PC-19416)

Issue – In Commercial Package Policy in the base configuration, if a policy term has a preemption that has been handled, when a renewal job is created, the **Edit Workorder** button is not always available on the renewal. When this occurs, the only options are to bind or not take the renewal.

Workaround – Guidewire is aware of this issue.

Copy-data with an out-of-range date (PC-20279)

Issue – When working with the copy data feature, if you specify a specific date to copy data from and that date is out of range for the policy, PolicyCenter presents a warning that no applicable policy version could be found. However, the date selector still indicates the requested date, not the date that PolicyCenter will use for the copy if you choose to proceed.

Workaround – As appropriate, modify the user interface to reset the date selector or to hide the ability to select data to copy.

Gosu classes not intended for use (PC-23009)

Issue – Some Gosu classes are visible in Studio, but are not intended for use. You can distinguish these Gosu classes because they have no visibility annotations (neither `@Export` nor `@ReadOnly`) and they are not in a `gw` package.

Workaround – Do not use these methods in configuration. The methods are unsupported and may change or be withdrawn without notice.

Unbound renewal policy transactions can provide invalid dates (PC-25288)

Issue – If a policy renewal is in the renewal process and the user goes from the renewal job wizard to the policy file summary, the as-of date default in the **Date** field of the **Actions** panel will not be entirely valid, as it will be the `editEffectiveDate` of an unbound period from an incomplete job.The quote, reinsurance, and audit schedule, if enabled, screens display exceptions because appropriate periods do not exist on the specified dates.

Workaround – Edit `JobWizardInfoBar.pcf` to filter out any `PolicyPeriod` objects that cannot provide the `editEffectiveDate`.

Billing integration error when final audit is waived from Audit job (PC-25500)

Issue – When the user waives a final audit from within a running Audit job, the BillingCenter integration sends an incomplete message to BillingCenter. The message does not include the term number of the policy period. BillingCenter needs the term number to identify the BillingCenter policy period that needs the change.

Workaround – Guidewire is aware of this issue.

Cannot change availability tables columns after data added (PC-26836)

Issue – The Product Model Guide provides instructions on how to add columns to an availability lookup table. However, after loading availability data, you cannot add or remove columns.

Workaround – Guidewire is aware of this issue.

Rate routine editor message needs notification of parameter change (PC-30084)

Issue – The rate routine editor does not display message if rate table definition parameter is modified.

A rate table definition is referenced in a rate routine. You modify a parameter in the rate table definition. When you return to the rate routine, the rate routine editor does not display a message warning that it is unable to reconcile the definition of rate table.

If you select a rate table, the rate routine editor does not warn you that parameters have been modified in the rate table. You are not given the option to accept this change or cancel.

Workaround – Guidewire is aware of this issue.

Payment screen not updated (PC-30594)

Issue – Updating a payment plan in BillingCenter does not update **Premium Summary**, **Payments**, and **Invoicing** fields on the PolicyCenter **Payment** screen. Even making a selection in the payment plan list view does not update these fields.

Workaround – To refresh these fields, navigate away from and return to the **Payment** screen.

PolicyPeriod.removeFromForms method causes DBCC error (PC-30947)

Issue – The `PolicyPeriod.removeFromForms` method causes a DBCC error.

Workaround – Guidewire is aware of this issue. Do not use this method.

List bill with no matching invoice stream for a payment plan generates WSDL fault (PC-31004)

Issue – This issue occurs when a list bill account does not have a compatible invoice stream for one of the payment plans associated with the list bill account. When adding that list bill account to the policy, PolicyCenter throws an exception:

```
WsdlFault: java.lang.IllegalStateException: Cannot lazily create an invoice stream for charge with
owner {policy period id} on list bill account {account number}
```

Workaround – All payment plans must have compatible invoice streams configured in the billing system.

Comparing policy transactions does not work correctly with out-of-sequence transactions (PC-31190)

Issue – When comparing two policy transactions from the Policy Transactions screen, any changes included in the policy revisions being compared that had been made effective on a later date than the edit effective date of the associated transaction (either within that transaction or in a previous transaction) are not displayed in the comparison. Changes made on earlier dates are included.

Workaround – Guidewire is aware of this issue.

Final audit setting changes if billing method changed in BillingCenter (PC-31360)

Issue – On renewal, final audit is changed from **No** to **Determined by business rule** when the billing method and payment plan have been changed in BillingCenter before starting the renewal. This also occurs with rewrite.

Workaround – Manually reset **Final Audit** to **No**.

Cannot complete unowned activities (PC-31610)

Issue – Cannot complete unowned activities even if you have the correct permissions. This issue is in PolicyCenter.

Workaround – Guidewire is aware of this issue.

Number of payment installments incorrect after reinstatement (PC-32171)

Issue – When integrated with BillingCenter, the number of payment installments may be incorrect after a reinstatement policy transaction. This affects general liability policies with final audit set to true after the policy is canceled. If the policy is reinstated, then the number of installments displayed may be incorrect. The installments in the billing system will still be calculated correctly.

Workaround – Guidewire is aware of this issue. The transaction can be completed without side effects.

Personal Auto coverages are not synchronized correctly (PC-32284)

Issue – In the base configuration, personal auto coverages are not synchronized correctly in the Personal Auto policy line. Initial Personal Auto coverages on vehicles are synchronized only when a user goes to the **PA Coverages** screen. At that point, the coverages get added as of the date that the slice selector is on. This means that the coverages could be added on an effective date later than the vehicle was added to the policy. At a later time, the coverages are added again on the effective date of the vehicle. These coverages are propagated forward to other slices, overwriting changes made to coverages or coverage term options when they were initially added.

Workaround – Change the product model synchronization logic for Personal Auto so that the initial vehicle coverages are added at the time the vehicle is added.

Message events against policy term archived in BillingCenter generates error (PC-33320)

Issue – If a policy period is archived in BillingCenter but not in PolicyCenter, that policy can be transferred to a new account owner in PolicyCenter. However, this causes an error in BillingCenter. The transfer is shown as successful in PolicyCenter, but the message queue includes this error: "The operation cannot be performed on an archived PolicyPeriod. The PolicyPeriod that is the target of the operation is archived." BillingCenter does not support transfer to a new account on archived policies or policies retrieved from archive.

Workaround – Guidewire is aware of this issue.

Can issue policy change against policy period retrieved from archive in BillingCenter (PC-33339)

Issue – If a policy period has been retrieved from the archive in BillingCenter and the policy period in BillingCenter is still closed, then you should not be allowed to issue a policy change in PolicyCenter. However, PolicyCenter allows you to issue a policy change. BillingCenter rejects the policy change message and sends an error to PolicyCenter. The message remain in an unsent state in the message queue.

Workaround – Reopen the policy period in BillingCenter and resend the message. This workaround requires that BC-15815 be fixed.

Waiving final audit or premium report throws BadIdentifierException (PC-33351)

Issue – If you start a final audit and select **Waive** as the close option, PolicyCenter throws a `BadIdentifierException`. The waive final audit message is not sent to BillingCenter. This also affect premium reports.

Workaround – Waive the final audit from the **Audit Schedule** screen.

Error when deleting a contingency document (PC-34412)

Issue – On a contingency, deleting a document whose contents were replaced using upload gives the error "Object you are trying to update was changed by another user."

Workaround – Guidewire is aware of this issue.

Hidden documents associated with a contingency appear on the Contingency screen (PC-33731)

Issue – In general, hidden documents do not appear in lists unless you make a selection to include them. However, documents associated with a contingency are not filtered and are visible to any user with permission to view the **Contingency** screen.

Workaround – If desired, add filtering of hidden documents to the PCF file for the **Contingency** screen.

Records not updated on import (PC-34657)

Issue – When using **Administration→Utilities→Import Data**, if there are records needing resolution and you select **Case-by-case resolution** in **Existing Record Resolution**, existing records are not updated.

Workaround – In **Existing Record Resolution**, select **Overwrite all existing records**.

Upgrade to PolicyCenter after running ETL Product Model Loader extension pack (DOC-10285)

Issue – Upgrading to PolicyCenter 8.0.3 or later after running ETL Product Model Loader extension pack may require a manual drop of some tables.

In PolicyCenter 8.0.3, the ETL Product Model Loader Extension Pack functionality moved into the core PolicyCenter product. The extension pack created ETL database tables with `pcx_` as the prefix. PolicyCenter 8.0.3 and later create ETL database tables with `pc_` as the prefix. If you ran this extension pack with a prior version of PolicyCenter, you will have stale tables after upgrading to PolicyCenter 8.0.3 or later. The upgrade does not fail, but the server detects extra tables in the database.

Workaround – To remove these stale tables, manually drop the old `pcx_` tables created by the ETL Product Model Loader extension pack. The ETL Product Model Loader Extension Pack created these tables:

- `pcx_etlclausepattern`
- `pcx_etlcovtermpattern`
- `pcx_etlcovtermoption`
- `pcx_etlcovtermpackage`
- `pcx_etlpackterm`
- `pcx_etlmodifierpattern`
- `pcx_etlratefactorpattern`

You may also need to update your ETL code to reference the new tables.

Product Designer Known Issues

Cannot use Product Designer to edit an extended object with no base object (PC-27763)

Issue – If you attempt to edit a part of the product model that is based on an entity extension file (`.etx`), and that extension has no base entity file (`.eti`), Product Designer displays errors and refuses to display the contents of the file.

Although Guidewire best practices recommend that you never create an entity file extension (`.etx` file) when no base entity (`.eti` file) exists, Guidewire Studio did not enforce this recommendation. However, the current version of Product Designer does not allow the use of an entity extension without a corresponding entity base file. If such an entity extension exists, Product Designer displays a **NullPointerException** when you attempt to access the affected objects. You then can use the Back command in the browser to view other pages, but you cannot configure the affected objects.

Workaround – Locate all `.etx` files that do not have corresponding `.eti` files, and rename them with an `.eti` extension.

PolicyCenter does not start in single currency display mode if the product model is configured to use multiple currencies (PC-27829)

Issue – Product Designer allows users to configure multicurrency features without regard for the setting of the `MultiCurrencyDisplayMode` in the PolicyCenter `config.xml` file. If you configure multiple currencies in a product model object, and the value of the `MultiCurrencyDisplayMode` property is `SINGLE`, PolicyCenter does not start.

For example, if you add clauses that specify multiple coverage currencies and the PolicyCenter `MultiCurrencyDisplayMode` is set to `SINGLE`, the server fails to start and displays messages similar to the following:

```
ERROR ... PolicyLinePattern : ... [field : AvailableCoverageCurrencies] : - Multiple Coverage
Currencies defined for product not allowed in single Currency mode
```

```
ERROR ... [ExclusionPattern] : ... [DirectCovTermPattern] : ... [LimitsSet] : - The pattern specifies
limits for different currencies in SingleCurrencyMode
```

Workaround – If you intend to deploy a product model on PolicyCenter servers that are not configured for multiple currency display mode, do not configure multiple currencies in any part of the product model. However, you can test multicurrency products on a staging server configured for multicurrency while the production system is configured for a single currency.

System table cells display public IDs when referencing another system table (PC-28368)

Issue – If the cells in a particular column of a system table are referencing values from another system table, only the public IDs of those values are shown, not the values, when the system table is edited in Product Designer. In addition, users cannot search for values in these system table cells. The referenced system table, however, does not display the public ids so they cannot be viewed from within Product Designer.

Workaround – View the XML directly for the system table being referenced to determine the public IDs associated with the values that need to be added, edited or deleted. You can then use those public IDs directly.

An exception might appear when you run Product Designer (PC-28531)

Issue – An `IncompatibleRemoteServiceException` might appear if you run Product Designer 8.0.3 or later after running a previous version of Product Designer.

Workaround – Refresh the browser cache by selecting `Ctrl+F5`.

Invalid audit schedules when audit schedule properties file is changed while user is editing audit schedule (PC-28146)

Issue – If you edit an audit schedule in Product Designer at the time that the associated audit schedule properties file is updated, you may not see the properties file changes reflected in Product Designer. This could result in an invalid audit schedule (that matches the previous audit schedule properties file).

Workaround – Perform the following steps:
- In Product Designer, revert the changes to affected audit schedules.
- To load the audit schedule properties file changes into Product Designer, restart Product Designer, logout, or change the changelist.
- Reopen the affected audit schedules.

Availability limitations on schedule coverages (PC-34620)

Issue – Setting availability criteria on scheduled item coverages results in a null pointer exception when viewing policy differences.

Workaround – When configuring schedule coverages:
- Availability is supported on the schedule coverage and its coverage terms (for example, Scheduled Personal Property).
- Availability is supported on the scheduled item coverage pattern options.
- Availability is not supported for the scheduled item coverage and coverage terms (for example, Scheduled Personal Property Item). Set these types of clauses to always available. These clauses inherit their availability from the schedule coverage. That is, the scheduled item coverage is available whenever the schedule coverage is available.

Studio/Platform Issues

Chrome browser cannot display product documentation in HTML format (DOC-7251)

Issue – If you use the Google Chrome browser, you can view the HTML Guidewire product documentation only if it is served by an HTTP server using the `http://` protocol. The Chrome browser cannot load HTML product documentation from your local disk by using the `file://` protocol.

Workaround – Use a different browser, such as Microsoft Internet Explorer. If you use the HTML documentation only to access the PDF files, you can open them directly in the `pdf` subdirectory of the `doc` directory.

Alignment Property Does Not Work for Inputs in a DetailView (PL-29429)

Issue – Inputs in a detail view do not align to the right or center. They always align to the left, regardless of the value of the `align` property.

Workaround – If you need to right align a series of items such as a set of monetary values, put them into a list view instead of a detail view. You can then set the alignment as needed.

JMS inbound integration implementations must be OSGi or Java plugins (PL-32054)

Issue – Writing JMS inbound integration handlers requires casting an object to the type `javax.jms.Message`. In Gosu, for some app servers this fails with the message: "Caused by: gw.lang.parser.exceptions.ErrantGosuClassException: GosuClass test.GosuJMSMessageReplyHandler has errors, and cannot be used at runtime."

Workaround – Implement JMS handler code for inbound integration in Java. Guidewire recommends that for inbound integration Java code, implement your code as an OSGi plugin, not a standard (non-OSGi) Java plugin.

Compilation errors after removing a delegate (PL-34619)

Issue – When you remove a delegate, some internal code is not generated properly, resulting in compilation errors.

Workaround – In Guidewire Studio, on the **Codegen** menu, click **Generate Metadata Classes**.

Free-text search is not supported on WebLogic (PL-34871)

Issue – Free-text search is not supported on a WebLogic application server.

Workaround – Guidewire is aware of this issue.

You may also need to update your ETL code to reference the new tables.

Several permissions are defined incorrectly (PL-34946)

Issue – The following permissions are defined incorrectly:
- `perm.Holiday.edit`
- `perm.Holiday.delete`
- `perm.Note.view`

Workaround – To any role that contains the `holidaymanage` permission, add the `buswkmanage` permission to that role. To any role that contains the `noteview` permission, add the `noteedit` permission to that role.

Alt+Shift+L to reload PCF files does not work when the server is run from the command line (PLWEB-5819)

Issue – If you run the server from the command line with the command `gwb runServer`, then pressing `Alt+Shift+L` in the application interface does not reload PCF files.

Workaround – Run the server from Guidewire Studio instead.

Server Tools downloads may time out (PLWEB-6184)

Issue – On the **Server Tools** pages, some downloads may take too long and generate a timeout error.

Workaround – Increase the value of the `WebUIAJAXTimeout` configuration parameter.

Import fails if source and target business rules data models do not match (BIZ-862)

Issue – During a business rules export and import operation from one system to another (for example, from a test system to a production system), the import fails if the business rules data model does not match between the two systems.

Workaround – Ensure that the business rules data model of the source and target systems is the same before exporting any business rule. If necessary:
1. Modify the business rules data model of one system so that it matches the business rules data model of the other system.

2. Regenerate the business rules export file.

3. Repeat the business rules import operation.

Compiling project from Studio causes rolling upgrade to fail (PL-36030)

Issue – After compiling your project from Studio, rolling upgrade reports a configuration mismatch, and that a full upgrade is required.

Workaround – Edit the file `PolicyCenter/modules/script/gw-build.gradle`. In that file, locate the following code block:

```
webapp {
  deployTemplates = project.rootProject.file('modules/script/deploy-templates')
  warIncludes['modules/configuration/plugins'] = file('plugins')
}
```

Replace the above code block with the following:

```
webapp {
  deployTemplates = project.rootProject.file('modules/script/deploy-templates')
    file('plugins')?.listFiles().each { File outerFile ->
      outerFile?.listFiles().each { File innerFile ->
        if(innerFile.name != 'idea-gclasses') {
          warIncludes['modules/configuration/plugins/' + outerFile.name + '/' + innerFile.name] =
innerFile
      }
    }
  }
}
```

During rule import, selecting the deployed rule results in incorrect rule version and status (BIZ-912)

Issue – Under certain circumstances, the business rule import process sets the rule version incorrectly. This occurs in the following circumstances:

1. The existing and importing rules show as a version conflict.

2. The existing or importing rule version is in the deployed state.

3. The other rule version has a status other than deployed (draft, staged, approved).

4. The deployed version of the rule is selected.

In all of these cases, the rule version is set to Approved with the deployed rule version number. For example, the deployed version is 1 and the version after import is 1+ Approved.

Workaround – Guidewire is aware of this issue.

Guidewire PolicyCenter 9.0.0 Release Notes

Release 9.0.0

Overview

These release notes contain the following topics:

- "Release Information" on page 305
- "Installing This Release" on page 306
- "Support" on page 306
- "Major Issues and Changes" on page 307
- "Improvements and Resolved Issues" on page 307
- "Known Issues and Limitations" on page 309

Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

IMPORTANT If you skipped one or more upgrade releases to PolicyCenter, be sure to read the Release Notes for those releases to learn about changes and fixed issues.

Version Number

This release of Guidewire PolicyCenter is 9.0.0.

- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 8.0.1 or later. The latest maintenance release is preferred.

  If you are implementing multiple settlement currencies, BillingCenter 8.0.2 or later is preferred.

- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 8.0.0 or later. The latest maintenance release is preferred.

- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 8.0.1 or later. The latest maintenance release is preferred.

Installing This Release

Refer to the following:

- The *PolicyCenter Installation Guide* for general installation information
- Prior PolicyCenter release notes for any versions that you have skipped

Supported Java Versions

The following are the supported Java versions for this release:

- Java JDK 1.8.0_92
- DCEVM version Java 8 update 92, build 1 (optional)

Version support for future releases will be published on the Guidewire Community.

Install the Latest Guidewire Studio Update

You can now download and apply updates to Guidewire Studio without needing to upgrade the entire PolicyCenter application. Studio automatically checks for updates and alerts you when they are available. As part of your PolicyCenter installation or upgrade, you should check for the latest update to Studio.

1. Install Guidewire Studio following the standard instructions in the *PolicyCenter Installation Guide*.
2. Run Guidewire Studio.
3. In the **Help** menu, click **Check for Studio Update**.
4. If an update is available, download and apply it.

Upgrade Information

IMPORTANT The procedure for upgrading to PolicyCenter 9 from PolicyCenter 8 or previous releases has changed significantly. In PolicyCenter 9, there are new tools available for performing the upgrade. Do not rely on your past knowledge of the upgrade process to begin the upgrade procedure. Before beginning your upgrade, review the topic "Configuration Upgrade Overview" in the *PolicyCenter Upgrade Guide*.

UWIssueType Version Trigger (PC-33820)

When upgrading from PolicyCenter 7.0, a version trigger upgrades the `ValueFormatter` on existing `UWIssueType` entity instances from `Currency` to `MonetaryAmount`.

In Gosu code, change the code for `UWIssueType` objects to use `MonetaryAmount` instead of `Currency`.

Exception when upgrading UWIssue objects in sample data from 7.0.x (PC-34046)

When upgrading from PolicyCenter 7.0.x, database upgrade throws an `illegationArgumentException` exception when encountering `UWIssue` objects in the sample data. The exception occurs because the upgrade expects to find typelist values added in PolicyCenter 8.0.x.

To workaround this issue, delete `UWIssue` objects in the sample data from the PolicyCenter 7.0.x database or add the typelist values necessary.

Support

For assistance, visit the Guidewire Community.

Guidewire Customers

- `https://community.guidewire.com`

- https://partner.guidewire.com

Major Issues and Changes

This topic contains major issues and changes that can affect your installation.

New and Changed Features

For information on new features and other changes, see the topic "New and Changed in PolicyCenter 9.0.0" in the *PolicyCenter New and Changed Guide*.

Oracle Support Planned

Due to the timing of Oracle releases, this release of Guidewire InsuranceSuite 9 does not support the Oracle database. Guidewire plans to add support for Oracle, and is continuing to evaluate and test Oracle releases. For further updates about Oracle support, please contact your Guidewire account manager.

> **Note:** In anticipation of future Oracle support, the product documentation describes the usage of Oracle with PolicyCenter. However, until Oracle is supported, this content is for informational purposes only.

Base PCF File Changes

The following link requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 8.0.5 to 9.0.0

To view a report of the changes to the base PCF files, `click here`.

Base Resource Changes

The following link requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 8.0.5 to 9.0.0

To view a report of the changes to the base resources, `click here`.

Changes in This Release Provided in Upgrade Diff Report

```
Guidewire provides a report detailing certain differences between the current release and your
prior release. This report describes changes in display keys, entities, typelists, and the
Gosu API. To obtain your custom Upgrade Diff Report, visit the Guidewire Community.
```

Improvements and Resolved Issues

This topic describes improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This list is not intended to be comprehensive.

Application improvements and resolved issues

ID	Description

Activities

| PC-32597 | The link to view or edit an existing activity is on the subject field. If no subject was specified on an activity, you could not view or edit the activity. The subject field for the create/edit activity is now mandatory. |

Administration

| PC-32624 | On the `MessageControlForDestinationScreen`, `MessageControlSearchItem` is now visible so that you can search for messages by account. |

Billing integration

PC-15309	Assume an account has an existing invoice stream and that this account is an Alt Billing Account for new policies (on new accounts). Fixed a previously known issue where a new invoice stream was not created when the new policy's new invoice stream periodicity matched the invoice stream on the Alt Billing Account.
PC-23211	Fixed a previously known issue which caused a billing system integration startup error when creating carrier organization.
PC-23570	Fixed a previously known issue where PolicyCenter allowed owner and overriding-payer accounts to be the same. Added validation to ensure that the Alt Billing Account is not the same as the policy owner.
PC-33371	Fixed a previously known issue which caused an error when revising final audit on a policy period archived in BillingCenter.

Contacts

| PC-32614 | Fixed an issue which caused the message: "There are multiple contacts with the same abuid". This also resulted in a `MultipleMatchesException` when ContactManager tried to update the contact. |

Jobs

PC-30635	With this fix, side-by-side base data copy does not perform base data copy if the selected period is withdrawn. Any changes made in the withdrawn period will not be copied over to other side-by-side periods.
PC-26292	Fixed an issue which caused an illegal argument exception in base data copy in side-by-side quoting.
PC-33164	For each step in the submission wizard, the `OnEnter` method is now executed only once.
PC-32872	Previously rescinded policies can now be cancelled and rescinded again.

Line of business - general liability

| PC-25559 | In the Commercial Auto, General Liability, and Commercial Property lines, users sometimes received an error message after clicking Quote on the Locations screen indicating that there are no locations even when there were. This error could occur in other lines of business and on other list views, depending on how they were configured. |

Line of business - personal auto

| PC-32363 | Fixed a previously known issue with the `NumDaysInCoverageRatedTerm` getter in personal auto. |

Rating

PC-30367	Fixed a previously known issue which cause import rate table from spreadsheet to fail.
PC-32080	Fixed a previously known issue where merged rate books lost references to rate tables. Tables in merged rate books now reference the rows if the source rate book has active status, and own the rows for all other tables.
PC-32602	A rate routine can now reference rate tables which are accessed by cascaded lookup, that is, the rate routine and the rate table data can be in different rate books. During export, the rate table definition for any referenced table is included even if the table data is owned by a different rate book.

Reinsurance

| PC-21720 | A previously known issue where reinsurance warnings were not refreshed even after saving a draft is not reproducible. Previously, PolicyCenter displayed validation errors and warnings both at the top and in the worksheet at the bottom of the screen depending on the action that caused validation to run (typically **Save**, or leaving the screen). Depending on the user flow, one or both of these areas displayed stale messages. |

Search

PC-32366	**Search Accounts** now works with Canadian postal code filter.
PC-30879	Fixed SOLR consistency check errors that were appearing for policies in which the company name changed while in draft status.
PC-32254	Fixed an issue in `PCSolrMessageTransport` which prevented index messages from being retried if the connection between PolicyCenter and Solr was interrupted.
PC-32414	The **Retry** button on `MessageSearch.pcf` now works as expected.
Underwriting	
PC-32551	The **Valid until** column on the **Risk Approval Details** history screen now includes the **Invalid From** type description if no expiration date is set. If you set the expiration date, then PolicyCenter will display it.
PC-32754	Fixed the `ReferenceValueCurrency` column on `UWApprovalLV.pcf` so that the header and cells have the same visibility. This keeps the header and cells in the same alignment.
User interface	
PC-29777	Keyboard shortcuts are now correctly listed on the help page.

Product Designer improvements and resolved issues

ID	Description
System tables	
PC-32776	Previously Product Designer could not handle system table entities that contained `java.lang.Long` data types, or data-types backed by `java.lang.Long`. You can now use the Product Designer system table editor to manage system tables backed by a long.
PC-33414	Fixed a previously known issue where the underwriting issue types must be kept in synchronization between Product Designer and PolicyCenter. If the `uw_issue_types.xml` file was not synchronized then it could cause an error to occur when you attempted to link underwriting issues with question sets in Product Designer.

Known Issues and Limitations

This topic describes known issues with this release of Guidewire PolicyCenter.

PolicyCenter Known Issues

Provide a way to filter out added and removed subentities in a policy differences tree (PC-7929)

Issue – Need a way to filter out the children of an added or removed entity in a policy differences tree.

Workaround – Guidewire is aware of this issue.

When using parallel sessions, approvals made by underwriter are not updated for second user simultaneously viewing same work order (PC-12322)

Issue – Approvals made by an underwriter are not updated for a second user simultaneously viewing the same work order. This occurs in the following sequence of events:

- An agent processes a submission and issues arise
- The agent calls an underwriter, and remains in the policy
- The underwriter approves the issue or issues
- The agent refreshes the **Risk Analysis** screen by going to a different wizard step and returning to the **Risk Analysis** screen

After this sequence of events:
- The new approvals do not appear on the agent's screen
- However, the agent can bind or issue the policy (based on the new approvals by the underwriter)

Workaround – The agent can leave the policy and return to it. The approvals appear as expected. Alternatively, the agent can see the approvals by clicking the **Details** button when issuing the policy.

Must enter VIN when moving from quick quote multi-version or side-by-side to full application (PC-13099)

Issue – In a quick quote, you do not have to enter the VIN or other fields that are required for a full application submission. When converting a quick quote with multi-version or side-by-side quoting to full application, to resolve validation issues, you must enter the VIN for each of the versions. In a job with side-by-side quoting, this issue can occur with fields that are not base data.

Workaround – Add code that copies this type of data to the other versions.

PolicyCenter does not send renewal billing instruction to BillingCenter (PC-15082)

Issue – On receiving payment notification from BillingCenter, PolicyCenter tries to bind. If the policy has a renewal transaction in draft status, PolicyCenter does not bind and send the Renewal Billing Instruction to BillingCenter. The policy might be in draft status after an underwriter edited something on the policy. This is an issue with BillingCenter 3.0.1 and later.

Workaround – Guidewire is aware of this issue.

Renewal processing for renewal conversion assumes BasedOn is populated (PC-15851)

Issue – The `startNewRenewal` method in `gw.webservice.pc.pc700.job.PolicyRenwalAPI` assumes the existence of a prior term or a `basedOn` version. Unless customized, this method can cause problems in certain convert-on-renewal scenarios. This method has been deprecated.

Workaround – If you are developing convert-on-renewal flows, use the `gw.webservice.pc.pc700.job.startConversionRenewal` method.

SQL Query performance in free-text search (PC-18794)

Issue – There are open issues related to the performance of SQL query in free-text search. This occurs the first time that you load the PolicyCenter data into the Solr index database.

Workaround – Guidewire is aware of this issue.

Cannot quote a renewal if the expiring policy term has a handled preemption (PC-19416)

Issue – In Commercial Package Policy in the base configuration, if a policy term has a preemption that has been handled, when a renewal job is created, the **Edit Workorder** button is not always available on the renewal. When this occurs, the only options are to bind or not take the renewal.

Workaround – Guidewire is aware of this issue.

Copy-data with an out-of-range date (PC-20279)

Issue – When working with the copy data feature, if you specify a specific date to copy data from and that date is out of range for the policy, PolicyCenter presents a warning that no applicable policy version could be found. However, the date selector still indicates the requested date, not the date that PolicyCenter will use for the copy if you choose to proceed.

Workaround – As appropriate, modify the user interface to reset the date selector or to hide the ability to select data to copy.

Gosu classes not intended for use (PC-23009)

Issue – Some Gosu classes are visible in Studio, but are not intended for use. You can distinguish these Gosu classes because they have no visibility annotations (neither `@Export` nor `@ReadOnly`) and they are not in a `gw` package.

Workaround – Do not use these methods in configuration. The methods are unsupported and may change or be withdrawn without notice.

Unbound renewal policy transactions can provide invalid dates (PC-25288)

Issue – If a policy renewal is in the renewal process and the user goes from the renewal job wizard to the policy file summary, the as-of date default in the **Date** field of the **Actions** panel will not be entirely valid, as it will be the `editEffectiveDate` of an unbound period from an incomplete job.The quote, reinsurance, and audit schedule, if enabled, screens display exceptions because appropriate periods do not exist on the specified dates.

Workaround – Edit `JobWizardInfoBar.pcf` to filter out any `PolicyPeriod` objects that cannot provide the `editEffectiveDate`.

Billing integration error when final audit is waived from Audit job (PC-25500)

Issue – When the user waives a final audit from within a running Audit job, the BillingCenter integration sends an incomplete message to BillingCenter. The message does not include the term number of the policy period. BillingCenter needs the term number to identify the BillingCenter policy period that needs the change.

Workaround – Guidewire is aware of this issue.

Cannot change availability tables columns after data added (PC-26836)

Issue – The Product Model Guide provides instructions on how to add columns to an availability lookup table. However, after loading availability data, you cannot add or remove columns.

Workaround – Guidewire is aware of this issue.

Rate routine editor message needs notification of parameter change (PC-30084)

Issue – The rate routine editor does not display message if rate table definition parameter is modified.

A rate table definition is referenced in a rate routine. You modify a parameter in the rate table definition. When you return to the rate routine, the rate routine editor does not display a message warning that it is unable to reconcile the definition of rate table.

If you select a rate table, the rate routine editor does not warn you that parameters have been modified in the rate table. You are not given the option to accept this change or cancel.

Workaround – Guidewire is aware of this issue.

Payment screen not updated (PC-30594)

Issue – Updating a payment plan in BillingCenter does not update **Premium Summary**, **Payments**, and **Invoicing** fields on the PolicyCenter **Payment** screen. Even making a selection in the payment plan list view does not update these fields.

Workaround – To refresh these fields, navigate away from and return to the **Payment** screen.

PolicyPeriod.removeFromForms method causes DBCC error (PC-30947)

Issue – The `PolicyPeriod.removeFromForms` method causes a DBCC error.

Workaround – Guidewire is aware of this issue. Do not use this method.

List bill with no matching invoice stream for a payment plan generates WSDL fault (PC-31004)

Issue – This issue occurs when a list bill account does not have a compatible invoice stream for one of the payment plans associated with the list bill account. When adding that list bill account to the policy, PolicyCenter throws an exception:

```
WsdlFault: java.lang.IllegalStateException: Cannot lazily create an invoice stream for charge with
owner {policy period id} on list bill account {account number}
```

Workaround – All payment plans must have compatible invoice streams configured in the billing system.

Comparing policy transactions does not work correctly with out-of-sequence transactions (PC-31190)

Issue – When comparing two policy transactions from the Policy Transactions screen, any changes included in the policy revisions being compared that had been made effective on a later date than the edit effective date of the associated transaction (either within that transaction or in a previous transaction) are not displayed in the comparison. Changes made on earlier dates are included.

Workaround – Guidewire is aware of this issue.

Final audit setting changes if billing method changed in BillingCenter (PC-31360)

Issue – On renewal, final audit is changed from **No** to **Determined by business rule** when the billing method and payment plan have been changed in BillingCenter before starting the renewal. This also occurs with rewrite.

Workaround – Manually reset **Final Audit** to **No**.

Cannot complete unowned activities (PC-31610)

Issue – Cannot complete unowned activities even if you have the correct permissions. This issue is in PolicyCenter.

Workaround – Guidewire is aware of this issue.

Number of payment installments incorrect after reinstatement (PC-32171)

Issue – When integrated with BillingCenter, the number of payment installments may be incorrect after a reinstatement policy transaction. This affects general liability policies with final audit set to true after the policy is canceled. If the policy is reinstated, then the number of installments displayed may be incorrect. The installments in the billing system will still be calculated correctly.

Workaround – Guidewire is aware of this issue. The transaction can be completed without side effects.

Personal Auto coverages are not synchronized correctly (PC-32284)

Issue – In the base configuration, personal auto coverages are not synchronized correctly in the Personal Auto policy line. Initial Personal Auto coverages on vehicles are synchronized only when a user goes to the **PA Coverages** screen. At that point, the coverages get added as of the date that the slice selector is on. This means that the coverages could be added on an effective date later than the vehicle was added to the policy. At a later time, the coverages are added again on the effective date of the vehicle. These coverages are propagated forward to other slices, overwriting changes made to coverages or coverage term options when they were initially added.

Workaround – Change the product model synchronization logic for Personal Auto so that the initial vehicle coverages are added at the time the vehicle is added.

Message events against policy term archived in BillingCenter generates error (PC-33320)

Issue – If a policy period is archived in BillingCenter but not in PolicyCenter, that policy can be transferred to a new account owner in PolicyCenter. However, this causes an error in BillingCenter. The transfer is shown as successful in PolicyCenter, but the message queue includes this error: "The operation cannot be performed on an archived PolicyPeriod. The PolicyPeriod that is the target of the operation is archived." BillingCenter does not support transfer to a new account on archived policies or policies retrieved from archive.

Workaround – Guidewire is aware of this issue.

Can issue policy change against policy period retrieved from archive in BillingCenter (PC-33339)

Issue – If a policy period has been retrieved from the archive in BillingCenter and the policy period in BillingCenter is still closed, then you should not be allowed to issue a policy change in PolicyCenter. However, PolicyCenter allows you to issue a policy change. BillingCenter rejects the policy change message and sends an error to PolicyCenter. The message remain in an unsent state in the message queue.

Workaround – Reopen the policy period in BillingCenter and resend the message. This workaround requires that BC-15815 be fixed.

Waiving final audit or premium report throws BadIdentifierException (PC-33351)

Issue – If you start a final audit and select **Waive** as the close option, PolicyCenter throws a `BadIdentifierException`. The waive final audit message is not sent to BillingCenter. This also affect premium reports.

Workaround – Waive the final audit from the **Audit Schedule** screen.

Error when deleting a contingency document (PC-33728)

Issue – On a contingency, deleting a document whose contents were replaced using upload gives the error "Object you are trying to change has been updated by other user."

Workaround – Guidewire is aware of this issue.

Hidden documents associated with a contingency appear on the Contingency screen (PC-33731)

Issue – In general, hidden documents do not appear in lists unless you make a selection to include them. However, documents associated with a contingency are not filtered and are visible to any user with permission to view the **Contingency** screen.

Workaround – If desired, add filtering of hidden documents to the PCF file for the **Contingency** screen.

Upgrade to PolicyCenter after running ETL Product Model Loader extension pack (DOC-10285)

Issue – Upgrading to PolicyCenter 8.0.3 or later after running ETL Product Model Loader extension pack may require a manual drop of some tables.

In PolicyCenter 8.0.3, the ETL Product Model Loader Extension Pack functionality moved into the core PolicyCenter product. The extension pack created ETL database tables with `pcx_` as the prefix. PolicyCenter 8.0.3 and later create ETL database tables with `pc_` as the prefix. If you ran this extension pack with a prior version of PolicyCenter, you will have stale tables after upgrading to PolicyCenter 8.0.3 or later. The upgrade does not fail, but the server detects extra tables in the database.

Workaround – To remove these stale tables, manually drop the old `pcx_` tables created by the ETL Product Model Loader extension pack. The ETL Product Model Loader Extension Pack created these tables:

- `pcx_etlclausepattern`
- `pcx_etlcovtermpattern`
- `pcx_etlcovtermoption`
- `pcx_etlcovtermpackage`
- `pcx_etlpackterm`
- `pcx_etlmodifierpattern`
- `pcx_etlratefactorpattern`

You may also need to update your ETL code to reference the new tables.

Product Designer Known Issues

Cannot use Product Designer to edit an extended object with no base object (PC-27763)

Issue – If you attempt to edit a part of the product model that is based on an entity extension file (`.etx`), and that extension has no base entity file (`.eti`), Product Designer displays errors and refuses to display the contents of the file.

Although Guidewire best practices recommend that you never create an entity file extension (`.etx` file) when no base entity (`.eti` file) exists, Guidewire Studio did not enforce this recommendation. However, the current version of Product Designer does not allow the use of an entity extension without a corresponding entity base file. If such an entity extension exists, Product Designer displays a **NullPointerException** when you attempt to access the affected objects. You then can use the Back command in the browser to view other pages, but you cannot configure the affected objects.

Workaround – Locate all `.etx` files that do not have corresponding `.eti` files, and rename them with an `.eti` extension.

PolicyCenter does not start in single currency display mode if the product model is configured to use multiple currencies (PC-27829)

Issue – Product Designer allows users to configure multicurrency features without regard for the setting of the `MultiCurrencyDisplayMode` in the PolicyCenter `config.xml` file. If you configure multiple currencies in a product model object, and the value of the `MultiCurrencyDisplayMode` property is `SINGLE`, PolicyCenter does not start.

For example, if you add clauses that specify multiple coverage currencies and the PolicyCenter `MultiCurrencyDisplayMode` is set to `SINGLE`, the server fails to start and displays messages similar to the following:

```
ERROR ... PolicyLinePattern : ... [field : AvailableCoverageCurrencies] : - Multiple Coverage
Currencies defined for product not allowed in single Currency mode
ERROR ... [ExclusionPattern] : ... [DirectCovTermPattern] : ... [LimitsSet] : - The pattern specifies
limits for different currencies in SingleCurrencyMode
```

Workaround – If you intend to deploy a product model on PolicyCenter servers that are not configured for multiple currency display mode, do not configure multiple currencies in any part of the product model. However, you can test multicurrency products on a staging server configured for multicurrency while the production system is configured for a single currency.

System table cells display public IDs when referencing another system table (PC-28368)

Issue – If the cells in a particular column of a system table are referencing values from another system table, only the public IDs of those values are shown, not the values, when the system table is edited in Product Designer. In addition, users cannot search for values in these system table cells. The referenced system table, however, does not display the public ids so they cannot be viewed from within Product Designer.

Workaround – View the XML directly for the system table being referenced to determine the public IDs associated with the values that need to be added, edited or deleted. You can then use those public IDs directly.

An exception might appear when you run Product Designer (PC-28531)

Issue – An `IncompatibleRemoteServiceException` might appear if you run Product Designer 8.0.3 or later after running a previous version of Product Designer.

Workaround – Refresh the browser cache by selecting `Ctrl+F5`.

Invalid audit schedules when audit schedule properties file is changed while user is editing audit schedule (PC-28146)

Issue – If you edit an audit schedule in Product Designer at the time that the associated audit schedule properties file is updated, you may not see the properties file changes reflected in Product Designer. This could result in an invalid audit schedule (that matches the previous audit schedule properties file).

Workaround – Perform the following steps:
- In Product Designer, revert the changes to affected audit schedules.
- To load the audit schedule properties file changes into Product Designer, restart Product Designer, logout, or change the changelist.
- Reopen the affected audit schedules.

Studio/Platform Issues

Chrome browser cannot display product documentation in HTML format (DOC-7251)

Issue – If you use the Google Chrome browser, you can view the HTML Guidewire product documentation only if it is served by an HTTP server using the `http://` protocol. The Chrome browser cannot load HTML product documentation from your local disk by using the `file://` protocol.

Workaround – Use a different browser, such as Microsoft Internet Explorer. If you use the HTML documentation only to access the PDF files, you can open them directly in the `pdf` subdirectory of the `doc` directory.

Alignment Property Does Not Work for Inputs in a DetailView (PL-29429)

Issue – Inputs in a detail view do not align to the right or center. They always align to the left, regardless of the value of the `align` property.

Workaround – If you need to right align a series of items such as a set of monetary values, put them into a list view instead of a detail view. You can then set the alignment as needed.

JMS inbound integration implementations must be OSGi or Java plugins (PL-32054)

Issue – Writing JMS inbound integration handlers requires casting an object to the type `javax.jms.Message`. In Gosu, for some app servers this fails with the message: "Caused by: gw.lang.parser.exceptions.ErrantGosuClassException: GosuClass test.GosuJMSMessageReplyHandler has errors, and cannot be used at runtime."

Workaround – Implement JMS handler code for inbound integration in Java. Guidewire recommends that for inbound integration Java code, implement your code as an OSGi plugin, not a standard (non-OSGi) Java plugin.

Compilation errors after removing a delegate (PL-34619)

Issue – When you remove a delegate, some internal code is not generated properly, resulting in compilation errors.

Workaround – In Guidewire Studio, on the **Codegen** menu, click **Generate Metadata Classes**.

Free-text search is not supported on WebLogic (PL-34871)

Issue – Free-text search is not supported on a WebLogic application server.

Workaround – Guidewire is aware of this issue.

You may also need to update your ETL code to reference the new tables.

Several permissions are defined incorrectly (PL-34946)

Issue – The following permissions are defined incorrectly:
- `perm.Holiday.edit`
- `perm.Holiday.delete`
- `perm.Note.view`

Workaround – To any role that contains the `holidaymanage` permission, add the `buswkmanage` permission to that role. To any role that contains the `noteview` permission, add the `noteedit` permission to that role.

Alt+Shift+L to reload PCF files does not work when the server is run from the command line (PLWEB-5819)

Issue – If you run the server from the command line with the command `gwb runServer`, then pressing `Alt+Shift+L` in the application interface does not reload PCF files.

Workaround – Run the server from Guidewire Studio instead.

Server Tools downloads may time out (PLWEB-6184)

Issue – On the **Server Tools** pages, some downloads may take too long and generate a timeout error.

Workaround – Increase the value of the `WebUIAJAXTimeout` configuration parameter.

Parenthesis and RuleOperator typelists are not localizable (BIZ-822)

Issue – The `Parenthesis` and `RuleOperator` typelists are not localizable, and any localization causes Gosu compilation to fail.

Workaround – Guidewire is aware of this issue.

Import fails if source and target business rules data models do not match (BIZ-862)

Issue – During a business rules export and import operation from one system to another (for example, from a test system to a production system), the import fails if the business rules data model does not match between the two systems.

Workaround – Ensure that the business rules data model of the source and target systems is the same before exporting any business rule. If necessary:

1. Modify the business rules data model of one system so that it matches the business rules data model of the other system.
2. Regenerate the business rules export file.
3. Repeat the business rules import operation.

Guidewire PolicyCenter 8.0.6 Release Notes

Release 8.0.6

Overview

These release notes contain the following:
- "Release Information" on page 317
- "Installing This Release" on page 318
- "Support" on page 318
- "Major Issues and Changes" on page 318
- "Improvements and Resolved Issues" on page 319
- "Known Issues and Limitations" on page 324

Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

IMPORTANT If you skipped one or more upgrade releases to PolicyCenter, be sure to read the release notes for those releases to learn about changes and fixed issues.

Version Number

This release of Guidewire PolicyCenter is 8.0.6.
- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 7.0.1 or later. The latest maintenance release is preferred.

  If you are implementing multiple settlement currencies, BillingCenter 8.0.2 or later is preferred.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 7.0.0 or later. The latest maintenance release is preferred.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 7.0.7, 8.0.1 or later. The latest maintenance release is preferred.

Installing This Release

Refer to the following:

- The *PolicyCenter Installation Guide* for general installation information
- Prior PolicyCenter release notes for any versions that you have skipped

Upgrade Information

Incorrect warning message when running Configuration Upgrade Tool (PL-28723)

When upgrading from one PolicyCenter 8 release to another PolicyCenter 8 release, running the Configuration Upgrade Tool may produce following warning message:

```
WARN cannot find Emerald base configuration zip, this could indicate a problem...
```

If the `PolicyCenter/modules/base.zip` file does exist, then you can safely ignore this warning.

Retaining Product Designer configuration data from previous releases (PC-30768)

By default Product Designer stores its configuration data in a subdirectory of its parent directory:

The configuration data includes information about users, change lists and workspaces that are associated with this installation of Product Designer. You can choose a different location for the configuration data by editing the `launchPD.bat` file.

If you want to reinstall or upgrade Product Designer but keep the existing configuration data, then create a separate configuration directory outside the PolicyCenter directory structure. After you reinstall or upgrade Product Designer, edit `launchPD.bat` to point to the separate configuration directory. See the *Product Designer Guide* for more information.

> **IMPORTANT** Commit your change lists before you reinstall or upgrade Product Designer. If you do not commit your changes then the new installation of Product Designer may not be able to read the existing change lists.

Support

For assistance, visit the Guidewire Community.

Guidewire Customers

- `https://community.guidewire.com`

Guidewire Partners

- `https://partner.guidewire.com`

Major Issues and Changes

This topic describes major issues and changes that can affect your installation.

New and Changed Features

For information on new features and other changes, see the topic "New and Changed in PolicyCenter 8.0.6" in the *New and Changed Guide*.

Base PCF File Changes

The link below requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 8.0.5 to 8.0.6

To view a report of the changes to the base PCF files, `click here`.

Base Resource Changes

The link below requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 8.0.5 to 8.0.6

To view a report of the changes to the base resources, `click here`.

Changes in This Release Provided in Upgrade Diff Report

Guidewire provides a report detailing certain differences between the current release and your prior release. This report describes changes in display keys, entities, typelists, and the Gosu API. To obtain your custom Upgrade Diff Report, visit the Guidewire Community.

Improvements and Resolved Issues

This topic describes improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This list is not intended to be comprehensive.

- "PolicyCenter improvements and resolved issues" on page 319
- "Platform improvements and resolved issues" on page 322

PolicyCenter improvements and resolved issues

The following are the primary improvements and issues corrected in this release:

ID	Description
Account	
PC-23329	**Search Accounts** now works with Canadian postal code filter.
PC-25348	On the `MessageControlForDestinationScreen`, `MessageControlSearchItem` is now visible so that you can search for messages by account.
Billing / Payments	
PC-26683	The **Billing** screen now displays the billing information for the latest version of the policy. This prevents displaying outdated information for canceled policies.
Contacts	
PC-31150	Fixed an issue where PolicyCenter created contacts with the same ABUID. This resulted in the message: "There are multiple contacts with the same abuid". This also resulted in a `MultipleMatchesException` when ContactManager tried to update the contact.
PC-33438	When creating a new primary named insured in a future dated policy change, the address is now populated on the policy summary screen.
Diffs	
PC-32751	Fixed several issues where PolicyCenter made incorrect assumptions when trying to apply differences that happen on changes independent of the `EditEffectiveDate`.

ID	Description
PC-32768	Fixed an issue with multi-version policy transactions where comparing branches at their respective slices switched the two versions.
Financials	
PC-33426	Transaction calculator now creates $0 offset transactions if the transaction end dates are before or the same as corresponding onset transactions.
General	
PC-32545	Fixed an issue which caused a Slice Date outside of range error in `AccountImpl.copyAccountContactsByIterator` method.
PC-32571	Verifying duplicate transactions no longer uses the `PC_SEQUENCE` table. `TransactionID` is now passed to the Bundle `CommitOptions`, which has an associated purge batch process.
PC-32722	Fixed issue with the `isOverlap` method that occurred with neighboring intervals in arbitrary order in an XML file.
PC-33130	The `ScheduledItemAdapter` interface has been adapted for schedule multi-patterns.
PC-33652	Fixed a performance issue with Team Screens batch processing.
PC-33717	Fixed an issue where canceled terms overlapped in effective time for a single policy.
PC-34112	Fixed an issue where the parallel degree for newly created indexes on primary keys was not being reset after a database upgrade.
Globalization, Language Packs	
PC-25833	Fixed an issue where using the French Language Pack with PolicyCenter 8.0.2 caused the DBCC to use invalid table names, and hence throw errors.
PC-32538	Fixed an issue in the Japanese version. In rate table definitions, the **Matching Rule** column was not translated.
Integration	
PC-33225	Several product model patterns that have accessor methods to `getByRefCode` now return `null` if `null` is provided. These patterns have also been modified to avoid null pointer exception errors.
Jobs	
PC-23022	On the `StartPolicyChange` screen, the `EffectiveTimePlugin` is now applied after the user selects the date. This fixes issues with multiple policy changes in the same day becoming out-of-sequence.
PC-24024	After renewal, clicking **View Policy** now shows the policy on the effective date of the last viewable bound policy period.
PC-30236	**Audit Details** screen now displays prior final audit payroll for audit revisions.
PC-30612	Previously rescinded policies can now be cancelled and rescinded again.
PC-32228	During a new submission, if the selected account primary location does not specify a state, you cannot select a product associated with a policy line that works with territory code lookup.
PC-32836	Fixed an issue which caused conversion on renewal to fail for policies when using the WC7 template. For this fix to work, `WC7PolicyLineMethods.gs` must contain the following code: `override property get WorkersComp(): boolean { return true }`
PC-33256	The hard coded string in parameter of `startChecksFor` method in `AuditProcess`, `CancellationProcess`, `IssuanceProcess`, `JobProcess`, `PolicyChangeProcess`, `RenewalProcess`, and `SubmissionProcess` are now localized.
Line of Business - Personal Auto	
PC-31987	Fixed a previously known issue with the `NumDaysInCoverageRatedTerm` getter in personal auto.
Line of Business - Workers' Compensation	

ID	Description
PC-31441	Workers' compensation class codes that contain letters no longer cause errors during quote.

Product Designer

PC-32592	Previously Product Designer could not handle system table entities that contained `java.lang.Long` data types backed by `java.lang.Long`. This has been updated so that you may now use the Product Designer system table editor to manage system tables backed by a long.

Performance Improvement

PC-34034	Improved performance of product model synchronization by adding an API that enables additional caching during product model synchronization.

The following Gosu classes now use the new API:

- `gw.job.JobProcess`
- `gw.validation.PCValidationContext`
- `gw.web.productmodel.ProductModelSyncIssuesHandler`

If you have modified these classes, merge the changes into your code. If you have implemented your own methods for product model synchronization or validation, consider using the new API in your code, and refer to these updates for examples.

The new API, available in Gosu through class `gw.api.productmodel.AvailabilityContextScope`, must only be used in situations like product model synchronization, where the base policy data accessed by `lookuptables.xml` cannot change.

When using this API, always use a Gosu using clause to wrap product model synchronization code. For example, `ProductModelSyncIssuesHandler#syncWithIssues` now sets up a new `AvailabilityContextScope` before calling its `addIssues` block:

```
private static function syncWithIssues(helper : JobWizardHelper,
addIssues
    (issues : ArrayList<ProductModelSyncIssueWrapper>) ) {
  var issuesList = new ArrayList<ProductModelSyncIssueWrapper>()
    using (new AvailabilityContextScope ()) {
      addIssues(issuesList)
    }
    if(helper != null){
      filterAndDisplayIssuesAfterSync( issuesList, helper )
    }
  }
```

Policy File

PC-33825	The `MonetaryAmountInput` on the **Policy Info** screen now sets currency to `policyPeriod.PreferredSettlementCurrency`.

Rating

PC-31401	Added localized `CascadedLookup` value and dates formatting in **Rate Book Merge** popup.
PC-32176	Fixed a previously known issue where merged rate books lost references to rate tables. Tables in merged rate books now reference the rows if the source rate book has active status, and own the rows for all other tables.
PC-32544	A rate routine can now reference rate tables which are accessed by cascaded lookup, that is, the rate routine and the rate table data can be in different rate books. During export, the rate table definition for any referenced table is included even if the table data is owned by a different rate book.
PC-33064	When two rate books are merged, the new merged book retains the reference links for rate tables from the two original books if those references are to active rate tables. In prior releases the new merged book contained a copy of all the table data.
PC-33565	Fixed an issue where promoting a rate book caused `IllegalStateException` if two policy lines were using the same rate table.

Search

ID	Description
PC-32166	In `AddressQueryBuilder`, restriction methods have changed slightly to provide a more consistent interface: • `withAddressLine1Kanji` restriction changed from `StartsWith` to `EqualsIgnoringCase`. There is also now a new method, `withAddressLine1KanjiStarting` that restricts `StartsWithIgnoringCase`. • `withAddressLine2Kanji` restriction changed from `StartsWith` to `EqualsIgnoringCase`. There is also now a new method, `withAddressLine2KanjiStarting` that restricts `StartsWithIgnoringCase`. • `withCityKanji` restriction changed from "`Equals` to `EqualsIgnoringCase` • `withCityKanjiStarting` restriction changed from `StartsWith` to `StartsWithIgnoringCase`.
PC-32225	Fixed several issues in `ContactQueryBuilder`. The `withFirstNameKanji` method now uses the `EqualsIgnoringCase` restriction instead of `StartsWith`. There is also a newly added `withFirstNameKanjiStarting` method that uses the `StartsWithIgnoringCase` restriction. Similar changes were made for `LastNameKanji`, `CompanyNameKanji`, `Particle`, `CityDenorm`, `CityKanjiDenorm`, and `PostalCodeDenorm`. Lastly, on the `PersonNameRestrictor` enum, the `FirstOrLast` element has been fixed.
PC-33796	Fixed a problem where the character sequence ## caused Solr Batchload to fail if encountered as part of the data being loaded.
PC-33797	Fixed a problem where invalid command line options were passed to external sort on AIX during Solr Batchload.
UI Configuration	
PC-23392	In the `ReferenceValueCurrency` column on `UWApprovalLV.pcf`, the visibility of the header and visibility of the cells is now the same. This fixes an issue where the header and the cells were not aligned.
PC-23424	The **Retry** button on `MessageSearch.pcf` now works as expected.
PC-33163	The `OnEnter` method for each step in the submission wizard is now executed only once.
User Tools (Activities, Notes, Documents)	
PC-22681	The link to view or edit an existing activity is on the subject field. If no subject was specified on an activity, you could not view or edit the activity. The subject field for the create/edit activity is now mandatory.
PC-23893	Fixed an issue which caused a null pointer exception when uploading changes to a document.

Platform improvements and resolved issues

The following are the primary improvements and issues corrected in this release:

ID	Description
Clustering	
PL-34221	Improved the security of Guidewire cluster operations using JGroups by minimizing the possibility of deserialized Java objects being used to perform remote code invocation attacks against a Guidewire application server. By default, Guidewire applications disallow any Java classes placed on a blacklist to be deserialized. If desired, use the `SerializationWhitelistEnabled` configuration parameter to place certain Java classes on a whitelist to be deserialized. Server Tools **Serialization Info** (under **Info Pages**) provides a means of monitoring the deserialization of Java classes.
Consistency Checker	
PL-33402	The **Consistency Checks Info Page** now provides a **Download Errors** button to download only the checks that had errors. This type of download eliminates reporting on the checks that succeeded, thus, speeding up the download process.
Database Instrumentation	
PL-32449	Fixed several issues that caused Null Pointer exceptions while downloading the Server Tools Oracle AWR reports.

ID	Description
PL-33517	Fixed an issue with the Oracle Statspack download where it did not show queries under various categories.

Database Support - Oracle

PL-30941	Guidewire applications now use Oracle JDBC driver 12.1.0.2.0, which fixes several `Protocol violation` exceptions.

Document Management

PL-32872	Fixed a document production issue where any `ContextObject` parameter that is not an entity or a typelist was treated as if it were of type `String`.

Entities/Metadata

PL-29536	Resolved an issue where GUnit tests that created an entity would fail because no implicit bundle was passed in.
PL-34196	Fixed a performance issue with `DistributedKeyGenerator.getNextID`.
PL-34313	Fixed an issue with slow performance when sorting by exposure on the **Workplan** screen.

Logging

PLWEB-5876	To improve the tracking and debugging of `CSRF token not found in the request` error messages in the application log, the error message now contains session and user information.

Other - Cloud

PL-34270	Upgraded to a release of the Apache Commons Collections library that fixes a security vulnerability.

PCF - Layout - List View

PLWEB-5673	Resolved an issue where columns that were hidden would reappear after the page changed to Edit mode.
PLWEB-5715	Fixed an issue where if you have a List Detail Panel showing the details of a newly created bean in a List View, and then you click **Cancel**, the exception `IllegalStateException: Attempt to access bean with null bundle` would be thrown.

Profiling

PL-33521	Improved the profiling results of the (Server Tools) Guidewire Profiler. The Profiler now merges data collected on background threads (such as message destination sender threads) into the overall visible profiling results.

Queries

PL-34312	The query builder API no longer generates table aliases for subselects that exceed the database limit for name length.

Security

PL-33622	Upgraded Apache WSS4J to version 1.6.18.

Web - UI/Runtime

PLWEB-1788	Addressed an issue where certain `ExitPoint` windows, such as the **About** or **Help** windows, were hidden behind the main browser window in Internet Explorer. This issue is partially due to Internet Explorer behavior, but this release minimizes the likelihood that it occurs.
PLWEB-2669	Updated the keyboard shortcuts documented on the **Help** pop-up.
PLWEB-5580	Resolved an issue with the BingMap implementation and its use of the CSRF token.
PLWEB-5616	Fixed an issue where selecting multiple rows in two separate List Views and then clicking **Remove** would remove the rows from both List Views. Instead, only the rows in the List View containing the clicked button are removed.
PLWEB-5953	Fixed an issue where a JavaScript error occurred on the Japanese Imperial Date widget when the Regional Format was changed.

Web Services - WSI (New)

ID	Description
PL-34336	Introduced the new annotation @WsiReduceDBConnections, which limits the number of connection requests that a web service call makes to the database connection pool. This annotation is not generally needed, and using it might cause a faulty or slow external web service to keep database connections active and use up the connection pool. Use this annotation with caution, as it defeats the purpose of the connection pool.

Known Issues and Limitations

This section describes known issues with this release of Guidewire PolicyCenter.

> **Note:** For maintenance releases, Guidewire often defers fixing configuration issues that require merging files during the upgrade. Workarounds to many of these issues are listed in the following topics. The goal of this policy is to make upgrades as straightforward as possible.

PolicyCenter Known Issues

Provide a way to filter out added and removed subentities in a policy differences tree (PC-7929)

Issue – Need a way to filter out the children of an added or removed entity in a policy differences tree.
Workaround – Guidewire is aware of this issue.

When using parallel sessions, approvals made by underwriter are not updated for second user simultaneously viewing same work order (PC-12322)

Issue – Approvals made by an underwriter are not updated for a second user simultaneously viewing the same work order. This occurs in the following sequence of events:

- An agent processes a submission and issues arise
- The agent calls an underwriter, and remains in the policy
- The underwriter approves the issue or issues
- The agent refreshes the **Risk Analysis** screen by going to a different wizard step and returning to the **Risk Analysis** screen

After this sequence of events:

- The new approvals do not appear on the agent's screen
- However, the agent can bind or issue the policy (based on the new approvals by the underwriter)

Workaround – The agent can leave the policy and return to it. The approvals appear as expected. Alternatively, the agent can see the approvals by clicking the **Details** button when issuing the policy.

Must enter VIN when moving from quick quote multi-version or side-by-side to full application (PC-13099)

Issue – In a quick quote, you do not have to enter the VIN or other fields that are required for a full application submission. When converting a quick quote with multi-version or side-by-side quoting to full application, to resolve validation issues, you must enter the VIN for each of the versions. In a job with side-by-side quoting, this issue can occur with fields that are not base data.
Workaround – Add code that copies this type of data to the other versions.

PolicyCenter does not send renewal billing instruction to BillingCenter (PC-15082)

Issue – On receiving payment notification from BillingCenter, PolicyCenter tries to bind. If the policy has a renewal transaction in draft status, PolicyCenter does not bind and send the Renewal Billing Instruction to BillingCenter. The policy might be in draft status after an underwriter edited something on the policy. This is an issue with BillingCenter 3.0.1 and later.

Workaround – Guidewire is aware of this issue.

BillingCenter integration: Automatic payment instrument is used to create invoice stream even if final selection is manual (PC-15307)

Issue – If an automatic payment instrument is setup and selected, the policy will create a new invoice stream with this payment instrument even if the user changes to Manual before issuing the policy.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Guidewire is aware of this issue.

BillingCenter integration: Removing alternate billing account when payment instrument selected generates WsdlFault (PC-15308)

Issue – If an automatic payment instrument is selected when using an Alt Billing Account to create a new policy, PolicyCenter throws a `WsdlFault` if the Alt Billing Account gets removed.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Guidewire is aware of this issue.

BillingCenter integration: New invoice streams not created when issuing policy with alternate billing account on new account (PC-15309)

Issue – Assume an account has an existing invoice stream. When this account is used as an **Alt Billing Account** for new policies on new accounts, a new invoice stream is not created when the new policy's new invoice stream periodicity matches the invoice stream on the **Alt Billing Account**.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Guidewire is aware of this issue.

Renewal processing for renewal conversion assumes BasedOn is populated (PC-15851)

Issue – The `startNewRenewal` method in `gw.webservice.pc.pc700.job.PolicyRenwalAPI` assumes the existence of a prior term or a `basedOn` version. Unless customized, this method can cause problems in certain convert-on-renewal scenarios. This method has been deprecated.

Workaround – If you are developing convert-on-renewal flows, use the `gw.webservice.pc.pc700.job.startConversionRenewal` method.

SQL Query performance in free-text search (PC-18794)

Issue – There are open issues related to the performance of SQL query in free-text search. This occurs the first time that you load the PolicyCenter data into the Solr index database.

Workaround – Guidewire is aware of this issue.

Cannot quote a renewal if the expiring policy term has a handled preemption (PC-19416)

Issue – In Commercial Package Policy in the base configuration, if a policy term has a preemption that has been handled, when a renewal job is created, the **Edit Workorder** button is not always available on the renewal. When this occurs, the only options are to bind or not take the renewal.

Workaround – Guidewire is aware of this issue.

Copy-data with an out-of-range date (PC-20279)

Issue – When working with the copy data feature, if you specify a specific date to copy data from and that date is out of range for the policy, PolicyCenter presents a warning that no applicable policy version could be found. However, the date selector still indicates the requested date, not the date that PolicyCenter will use for the copy if you choose to proceed.

Workaround – As appropriate, modify the user interface to reset the date selector or to hide the ability to select data to copy.

Error when quoting after adding new location (PC-25559)

Issue – In the Commercial Auto, General Liability, and Commercial Property lines, users sometimes receive an error message after clicking **Quote** on the **Locations** screen indicating that there are no locations even when there are. This error could potentially occur in other lines of business and on other list views, depending on how they are configured.

Workaround – Click **Next** and then **Back** to navigate away from and back to the **Locations** screen before quoting. Alternately, remove the client side validation on the number of rows.

Reinsurance warnings not refreshed even after saving a draft (PC-21720)

Issue – When working with reinsurance, PolicyCenter displays validation errors and warnings both at the top and in the worksheet at the bottom of the screen depending on the action that caused validation to run (typically **Save**, or leaving the screen). Depending on the user flow, one or both of these areas may display stale messages, and the messages may not be the same.

Workaround – Refresh the messages in either or both locations before taking the action again.

Gosu classes not intended for use (PC-23009)

Issue – Some Gosu classes are visible in Studio, but are not intended for use. You can distinguish these Gosu classes because they have no visibility annotations (neither @Export nor @ReadOnly) and they are not in a gw package.

Workaround – Do not use these methods in configuration. The methods are unsupported and may change or be withdrawn without notice.

Billing system integration startup error when creating carrier organization (PC-23211, PC-15682)

Issue – When PolicyCenter is first started it creates a carrier organization as part of the start up process. The carrier organization by definition is also a producer. PolicyCenter sends a message to the billing system when a producer is created so that the producer can also be created in the billing system. However at the point in start up when the carrier organization is created, the messaging system has not been started. Therefore no create producer message is sent. When subsequent changes are made to the carrier organization an update message is sent to the billing system to update the carrier organization. This may result in errors being generated in the billing system.

PolicyCenter reports the following error:

```
Retryable Error at startup: Cannot find producer to update with publicid: systemTables: 1
```

In the case of PolicyCenter to BillingCenter integration BillingCenter responds to the update message from PolicyCenter with the error message:

```
BadIdentifierException: Producer with publicId systemTables:1 was not found
```

Workaround – Guidewire is aware of this issue.

PolicyCenter allows owner and overriding-payer accounts to be the same (PC-23570)

Issue – In PolicyCenter, you can set the overriding payer on a list bill policy to the same account as that account which owns the policy. You set the overriding on the **Account→Billing** screen. Systems which receive data from PolicyCenter may report this as an error. For example, BillingCenter requires an overriding payer account to be different than the owner account.

Workaround – Add validation or other logic to PolicyCenter that ensures that, if present, an overriding-payer account is different than the owner account.

Unbound renewal policy transactions can provide invalid dates (PC-25288)

Issue – If a policy renewal is in the renewal process and the user goes from the renewal job wizard to the policy file summary, the as-of date default in the **Date** field of the **Actions** panel will not be entirely valid, as it will be the editEffectiveDate of an unbound period from an incomplete job.The quote, reinsurance, and audit schedule, if enabled, screens display exceptions because appropriate periods do not exist on the specified dates.

Workaround – Edit `JobWizardInfoBar.pcf` to filter out any `PolicyPeriod` objects that cannot provide the `editEffectiveDate`.

Billing integration error when final audit is waived from Audit job (PC-25500)

Issue – When the user waives a final audit from within a running Audit job, the BillingCenter integration sends an incomplete message to BillingCenter. The message does not include the term number of the policy period. BillingCenter needs the term number to identify the BillingCenter policy period that needs the change.

Workaround – Guidewire is aware of this issue.

Insurance Suite integration issues (PC-25732, PC-23873)

Issue – There are issues with creating and sharing contact information between PolicyCenter, BillingCenter, and ContactManager if the applications are different versions. For example, these issues arise if you have PolicyCenter 8.0 integrated with BillingCenter 7.0. These issues include:

1. When PolicyCenter 8.0 is integrated with BillingCenter 7.0, contacts sent to BillingCenter have a null `AddressBookUID` even though PolicyCenter has called ContactManager with the suggested `UniqueID`. This results in BillingCenter also creating the contact in ContactManager, leading to duplicate contacts.

2. When BillingCenter creates a new Contact for an Account that had been created in PolicyCenter, PolicyCenter does not recognize the Contact and throws a `BadIdentifierException`.

Workaround – Guidewire is aware of these issues. In the Knowledge Base, see article 3183.

In addition, in the `bc700` package, update the sync method in the `PCContactInfoEnhancement` class to match the version in the `bc800` package.

Rate routine editor message needs notification of parameter change (PC-30084)

Issue – The rate routine editor does not display message if rate table definition parameter is modified.

A rate table definition is referenced in a rate routine. You modify a parameter in the rate table definition. When you return to the rate routine, the rate routine editor does not display a message warning that it is unable to reconcile the definition of rate table.

If you select a rate table, the rate routine editor does not warn you that parameters have been modified in the rate table. You are not given the option to accept this change or cancel.

Workaround – Guidewire is aware of this issue.

Import rate table from spreadsheet fails (PC-30367)

Issue – On the **Rate Book Details** screen, several issues cause import rate table from spreadsheet to fail:

1. If values in the spreadsheet exceed the precision for a target column. Errors are logged in the spreadsheet validation error file. In the error file, the value is rounded to the precision of the PolicyCenter rate table column.

2. If data entered into a spreadsheet does not match the data type of the PolicyCenter rate table column. For example, if text is entered into a spreadsheet cell when the PolicyCenter rate table column expects a decimal value. In the spreadsheet validation error file, the value is null when the data is not the expected type.

Workaround – To workaround these issues, fix problems logged in the validation error file:

1. After import fails, fix the data values with precision errors, and try importing again.

2. Fix data type mismatches in the spreadsheet, and try importing again.

Payment screen not updated (PC-30594)

Issue – Updating a payment plan in BillingCenter does not update **Premium Summary**, **Payments**, and **Invoicing** fields on the PolicyCenter **Payment** screen. Even making a selection in the payment plan list view does not update these fields.

Workaround – To refresh these fields, navigate away from and return to the **Payment** screen.

PolicyPeriod.removeFromForms method causes DBCC error (PC-30947)

Issue – The `PolicyPeriod.removeFromForms` method causes a DBCC error.

Workaround – Guidewire is aware of this issue. Do not use this method.

Comparing policy transactions does not work correctly with out-of-sequence transactions (PC-31190)

Issue – When comparing two policy transactions from the Policy Transactions screen, any changes included in the policy revisions being compared that had been made effective on a later date than the edit effective date of the associated transaction (either within that transaction or in a previous transaction) are not displayed in the comparison. Changes made on earlier dates are included.
Workaround – Guidewire is aware of this issue.

Cannot complete unowned activities (PC-31610)

Issue – Cannot complete unowned activities even if you have the correct permissions. This issue is in PolicyCenter.
Workaround – Guidewire is aware of this issue.

Personal Auto coverages are not synchronized correctly (PC-32284)

Issue – In the base configuration, personal auto coverages are not synchronized correctly in the Personal Auto policy line. Initial Personal Auto coverages on vehicles are synchronized only when a user goes to the **PA Coverages** screen. At that point, the coverages get added as of the date that the slice selector is on. This means that the coverages could be added on an effective date later than the vehicle was added to the policy. At a later time, the coverages are added again on the effective date of the vehicle. These coverages are propagated forward to other slices, overwriting changes made to coverages or coverage term options when they were initially added.
Workaround – Change the product model synchronization logic for Personal Auto so that the initial vehicle coverages are added at the time the vehicle is added.

Policy quote cloning missing sequencing (PC-33297)

Issue – In quote cloning, the newly created and cloned `PolicyPeriod` does not have a link to the original `PolicyPeriod` and hence is not discoverable. In addition, `PolicyPeriod` lacks a sequence number so that external systems do not know the creation order of the clones.
Workaround – Create an enhancement `PolicyPeriodQuoteCloneBasedOnPublicID_Ext`. In this enhancement, populate the `PublicID` from the originating `PolicyPeriod.PublicID` using the `Job.Periods` array where the `BranchNumber` is equal to `Clone.BranchNumber`. This information is in the GXModel. This will work for submissions and all other job types.

Upgrade to PolicyCenter after running ETL Product Model Loader extension pack (DOC-10285)

Issue – Upgrading to PolicyCenter 8.0.3 or later after running ETL Product Model Loader extension pack may require a manual drop of some tables.
In PolicyCenter 8.0.3, the ETL Product Model Loader Extension Pack functionality moved into the core PolicyCenter product. The extension pack created ETL database tables with `pcx_` as the prefix. PolicyCenter 8.0.3 and later create ETL database tables with `pc_` as the prefix. If you ran this extension pack with a prior version of PolicyCenter, you will have stale tables after upgrading to PolicyCenter 8.0.3 or later. The upgrade does not fail, but the server detects extra tables in the database.
Workaround – To remove these stale tables, manually drop the old `pcx_` tables created by the ETL Product Model Loader extension pack. The ETL Product Model Loader Extension Pack created these tables:

- `pcx_etlclausepattern`
- `pcx_etlcovtermpattern`
- `pcx_etlcovtermoption`
- `pcx_etlcovtermpackage`
- `pcx_etlpackterm`
- `pcx_etlmodifierpattern`
- `pcx_etlratefactorpattern`

You may also need to update your ETL code to reference the new tables.

Product Designer Known Issues

Cannot use Product Designer to edit an extended object with no base object (PC-27763)

Issue – If you attempt to edit a part of the product model that is based on an entity extension file (`.etx`), and that extension has no base entity file (`.eti`), Product Designer displays errors and refuses to display the contents of the file.

Although Guidewire best practices recommend that you never create an entity file extension (`.etx` file) when no base entity (`.eti` file) exists, Guidewire Studio did not enforce this recommendation. However, the current version of Product Designer does not allow the use of an entity extension without a corresponding entity base file. If such an entity extension exists, Product Designer displays a **NullPointerException** when you attempt to access the affected objects. You then can use the Back command in the browser to view other pages, but you cannot configure the affected objects.

Workaround – Locate all `.etx` files that do not have corresponding `.eti` files, and rename them with an `.eti` extension.

PolicyCenter does not start in single currency display mode if the product model is configured to use multiple currencies (PC-27829)

Issue – Product Designer allows users to configure multicurrency features without regard for the setting of the `MultiCurrencyDisplayMode` in the PolicyCenter `config.xml` file. If you configure multiple currencies in a product model object, and the value of the `MultiCurrencyDisplayMode` property is `SINGLE`, PolicyCenter does not start.

For example, if you add clauses that specify multiple coverage currencies and the PolicyCenter `MultiCurrencyDisplayMode` is set to `SINGLE`, the server fails to start and displays messages similar to the following:

```
ERROR ... PolicyLinePattern : ... [field : AvailableCoverageCurrencies] : - Multiple Coverage
Currencies defined for product not allowed in single Currency mode
ERROR ... [ExclusionPattern] : ... [DirectCovTermPattern] : ... [LimitsSet] : - The pattern specifies
limits for different currencies in SingleCurrencyMode
```

Workaround – If you intend to deploy a product model on PolicyCenter servers that are not configured for multiple currency display mode, do not configure multiple currencies in any part of the product model. However, you can test multicurrency products on a staging server configured for multicurrency while the production system is configured for a single currency.

System table cells display public IDs when referencing another system table (PC-28368)

Issue – If the cells in a particular column of a system table are referencing values from another system table, only the public IDs of those values are shown, not the values, when the system table is edited in Product Designer. In addition, users cannot search for values in these system table cells. The referenced system table, however, does not display the public ids so they cannot be viewed from within Product Designer.

Workaround – View the XML directly for the system table being referenced to determine the public IDs associated with the values that need to be added, edited or deleted. You can then use those public IDs directly.

An exception might appear when you run Product Designer (PC-28531)

Issue – An `IncompatibleRemoteServiceException` might appear if you run Product Designer 8.0.3 or later after running a previous version of Product Designer.

Workaround – Refresh the browser cache by selecting `Ctrl+F5`.

Invalid audit schedules when audit schedule properties file is changed while user is editing audit schedule (PC-28146)

Issue – If you edit an audit schedule in Product Designer at the time that the associated audit schedule properties file is updated, you may not see the properties file changes reflected in Product Designer. This could result in an invalid audit schedule (that matches the previous audit schedule properties file).

Workaround – Perform the following steps:

- In Product Designer, revert the changes to affected audit schedules.
- To load the audit schedule properties file changes into Product Designer, restart Product Designer, logout, or change the changelist.
- Reopen the affected audit schedules.

Studio/Platform Known Issues

Renaming method or property throws ParseResultsException (PL-16633)

Issue – If you rename a property or a method or you change a method signature, and a workflow references that property or method in a Gosu field, PolicyCenter throws `ParseResultsException`. This is the intended behavior.

Workaround – Restart the workflow engine. To do so:

1. Log in to PolicyCenter using an administrative account.
2. Access **Internal Tools→Reload**.
3. Click **Reload Workflow Engine**.

Long text in table cells can add white space to the right of the page (PL-28288)

Issue – In an editable list view, extremely long text entered in a single cell can cause additional white space on the right side of the page. Long text is text that occupies approximately the entire width of the screen. This issue occurs primarily in Chrome.

Workaround – If you expect users to enter large amounts of text into the cells of a column, configure the column to support text wrapping.

Administrative command-line tools cannot refresh WSDL (PL-29021)

Issue – Administrative command-line tools rely on web service implementation classes such as `MaintenanceToolsAPI.gs`. If you change the web service implementation classes, administrative tools might fail because the tool's WSDL does not match the server WSDL. Some changes do not affect the WSDL. For example, adding a `@WsiPermission annotation`.

Workaround – Do the following:

From the `PolicyCenter/bin` directory, at a command prompt type the command:

```
gwpc regen-soap-api
```

In Windows Explorer, copy the WSDL files from the location:

```
PolicyCenter/soap-api/wsi/wsdl
```

to:

```
PolicyCenter/admin/res/wsi
```

No entity changed event fired through a one-to-one entity relationship (PL-26224)

Issue – The existence of a one-to-one entity relationship between two entities prevents an entity change event from being fired, even if it is supposed to. For example, in ContactManager there is a foreign key from `ABContact` to `Address` for the primary address, and a one-to-one relationship from `Address` to `ABContact`. Because of the one-to-one link, no event changed event is fired for any change in the primary address.

More generally, if entity E1 has a one-to-one relationship to entity E2, no entity changed event is fired for E2 if a property on E1 changes.

Workaround – You might be able to take advantage of a secondary change caused by the initial change, with the secondary change triggering the event to be fired instead. In the previous ContactManager example, making sure

that changes to all properties in `Address` cause a `History` record to be generated ensures that the entity changed event will be fired.

Free-text Search indexes cannot be upgraded from 7.0 to 8.0 (PL-29062)

Issue – To upgrade free-text search in PolicyCenter from 7.0 to 8.0, you install the newer version of the Guidewire Solr Extension in 8.0 on top of the version from 7.0. Because the PolicyCenter data model also changes, the free-text search indexes from 7.0 become obsolete. Guidewire does not support migrating free-text search indexes from 7.0. to 8.0.

Workaround – For Oracle, upgrade the Guidewire Solr Extension by following instructions in the *Upgrade Guide*. Then, run the new free-text batch load command to rebuild the indexes based on the new data model, after you complete your database upgrade. For SQL Server, Guidewire is aware of potential performance problems running the free-text batch load command after you complete your database upgrade.

Chrome browser cannot display product documentation in HTML format (DOC-7251)

Issue – If you use the Google Chrome browser, you can view the HTML Guidewire product documentation only if it is served by an HTTP server using the `http://` protocol. The Chrome browser cannot load HTML product documentation from your local disk by using the `file://` protocol.

Workaround – Use a different browser, such as Microsoft Internet Explorer. If you use the HTML documentation only to access the PDF files, you can open them directly in the `pdf` subdirectory of the `doc` directory.

Alignment Property Does Not Work for Inputs in a Detail View (PL-29429)

Issue – Inputs in a detail view do not align to the right or center. They always align to the left, regardless of the value of the `align` property.

Workaround – If you need to right align a series of items such as a set of monetary values, put them into a list view instead of a detail view. You can then set the alignment as needed.

JMS inbound integration implementations must be OSGi or Java plugins (PL-32054)

Issue – Writing JMS inbound integration handlers requires casting an object to the type `javax.jms.Message`. In Gosu, for some app servers this fails with the message: "Caused by: gw.lang.parser.exceptions.ErrantGosuClassException: GosuClass test.GosuJMSMessageReplyHandler has errors, and cannot be used at runtime."

Workaround – Implement JMS handler code for inbound integration in Java. Guidewire recommends that for inbound integration Java code, implement your code as an OSGi plugin, not a standard (non-OSGi) Java plugin.

Document Assistant incompatible with Chrome browser version 45 and later (PL-33648)

Issue – The Guidewire Document Assistant is incompatible with Chrome browser version 45 and later. Chrome browser versions 45 and later no longer support the Netscape Plugin Application Programming Interface (NPAPI).

Workaround – Use the browsers Firefox or Internet Explorer.

Guidewire PolicyCenter 8.0.5 Release Notes

Release 8.0.5

Overview

These release notes contain the following:
- "Release Information" on page 333
- "Installing This Release" on page 334
- "Support" on page 334
- "Major Issues and Changes" on page 334
- "Improvements and Resolved Issues" on page 335
- "Known Issues and Limitations" on page 341

Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

IMPORTANT If you skipped one or more upgrade releases to PolicyCenter, be sure to read the release notes for those releases to learn about changes and fixed issues.

Version Number

This release of Guidewire PolicyCenter is 8.0.5.
- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 7.0.1 or later. The latest maintenance release is preferred.

  If you are implementing multiple settlement currencies, BillingCenter 8.0.2 or later is preferred.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 7.0.0 or later. The latest maintenance release is preferred.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 7.0.7, 8.0.1 or later. The latest maintenance release is preferred.

Installing This Release

Refer to the following:

- The *PolicyCenter Installation Guide* for general installation information
- Prior PolicyCenter release notes for any versions that you have skipped

Upgrade Information

Incorrect warning message when running Configuration Upgrade Tool (PL-28723)

When upgrading from one PolicyCenter 8 release to another PolicyCenter 8 release, running the Configuration Upgrade Tool may produce the following warning message:

```
WARN cannot find Emerald base configuration zip, this could indicate a problem...
```

If the `PolicyCenter/modules/base.zip` file does exist, then you can safely ignore this warning.

Retaining Product Designer configuration data from previous releases (PC-30768)

By default Product Designer stores its configuration data in a subdirectory of its parent directory:

The configuration data includes information about users, change lists and workspaces that are associated with this installation of Product Designer. You can choose a different location for the configuration data by editing the `launchPD.bat` file.

If you want to reinstall or upgrade Product Designer but keep the existing configuration data, then create a separate configuration directory outside the PolicyCenter directory structure. After you reinstall or upgrade Product Designer, edit `launchPD.bat` to point to the separate configuration directory. See the *Product Designer Guide* for more information.

IMPORTANT Commit your change lists before you reinstall or upgrade Product Designer. If you do not commit your changes then the new installation of Product Designer may not be able to read the existing change lists.

Support

For assistance, visit the Guidewire Community.

Guidewire Customers

- `https://community.guidewire.com`

Guidewire Partners

- `https://partner.guidewire.com`

Major Issues and Changes

This topic describes major issues and changes that can affect your installation.

New and Changed Features

For information on new features and other changes, see the topic "New and Changed in PolicyCenter 8.0.5" in the *New and Changed Guide*.

Base PCF File Changes

The link below requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 8.0.4 to 8.0.5

To view a report of the changes to the base PCF files, *click here*.

Base Resource Changes

The link below requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 8.0.4 to 8.0.5

To view a report of the changes to the base resources, *click here*.

Changes in This Release Provided in Upgrade Diff Report

Guidewire provides a report detailing certain differences between the current release and your prior release. This report describes changes in display keys, entities, typelists, and the Gosu API. To obtain your custom Upgrade Diff Report, visit the Guidewire Community.

Improvements and Resolved Issues

This topic describes improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This list is not intended to be comprehensive.

- "PolicyCenter improvements and resolved issues" on page 335
- "Product Designer improvements and resolved issues" on page 338
- "Platform improvements and resolved issues" on page 339

PolicyCenter improvements and resolved issues

The following are the primary improvements and issues corrected in this release:

ID	Description
Contacts	
PC-29791	`ContactMessageTransport` used to send `message.reportAck` even when it received a non-retryable error. It has now been changed to send `reportNotRetryableError`.
Data Model / Core Entities	
PC-30235	Made consistency checks for `LinkedAddress` searchable and runnable from the **Consistency Checks** screen.
Financials	
PC-30374	Fixed an issue which caused a date overlap between transactions. This in turn could cause $1 transactions for the affected date range and neighboring ranges. As a result of this change, a zero offset transaction may be emitted if the proration for that range rounds to zero. PolicyCenter needs these transactions in order to account for the entire date range. These zero transactions may appear on transaction display screens and be included in downstream integrations unless you configure those spots to filter the zero transactions.
Infrastructure	
PC-31395	(PL-33517) Fixed an issue with the Oracle Statspack download where it did not show queries under various categories.

ID	Description
PC-31396	(PL-32872) Fixed a document production issue where any `ContextObject` parameter that is not an entity or a typelist was treated as if it were of type `String`.
Integration	
PC-25733	Fixed a previously known issue in the `BillingMessageTransport` Gosu class. The commented-out code in the send method now compiles.
PC-30634	When updating an organization address from PolicyCenter, the change was not reflected in BillingCenter because organization contacts were never synced with ContactManager. PolicyCenter now sends a `ContactUpdated` event message directly from PolicyCenter to BillingCenter with organization contact changes.
PC-30986	Fixed an issue with the BillingCenter integration which resulted in sending duplicate policy change transactions to BillingCenter. The integration now uses the `PolicyPeriod` ID, message `EventName`, and message `DestinationID` as the transaction ID for duplicate checking.
Jobs (Policy Transactions)	
PC-26159 PC-26160 PC-26161	After Applying Changes for preemption and future renewals (as described in the *Application Guide*, "Applying Changes Details"), PolicyCenter displays a worksheet only if some changes could not be automatically applied. Prior to this fix, PolicyCenter applied the changes first, then generated the list of conflicts. This caused issues such as: removing an entity and applying it to a future branch appeared as a conflict, despite being successfully applied. With this fix, PolicyCenter now generates the conflict list before applying changes, thus generating a correct list of conflicts.
PC-30542	Previously, the SideBySide data copy configuration copied `PolicyPeriod.TermNumber`, which created problems when a selected side-by-side version was withdrawn, and its null `TermNumber` was copied to the surviving version. With this fix, `TermNumber` is always excluded from the SideBySide data copy.
PC-30752	(PL-33456) Resolved an issue where quoting a submission after changing the policy term type might cause an `IllegalStateException` error.
PC-30905	Applying changes to future bound renewals opened the `ViewChangeConflictsWorksheet` PCF file, even when there were no differences between branches. The worksheet is now displayed only if there are some differences, or if the difference is null, as in archiving.
PC-30917	The dates used for determining if jobs (policy transactions) are out of sequence now include all slice dates on the policy graph or revision. Previously, the set of dates included only the edit effective dates of jobs. This change allows you to add entities in slice mode within all policy transactions, including submissions and renewals, on dates other than the edit effective date of the job. Note: This affected all policy lines except for Workers' Compensation provided with PolicyCenter and with the Basic and Standards-based Templates extension packs.
PC-31233	Fixed an issue in a policy change in which two vehicles were added in the future on dates other than the edit effective date. Coverages were charged for incorrect policy period when the PA Coverage screen was visited for a future slice first.
PC-31235	Fixed an issue where making a change on an out-of-sequence endorsement, then undoing the change, did not always reverse the change on later slices.
PC-31671	Changed type of **Earned Premium** table cell on **Policy File Summary** screen to `TextCell` which allows changing field width.
Language Packs	
PC-23848	Fixed an issue with **Vehicle Rate Modifiers** columns on the **Policy Review** screen. If the column text is not visible, you can now increase the column width to view the entire column label.
PC-31402	Fixed an issue where the **Allow Multiple Values** label was not translated on the **New Rate Table** screen. There is now a display key `Web.Rating.RateTableDefinition.AllowMultipleValues` for localization.
PC-31404	Fixed an issue in the Japanese version. In rate table definitions, the **Matching Rule** column was not translated.
LOB	

ID	Description
PC-30932	Fixed an issue where an array of jurisdictions in state lookups caused an exception. This problem could occur in any line of business in which line level coverages are restricted to a single jurisdiction.
LOB - BOP	
PC-30334	Fixed a previously known issue in the base configuration, Businessowners line of business where quote validation required a justification if the `BOPConditionalSurcharge` modifier had a non-zero credit/debit. Added a **Justification** field.
LOB - WC	
PC-30845	Fixed an issue where PolicyCenter threw an exception when starting a Workers' Compensation final audit on a canceled policy. The exception occurred if the cancellation time was not equal to the period start time, and a rating period start date existed on cancellation date. The `RatingPeriodStartDate` entity is created with a normalized date (date with a time being equal to the period start time). With this fix, the date is now normalized before calling the `canAddRPSD` method in the `WCJurisdictionEnhancement` enhancement. The `canAddRPSD` method returns an error message if you cannot add a rating period start date.
PC-31255	Changing the anniversary date on the **Coverage** screen now triggers a `PostOnChange` to display the updated rating periods. That way, other widgets that `postOnChange` with DATA*ONLY* will not throw exceptions when updated.
Product Model	
PC-31078	Fixed an issue where product model startup verification failed if you installed PolicyCenter onto a file path containing top-level product model directories like `products` or `policylinepatterns`.
Rating	
PC-22301	To optimize query performance in PCF files, changed `HasElements` or `Count > 0` in query results to `!Empty`. `HasElements` and `Count` have to search all the results to get a count. `!Empty` just checks for at least one result.
PC-25227	Improved performance of interpolated rate table match.
PC-25351	In PolicyCenter, the proration value is only shown for costs with `ProrationMethod` set to `ProRataByDays` and not `Flat`.
PC-26278	Date ranges ending on February 28 of a leap year no longer return one day less than expected when PolicyCenter is configured to ignore leap days.
PC-29703	Fixed an issue where `StandardAmount` was not being correctly prorated or merged.
PC-30960	When exporting a rate routine to spreadsheet format, the `Instruction` column now indents rows nested inside IF statements.
PC-31066	Fixed an issue which prevented exporting rate books to spreadsheet because PolicyCenter generated duplicate sheet names. In the spreadsheet format, the sheet name must be unique and 31 characters or fewer. With this fix, PolicyCenter creates unique names for all sheets by appending a counter to the sheet name. The counter is in the format (1).
PC-31302	Removed the default implementation of the `NumDaysInCoverageRatedTerm` property from `AbstractRatingEngineBase`. This property is supposed to be overridden in configuration.
PC-31411	To be subject to accrual, a `Cost` has to be premium (standard or non-standard) `RateAmountType` and have a non-flat proration method. In earlier versions of PolicyCenter, the `ProrationMethod` had to be `ProRataByDays`. With this fix, two costs with the same `CostKey` can have different proration methods. In previous releases, this could cause exceptions at quote.
PC-32149	The representation used for rate routine references to `CovTerm` data has changed in this release. Existing rate routines will execute correctly, but if you edit a rate routine created in an previous version of PolicyCenter, you may see a warning when you validate. Clear the warning by reselecting the coverage term value.
Reinsurance	

ID	Description
PC-30464	Improved the performance of the reinsurance implementation. Improvements included data redesign, so there is an upgrade trigger that creates instances of a new entity called `RIRiskVLContainer`.
PC-30751	This fix avoids an edge case where the Reinsurance Management module is turned on mid-term, and later some preempted and preempting branches generate a different `RiskNumber` for the same `Reinsurable`. PolicyCenter now generates a `RiskNumber` even though the module is off.
PC-31248	Fixed an issue which prevented the `ReinsuranceConfigPlugin` from being able to consult policy dates when deciding whether to enable reinsurance for that policy.
PC-31318	The `Cost` used to retrieve the `Reinsurable` is now explicitly resolved to slice mode as of its `EffectiveDate` to avoid unexpected implicit resolution with `Cost.Coverable.CoverableReinsurable.Reinsurables`.
PC-31487	Fixed an issue where ceding premiums were not generated for removed risks when a new program was created.
Revisioning	
PC-31287	Fixed an issue where quote incorrectly used the currently selected slice data instead of the `EditEffectiveDate`. Modified the `QuoteProcess` constructor to: branch = jobProcess.branch.getSlice(jobProcess.branch.EditEffectiveDate)
PC-31288	Fixed an issue in the `PolicyPeriod.mergeDuplicateAdds` method. Duplicate adds were not correctly merged for new beans that were matches of each other.
PC-32006	In this release, there is a change to the `SliceDateRange` property on `PolicyPeriod`. In previous releases, this property was calculated by considering every edit effective date on the policy. In this release, the `SliceDateRange` calculation also includes the effective and expiration dates of the objects in the policy graph.
Search	
PC-30636	(Windows Server 2008) Fixed an issue that could cause the free-text batch load command to hang.
PC-30959	Fixed an issue in `PCSolrMessageTransport` which prevented index messages from being retried if the connection between PolicyCenter and Solr was interrupted.
Security	
PC-31679	(PL-33622) Due to the vulnerabilities CVE-2015-0226 and CVE-2015-0227 impacting Apache WSS4J, upgraded the WSS4J version from 1.0.1 to 1.0.3.
UI Configuration	
PC-25649	In an issuance policy transaction, the info bar now displays a policy number link that goes to the **Policy File Summary** screen.

Product Designer improvements and resolved issues

The following are the primary improvements and issues corrected in this release:

ID	Description
Availability	
PC-30189	Fixed an issue where clicking **Validate** a second time on the **Availability** screen caused the **Duplicate** button to appear. When clicked, Product Designer threw an exception.
Configuration	
PC-30768	By default Product Designer stores its configuration data in a subdirectory of its parent directory: The configuration data includes information about users, change lists and workspaces that are associated with this installation of Product Designer. You can choose a different location for the configuration data by editing the `launchPD.bat` file.

ID	Description
	If you want to reinstall or upgrade Product Designer but keep the existing configuration data, then create a separate configuration directory outside the PolicyCenter directory structure. After you reinstall or upgrade Product Designer, edit `launchPD.bat` to point to the separate configuration directory. See the Product Designer Guide for more information. Important: Commit your change lists before you reinstall or upgrade Product Designer. If you do not commit your changes then the new installation of Product Designer may not be able to read the existing change lists.
Offerings	
PC-31907	Fixed an issue that prevented marking multiple offerings as not-selected within one commit. This issue was introduced in Product Designer 8.0.4 and is now resolved.
Question Sets	
PC-28498	Fixed an issue that prevented you from using the mouse to select a row in the **Question Choices** list. This issue only occurred in the Google Chrome browser.
Workspace Management	
PC-28406	Product Designer has been modified such that if a workspace is marked as using perforce, but perforce is offline, the workspace and changelist are marked as invalid. Previously, Product Designer threw a system error and had no user flow for disabling the perforce integration.

Platform improvements and resolved issues

The following are the primary improvements and issues corrected in this release:

ID	Description
Appserver Support - WebLogic	
PL-33301	Resolved an issue that caused the Guidewire Profiler to not show operation names for RPC Web Services.
Archiving	
PL-32879	Resolved an issue with restoring entities in archived documents that contain cross-graph links in the archiving domain graph.
Database Support - General	
PL-31932	To speed up staging table loading, Guidewire has added the ability to drop deferrable indexes before the insert operation and recreate them afterwards.
Database Support - SQL Server	
PL-32898	Improved the performance of the production of the (Server Tools Info Pages) SQL Server DMV Snapshot by restricting the data extract to the top 400 queries.
Database Upgrade	
PL-22087	You can now change the precision and scale of a numeric column in the data model configuration and the server will update the database next time it starts. You must ensure that digits to the left and right of the decimal point are not lost due to the change.
Entities/Metadata	
PL-29955	Resolved an issue where a typekey with the same name as another retired typekey would sort incorrectly in a typelist.
PL-32640	Resolved an issue that caused `EffDatedBaseProxy#changeNonDefaultLocaleValue` to create duplicate slices.
Gosu	
PL-32552	Resolved an issue in Gosu in which, for rare cases, an overridden method is called on the superclass not the subclass.

ID	Description
Inbound Integration	
PL-33163	Resolved an issue that resulted in an error when changing server run level from `maintenance` to `multiuser`.
Messaging	
PL-31705	The link from a workflow to the `MessageHistory` table is now a soft link instead of a direct foreign key. This change supports improved purging of `MessageHistory` table.
PL-33144	To improve messaging database query performance on Oracle, messaging queries now limit result set size with a `rownum` clause.
Other	
PLWEB-5260	Resolved a cross-site scripting security issue in the scroll position.
PLWEB-5360	Added more explicit documentation about the Terms of Service to the `customer.js` file. Note that the modification or additional use of certain JavaScript APIs in the product is not permitted.
PCF - Layout - List View	
PLWEB-1774	Resolved an issue where using the arrow keys to scroll down in a wide list view would show excessive screen flickering.
PLWEB-4618	Changed table cell selection to not persist between page views.
PCF - Layout - Workspace and Worksheets	
PLWEB-4305	Resolved an issue where triggering an `ExitPoint` would lead to pending updates on worksheets—such as changes to user notes—being lost.
PCF - Navigation - Left Navigation Panel	
PLWEB-5393	Resolved an issue where the highlight in the **Action** menu would display incorrectly.
PCF - Navigation - Tab Bar - Quick Jump	
PLWEB-4431	Resolved an issue where incorrect text was displayed in the QuickJump box while typing a command.
Plugins	
PL-33095	Resolved `IllegalArgumentException` thrown by the `GosuPluginContainer.getResourceAsStream` method.
Revisioning	
PL-32970	Querying an `EffDated` entity no longer returns instances in temporary branches. To access `EffDated` entities in temporary branches, first call `withFindTemporaryBranches(true)` on the query object.
Search	
PL-32463	Added Boolean parameter `securetransport` to `solrserver-config.xml` for `http` and `cluster` server types. When set to `true`, HTTPS is used instead of HTTP for connections between the application and the Guidewire Solr Extension.
PL-32627	Upgraded `solr.xml` to Solr 4.x format, with supports core autodiscovery. The `gwpc solr` command generates `solr.xml` in the new format whenever it generates the `pc-gwsolr.zip` deployment file. If you previously modified your copy of `solr.xml`, you can continue to use it unchanged. When time permits, upgrade your custom copy to the new Solr 4.x format.
PL-32793	The `gw.solr.GWSolrServerMgr` class which methods `findServer` and `regsiterServers` is available again.
PL-33243	(Windows Server 2008) Resolved an issue that could cause the free-text batch load command to hang.
PL-33311	Upgraded Zookeeper from version 3.4.5 to 3.4.6 for the Guidewire Solr Extension, which is a component of free-text search. The upgrade resolves an issue with WebLogic if you configure the Guidewire Solr Extensions for high availability.
Staging Tables	

ID	Description
PL-32354	(Oracle) To improve performance, the `-integritycheckandload` option of the `table`*import* command now calculates row counts on production tables only if you specify the `-estimateorastats` command option. Otherwise, the `-integritycheckandload` option uses information in database statistics to calculate approximate row counts. Use the `-estimateorastats` option to load production tables that are empty or have very few rows.
PL-32546	Back-to-back integrity check and load operations now fail before actually running if the first loader fails during insert/select.
Web - Configuration	
PLWEB-2083	Restored the functionality of IME mode in Internet Explorer. Note that this mode is not available in other browsers.
Web - IE Support	
PLWEB-4198	Resolved an issue in PCF elements that have the attribute `download` set to `true`. When these attributes triggered an error in Internet Explorer, the browser would prompt to download a `.do` or `.json` file instead of displaying an error message. Now all browsers display the proper error message.
Web - Performance	
PLWEB-5278	Resolved a performance issue where searches with no returned values ran unnecessary, additional queries.
Web - UI/Runtime	
PLWEB-1922	Resolved an issue where the progress bar was not displaying correctly.
PLWEB-2093	Added a `pickWidth` property to the **Select** column in tables. This allows a wider column to support configurations with longer alternatives to the word "Select".
PLWEB-2123	Resolved an issue where the progress bar would display "-1%" instead of an accurate progress percentage.
PLWEB-4290	Improved the error message if the web UI times out when trying to contact the server. Also added the new configuration parameter `WebUIAJAXTimeout`, which specifies the number of seconds to wait before showing a timeout message. The default value is 600 (10 minutes).
PLWEB-4304	Improved and resolved issues with keyboard navigation through the **Actions** menu.
PLWEB-4554	Resolved an issue where the JavaScript action wasn't working for a `menuItem`.
PLWEB-4645	Resolved an issue where `textfield` and `typekey` cells in a `ListView` could show the wrong display name.
PLWEB-4690	Improved the security of the CSRF token by removing it from URLs.
PLWEB-5510	Resolved an issue where half-width characters in Japanese were not wrapping in Google Chrome.
XMLElement (and XSD types)	
PL-31865	Resolved compatibility issue with `WSS4J` and `xmlsec` Java libraries.
PL-32193	Improved XML sorting performance during serialization.
PL-33338	For improved security, XML documents cannot use DTDs. This affects XML parsed by WS-I web services or XSDs.

Known Issues and Limitations

This section describes known issues with this release of Guidewire PolicyCenter.

Note: For maintenance releases, Guidewire often defers fixing configuration issues that require merging files during the upgrade. Workarounds to many of these issues are listed in the following topics. The goal of this policy is to make upgrades as straightforward as possible.

PolicyCenter Known Issues

Provide a way to filter out added and removed subentities in a policy differences tree (PC-7929)

Issue – Need a way to filter out the children of an added or removed entity in a policy differences tree.
Workaround – Guidewire is aware of this issue.

When using parallel sessions, approvals made by underwriter are not updated for second user simultaneously viewing same work order (PC-12322)

Issue – Approvals made by an underwriter are not updated for a second user simultaneously viewing the same work order. This occurs in the following sequence of events:
 • An agent processes a submission and issues arise
 • The agent calls an underwriter, and remains in the policy
 • The underwriter approves the issue or issues
 • The agent refreshes the **Risk Analysis** screen by going to a different wizard step and returning to the **Risk Analysis** screen
After this sequence of events:
 • The new approvals do not appear on the agent's screen
 • However, the agent can bind or issue the policy (based on the new approvals by the underwriter)
Workaround – The agent can leave the policy and return to it. The approvals appear as expected. Alternatively, the agent can see the approvals by clicking the **Details** button when issuing the policy.

Must enter VIN when moving from quick quote multi-version or side-by-side to full application (PC-13099)

Issue – In a quick quote, you do not have to enter the VIN or other fields that are required for a full application submission. When converting a quick quote with multi-version or side-by-side quoting to full application, to resolve validation issues, you must enter the VIN for each of the versions. In a job with side-by-side quoting, this issue can occur with fields that are not base data.
Workaround – Add code that copies this type of data to the other versions.

PolicyCenter does not send renewal billing instruction to BillingCenter (PC-15082)

Issue – On receiving payment notification from BillingCenter, PolicyCenter tries to bind. If the policy is in a draft status, PolicyCenter does not bind and send the Renewal Billing Instruction to BillingCenter. The policy might be in draft status after an underwriter edited something on the policy. This is an issue with BillingCenter 3.0.1 and later.
Workaround – Guidewire is aware of this issue.

BillingCenter integration: Automatic payment instrument is used to create invoice stream even if final selection is manual (PC-15307)

Issue – If an automatic payment instrument is setup and selected, the policy will create a new invoice stream with this payment instrument even if the user changes to Manual before issuing the policy.
This is an issue when integrating with BillingCenter 7.0.0 and later.
Workaround – Guidewire is aware of this issue.

BillingCenter integration: Removing alternate billing account when payment instrument selected generates WsdlFault (PC-15308)

Issue – If an automatic payment instrument is selected when using an Alt Billing Account to create a new policy, PolicyCenter throws a `WsdlFault` if the Alt Billing Account gets removed.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Guidewire is aware of this issue.

BillingCenter integration: New invoice streams not created when issuing policy with alternate billing account on new account (PC-15309)

Issue – Assume an account has an existing invoice stream. When this account is used as an **Alt Billing Account** for new policies on new accounts, a new invoice stream is not created when the new policy's new invoice stream periodicity matches the invoice stream on the **Alt Billing Account**.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Guidewire is aware of this issue.

Renewal processing for renewal conversion assumes BasedOn is populated (PC-15851)

Issue – The `startNewRenewal` method in `gw.webservice.pc.pc700.job.PolicyRenwalAPI` assumes the existence of a prior term or a `basedOn` version. Unless customized, this method can cause problems in certain convert-on-renewal scenarios. This method has been deprecated.

Workaround – If you are developing convert-on-renewal flows, use the `gw.webservice.pc.pc700.job.startConversionRenewal` method.

SQL Query performance in free-text search (PC-18794)

Issue – There are open issues related to the performance of SQL query in free-text search. This occurs the first time that you load the PolicyCenter data into the Solr index database.

Workaround – Guidewire is aware of this issue.

Cannot quote a renewal if the expiring policy term has a handled preemption (PC-19416)

Issue – In Commercial Package Policy in the base configuration, if a policy term has a preemption that has been handled, when a renewal job is created, the **Edit Workorder** button is not always available on the renewal. When this occurs, the only options are to bind or not take the renewal.

Workaround – Guidewire is aware of this issue.

Copy-data with an out-of-range date (PC-20279)

Issue – When working with the copy data feature, if you specify a specific date to copy data from and that date is out of range for the policy, PolicyCenter presents a warning that no applicable policy version could be found. However, the date selector still indicates the requested date, not the date that PolicyCenter will use for the copy if you choose to proceed.

Workaround – As appropriate, modify the user interface to reset the date selector or to hide the ability to select data to copy.

Error when quoting after adding new location (PC-25559)

Issue – In the Commercial Auto, General Liability, and Commercial Property lines, users sometimes receive an error message after clicking **Quote** on the **Locations** screen indicating that there are no locations even when there are. This error could potentially occur in other lines of business and on other list views, depending on how they are configured.

Workaround – Click **Next** and then **Back** to navigate away from and back to the **Locations** screen before quoting. Alternately, remove the client side validation on the number of rows.

Reinsurance warnings not refreshed even after saving a draft (PC-21720)

Issue – When working with reinsurance, PolicyCenter displays validation errors and warnings both at the top and in the worksheet at the bottom of the screen depending on the action that caused validation to run (typically **Save**, or leaving the screen). Depending on the user flow, one or both of these areas may display stale messages, and the messages may not be the same.

Workaround – Refresh the messages in either or both locations before taking the action again.

Gosu classes not intended for use (PC-23009)

Issue – Some Gosu classes are visible in Studio, but are not intended for use. You can distinguish these Gosu classes because they have no visibility annotations (neither `@Export` nor `@ReadOnly`) and they are not in a `gw` package.
Workaround – Do not use these methods in configuration. The methods are unsupported and may change or be withdrawn without notice.

Billing system integration startup error when creating carrier organization (PC-23211, PC-15682)

Issue – When PolicyCenter is first started it creates a carrier organization as part of the start up process. The carrier organization by definition is also a producer. PolicyCenter sends a message to the billing system when a producer is created so that the producer can also be created in the billing system. However at the point in start up when the carrier organization is created, the messaging system has not been started. Therefore no create producer message is sent. When subsequent changes are made to the carrier organization an update message is sent to the billing system to update the carrier organization. This may result in errors being generated in the billing system.
PolicyCenter reports the following error:

```
Retryable Error at startup: Cannot find producer to update with publicid: systemTables: 1
```

In the case of PolicyCenter to BillingCenter integration BillingCenter responds to the update message from PolicyCenter with the error message:

```
BadIdentifierException: Producer with publicId systemTables:1 was not found
```

Workaround – Guidewire is aware of this issue.

PolicyCenter allows owner and overriding-payer accounts to be the same (PC-23570)

Issue – In PolicyCenter, you can set the overriding payer on a list bill policy to the same account as that account which owns the policy. You set the overriding on the **Account→Billing** screen. Systems which receive data from PolicyCenter may report this as an error. For example, BillingCenter requires an overriding payer account to be different than the owner account.
Workaround – Add validation or other logic to PolicyCenter that ensures that, if present, an overriding-payer account is different than the owner account.

Unbound renewal policy transactions can provide invalid dates (PC-25288)

Issue – If a policy renewal is in the renewal process and the user goes from the renewal job wizard to the policy file summary, the as-of date default in the **Date** field of the **Actions** panel will not be entirely valid, as it will be the `editEffectiveDate` of an unbound period from an incomplete job.The quote, reinsurance, and audit schedule, if enabled, screens display exceptions because appropriate periods do not exist on the specified dates.
Workaround – Edit `JobWizardInfoBar.pcf` to filter out any `PolicyPeriod` objects that cannot provide the `editEffectiveDate`.

Billing integration error when final audit is waived from Audit job (PC-25500)

Issue – When the user waives a final audit from within a running Audit job, the BillingCenter integration sends an incomplete message to BillingCenter. The message does not include the term number of the policy period. BillingCenter needs the term number to identify the BillingCenter policy period that needs the change.
Workaround – Guidewire is aware of this issue.

Insurance Suite integration issues (PC-25732, PC-23873)

Issue – There are issues with creating and sharing contact information between PolicyCenter, BillingCenter, and ContactManager if the applications are different versions. For example, these issues arise if you have PolicyCenter 8.0 integrated with BillingCenter 7.0. These issues include:

1. When PolicyCenter 8.0 is integrated with BillingCenter 7.0, contacts sent to BillingCenter have a null `AddressBookUID` even though PolicyCenter has called ContactManager with the suggested `UniqueID`. This results in BillingCenter also creating the contact in ContactManager, leading to duplicate contacts.
2. When BillingCenter creates a new Contact for an Account that had been created in PolicyCenter, PolicyCenter does not recognize the Contact and throws a `BadIdentifierException`.

Workaround – Guidewire is aware of these issues. In the Knowledge Base, see article 3183.

In addition, in the `bc700` package, update the sync method in the `PCContactInfoEnhancement` class to match the version in the `bc800` package.

Rate routine editor message needs notification of parameter change (PC-30084)

Issue – The rate routine editor does not display message if rate table definition parameter is modified.

A rate table definition is referenced in a rate routine. You modify a parameter in the rate table definition. When you return to the rate routine, the rate routine editor does not display a message warning that it is unable to reconcile the definition of rate table.

If you select a rate table, the rate routine editor does not warn you that parameters have been modified in the rate table. You are not given the option to accept this change or cancel.

Workaround – Guidewire is aware of this issue.

Import rate table from spreadsheet fails (PC-30367)

Issue – On the **Rate Book Details** screen, several issues cause import rate table from spreadsheet to fail:
1. If values in the spreadsheet exceed the precision for a target column. Errors are logged in the spreadsheet validation error file. In the error file, the value is rounded to the precision of the PolicyCenter rate table column.
2. If data entered into a spreadsheet does not match the data type of the PolicyCenter rate table column. For example, if text is entered into a spreadsheet cell when the PolicyCenter rate table column expects a decimal value. In the spreadsheet validation error file, the value is null when the data is not the expected type.

Workaround – To workaround these issues, fix problems logged in the validation error file:
1. After import fails, fix the data values with precision errors, and try importing again.
2. Fix data type mismatches in the spreadsheet, and try importing again.

Payment screen not updated (PC-30594)

Issue – Updating a payment plan in BillingCenter does not update **Premium Summary**, **Payments**, and **Invoicing** fields on the PolicyCenter **Payment** screen. Even making a selection in the payment plan list view does not update these fields.
Workaround – To refresh these fields, navigate away from and return to the **Payment** screen.

Comparing policy transactions does not work correctly with out-of-sequence transactions (PC-31190)

Issue – When comparing two policy transactions from the Policy Transactions screen, any changes included in the policy revisions being compared that had been made effective on a later date than the edit effective date of the associated transaction (either within that transaction or in a previous transaction) are not displayed in the comparison. Changes made on earlier dates are included.
Workaround – Guidewire is aware of this issue.

Reimplement NumDaysInCoverageRatedTerm getter in Personal Auto (PC-31987)

Issue – In the Personal Auto line of the base configuration, do not use the implementation of the `NumDaysInCoverageRatedTerm` property getter in production. Implementations of this getter are in `PARatingEngine.gs` and `PASysTableRatingEngine.gs`. The getter has code comments starting with:

```
// THIS IS A TEST IMPLEMENTATION. DO NOT COPY IT FOR PRODUCTION PURPOSES.
...
override protected property get NumDaysInCoverageRatedTerm()
```

The implementation returns inconsistent results for any term number that is not a multiple of one year.

Workaround – To avoid this problem, you must change the implementation of the `NumDaysInCoverageRatedTerm` property getter. A recommended approach for rating 6-month terms is to capture the term length as part of beginning-of-term processing.

For example:

- Add a `NumDaysInCoverageRatedTerm` property to the `PolicyPeriod` entity
- Change the rating engine to use the `NumDaysInCoverageRatedTerm` property value instead of computing it:

```
override protected property get NumDaysInCoverageRatedTerm() : int {
  return Branch.NumDaysInCoverageRatedTerm
}
```

- Modify the `NewTermProcess.runPreQuote` method to set the `NumDaysInCoverageRatedTerm` property:

```
override protected function runPreQuote() {
  setRateAsOfDate()
  if (_branch.Policy.Product.DefaultTermType != TC_ANNUAL) {
    _branch.NumDaysInCoverageRatedTerm =
        Plugins.get(IPolicyTermPlugin).calculatePeriodEnd(Branch.StartOfRatedTerm,
                  Branch.Policy.Product.DefaultTermType, Branch)
  }
}
```

Merged rate books can lose references to rate tables (PC-32176)

Issue – Merged rate books created can lose references to included rate tables if one of the source rate books is in, or is moved to, draft status. A rate book in draft status can be edited and rate tables deleted, or the rate book itself can be deleted. If these types of changes are made to a source rate book, the merged rate book will have broken references to rate tables.

> **Note:** The *source rate book* is the rate book that owns the rate table. The source rate book is not necessarily one of the rate books that was merged. For example, rate book A owns rate table X. Rate book B includes rate table X by reference. Rate books B and C are merged to create D. Rate book A is the source rate book for rate table X.

When a rate book is in active status, the rate book and its contents cannot be changed. However, there is an issue with merged rate books where the contents of a referenced rate table can be modified. If the source rate book for the rate table is in draft status, the rate table contents can be changed.

Workaround – When merging rate books either:

1. Make sure the supporting rate books remain intact in every system that the merged rate book exists. That is, do not drop a supporting rate book to draft status and then delete a component referenced by the merged rate book, or change rate table content. Also migrate the supporting rate books if they do not exist in the target system.
2. Make the rate book self-contained. Do this when you want to create a single rate book that is all-encompassing. You can also do this to consolidate many development versions. Do not do this regularly as this will duplicate the space for all rate tables even if they have not changed.

Personal Auto coverages are not synchronized correctly (PC-32284)

Issue – In the base configuration, personal auto coverages are not synchronized correctly in the Personal Auto policy line. Initial Personal Auto coverages on vehicles are synchronized only when a user goes to the **PA Coverages** screen. At that point, the coverages get added as of the date that the slice selector is on. This means that the coverages could be added on an effective date later than the vehicle was added to the policy. At a later time, the coverages are added again on the effective date of the vehicle. These coverages are propagated forward to other slices, overwriting changes made to coverages or coverage term options when they were initially added.

Workaround – Change the product model synchronization logic for Personal Auto so that the initial vehicle coverages are added at the time the vehicle is added.

Upgrade to PolicyCenter after running ETL Product Model Loader extension pack (DOC-10285)

Issue – Upgrading to PolicyCenter 8.0.3 or later after running ETL Product Model Loader extension pack may require a manual drop of some tables.

In PolicyCenter 8.0.3, the ETL Product Model Loader Extension Pack functionality moved into the core PolicyCenter product. The extension pack created ETL database tables with `pcx_` as the prefix. PolicyCenter 8.0.3 and later create ETL database tables with `pc_` as the prefix. If you ran this extension pack with a prior version of PolicyCenter, you will have stale tables after upgrading to PolicyCenter 8.0.3 or later. The upgrade does not fail, but the server detects extra tables in the database.

Workaround – To remove these stale tables, manually drop the old `pcx_` tables created by the ETL Product Model Loader extension pack. The ETL Product Model Loader Extension Pack created these tables:

- `pcx_etlclausepattern`
- `pcx_etlcovtermpattern`
- `pcx_etlcovtermoption`
- `pcx_etlcovtermpackage`
- `pcx_etlpackterm`
- `pcx_etlmodifierpattern`
- `pcx_etlratefactorpattern`

Product Designer Known Issues

Cannot use Product Designer to edit an extended object with no base object (PC-27763)

Issue – If you attempt to edit a part of the product model that is based on an entity extension file (`.etx`), and that extension has no base entity file (`.eti`), Product Designer displays errors and refuses to display the contents of the file.

Although Guidewire best practices recommend that you never create an entity file extension (`.etx` file) when no base entity (`.eti` file) exists, Guidewire Studio did not enforce this recommendation. However, the current version of Product Designer does not allow the use of an entity extension without a corresponding entity base file. If such an entity extension exists, Product Designer displays a **NullPointerException** when you attempt to access the affected objects. You then can use the Back command in the browser to view other pages, but you cannot configure the affected objects.

Workaround – Locate all `.etx` files that do not have corresponding `.eti` files, and rename them with an `.eti` extension.

PolicyCenter does not start in single currency display mode if the product model is configured to use multiple currencies (PC-27829)

Issue – Product Designer allows users to configure multicurrency features without regard for the setting of the `MultiCurrencyDisplayMode` in the PolicyCenter `config.xml` file. If you configure multiple currencies in a product model object, and the value of the `MultiCurrencyDisplayMode` property is `SINGLE`, PolicyCenter does not start.

For example, if you add clauses that specify multiple coverage currencies and the PolicyCenter `MultiCurrencyDisplayMode` is set to `SINGLE`, the server fails to start and displays messages similar to the following:

```
ERROR ... PolicyLinePattern : ... [field : AvailableCoverageCurrencies] : - Multiple Coverage
Currencies defined for product not allowed in single Currency mode
ERROR ... [ExclusionPattern] : ... [DirectCovTermPattern] : ... [LimitsSet] : - The pattern specifies
limits for different currencies in SingleCurrencyMode
```

Workaround – If you intend to deploy a product model on PolicyCenter servers that are not configured for multiple currency display mode, do not configure multiple currencies in any part of the product model. However, you can test multicurrency products on a staging server configured for multicurrency while the production system is configured for a single currency.

System table cells display public IDs when referencing another system table (PC-28368)

Issue – If the cells in a particular column of a system table are referencing values from another system table, only the public IDs of those values are shown, not the values, when the system table is edited in Product Designer. In addition, users cannot search for values in these system table cells. The referenced system table, however, does not display the public ids so they cannot be viewed from within Product Designer.

Workaround – View the XML directly for the system table being referenced to determine the public IDs associated with the values that need to be added, edited or deleted. You can then use those public IDs directly.

An exception might appear when you run Product Designer (PC-28531)

Issue – An `IncompatibleRemoteServiceException` might appear if you run Product Designer 8.0.3 or later after running a previous version of Product Designer.

Workaround – Refresh the browser cache by selecting `Ctrl+F5`.

Invalid audit schedules when audit schedule properties file is changed while user is editing audit schedule (PC-28146)

Issue – If you edit an audit schedule in Product Designer at the time that the associated audit schedule properties file is updated, you may not see the properties file changes reflected in Product Designer. This could result in an invalid audit schedule (that matches the previous audit schedule properties file).

Workaround – Perform the following steps:

- In Product Designer, revert the changes to affected audit schedules.
- To load the audit schedule properties file changes into Product Designer, restart Product Designer, logout, or change the changelist.
- Reopen the affected audit schedules.

Studio/Platform Known Issues

Renaming method or property throws ParseResultsException (PL-16633)

Issue – If you rename a property or a method or you change a method signature, and a workflow references that property or method in a Gosu field, PolicyCenter throws `ParseResultsException`. This is the intended behavior.

Workaround – Restart the workflow engine. To do so:

1. Log in to PolicyCenter using an administrative account.
2. Access **Internal Tools→Reload**.
3. Click **Reload Workflow Engine**.

Long text in table cells can add white space to the right of the page (PL-28288)

Issue – In an editable list view, extremely long text entered in a single cell can cause additional white space on the right side of the page. Long text is text that occupies approximately the entire width of the screen. This issue occurs primarily in Chrome.

Workaround – If you expect users to enter large amounts of text into the cells of a column, configure the column to support text wrapping.

Administrative command-line tools cannot refresh WSDL (PL-29021)

Issue – Administrative command-line tools rely on web service implementation classes such as `MaintenanceToolsAPI.gs`. If you change the web service implementation classes, administrative tools might fail because the tool's WSDL does not match the server WSDL. Some changes do not affect the WSDL. For example, adding a `@WsiPermission annotation`.

Workaround – Do the following:

From the `PolicyCenter/bin` directory, at a command prompt type the command:

```
gwpc regen-soap-api
```

In Windows Explorer, copy the WSDL files from the location:

```
PolicyCenter/soap-api/wsi/wsdl
```

to:

```
PolicyCenter/admin/res/wsi
```

No entity changed event fired through a one-to-one entity relationship (PL-26224)

Issue – The existence of a one-to-one entity relationship between two entities prevents an entity change event from being fired, even if it is supposed to. For example, in ContactManager there is a foreign key from `ABContact` to `Address` for the primary address, and a one-to-one relationship from `Address` to `ABContact`. Because of the one-to-one link, no event changed event is fired for any change in the primary address.

More generally, if entity E1 has a one-to-one relationship to entity E2, no entity changed event is fired for E2 if a property on E1 changes.

Workaround – You might be able to take advantage of a secondary change caused by the initial change, with the secondary change triggering the event to be fired instead. In the previous ContactManager example, making sure that changes to all properties in `Address` cause a `History` record to be generated ensures that the entity changed event will be fired.

Free-text Search indexes cannot be upgraded from 7.0 to 8.0 (PL-29062)

Issue – To upgrade free-text search in PolicyCenter from 7.0 to 8.0, you install the newer version of the Guidewire Solr Extension in 8.0 on top of the version from 7.0. Because the PolicyCenter data model also changes, the free-text search indexes from 7.0 become obsolete. Guidewire does not support migrating free-text search indexes from 7.0. to 8.0.

Workaround – For Oracle, upgrade the Guidewire Solr Extension by following instructions in the *Upgrade Guide*. Then, run the new free-text batch load command to rebuild the indexes based on the new data model, after you complete your database upgrade. For SQL Server, Guidewire is aware of potential performance problems running the free-text batch load command after you complete your database upgrade.

Chrome browser cannot display product documentation in HTML format (DOC-7251)

Issue – If you use the Google Chrome browser, you can view the HTML Guidewire product documentation only if it is served by an HTTP server using the `http://` protocol. The Chrome browser cannot load HTML product documentation from your local disk by using the `file://` protocol.

Workaround – Use a different browser, such as Microsoft Internet Explorer. If you use the HTML documentation only to access the PDF files, you can open them directly in the `pdf` subdirectory of the `doc` directory.

Alignment Property Does Not Work for Inputs in a DetailView (PL-29429)

Issue – Inputs in a detail view do not align to the right or center. They always align to the left, regardless of the value of the `align` property.

Workaround – If you need to right align a series of items such as a set of monetary values, put them into a list view instead of a detail view. You can then set the alignment as needed.

JMS inbound integration implementations must be OSGi or Java plugins (PL-32054)

Issue – Writing JMS inbound integration handlers requires casting an object to the type `javax.jms.Message`. In Gosu, for some app servers this fails with the message: "Caused by: gw.lang.parser.exceptions.ErrantGosuClassException: GosuClass test.GosuJMSMessageReplyHandler has errors, and cannot be used at runtime."

Workaround – Implement JMS handler code for inbound integration in Java. Guidewire recommends that for inbound integration Java code, implement your code as an OSGi plugin, not a standard (non-OSGi) Java plugin.

Document Assistant incompatible with Chrome browser version 45 and later (PL-33648)

Issue – The Guidewire Document Assistant is incompatible with Chrome browser version 45 and later. Chrome browser versions 45 and later no longer support the Netscape Plugin Application Programming Interface (NPAPI).

Workaround – Use the browsers Firefox or Internet Explorer.

Guidewire PolicyCenter 8.0.4 Release Notes

Release 8.0.4

Release Notes Update

IMPORTANT These release notes replace the release notes that were included with the official product release. Please disregard the earlier version of the release notes.

Overview

These release notes contain the following:

- "Release Information" on page 351
- "Installing This Release" on page 352
- "Support" on page 352
- "Major Issues and Changes" on page 353
- "Improvements and Resolved Issues" on page 353
- "Known Issues and Limitations" on page 360

Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

IMPORTANT If you skipped one or more upgrade releases to PolicyCenter, be sure to read the release notes for those releases to learn about changes and fixed issues.

Version Number

This release of Guidewire PolicyCenter is 8.0.4.

- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 7.0.1 or later. The latest maintenance release is preferred.

If you are implementing multiple settlement currencies, BillingCenter 8.0.2 or later is preferred.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 7.0.0 or later. The latest maintenance release is preferred.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 7.0.7, 8.0.1, or later. The latest maintenance release is preferred.

Installing This Release

Refer to the following:
- The *PolicyCenter Installation Guide* for general installation information
- Prior PolicyCenter release notes for any versions that you have skipped

Upgrade Information

Incorrect warning message when running Configuration Upgrade Tool (PL-28723)

When upgrading from one PolicyCenter 8 release to another PolicyCenter 8 release, running the Configuration Upgrade Tool may produce the following warning message:

```
WARN cannot find Emerald base configuration zip, this could indicate a problem...
```

If the `PolicyCenter/modules/base.zip` file does exist, then you can safely ignore this warning.

Upgrade Product Model Structure (PC-26136, PC-25834)

In PolicyCenter 8.0.4, the structure of some product model files changed. Added new `CodeIdentifier` attribute with the same value as the `PublicID` to all product model patterns. The `CodeIdentifier` is set to the same value as the `PublicID` during configuration upgrade. The Configuration Upgrade Tool updates the product model structure for all lines of business in your PolicyCenter installation.

For audit schedule patterns, the automatic upgrade will change the type of the pattern from `AuditSchedulePattern` (with series flag `false`) or `SingleCheckingAuditSchedule` to `SingleAuditSchedulePattern`. Also, the `series` attribute is removed from all `AuditSchedulePattern` instances. There may be cases in which manual intervention is required. The Configuration Upgrade Tool will log a warning to the console indicating which patterns may require additional configuration.

To explicitly upgrade a 7.0.x or 8.0.x product model structure to the current 8.0.x structure, use the `upgrade-productmodel-structure` target of `gwpc`. This replaces the `upgrade-productmodel-to-803` target. See the *Installation Guide* for more information. The documentation incorrectly refers to the new target as `upgrade-productmodel-to-structure`.

Version trigger removes parameter from argument source in sample data (PC-30023)

In PolicyCenter 8.0.4, a version trigger removes one parameter from the argument source in the `AutomaticRenewalCap` rate table definition. This rate table definition is in the sample data.

In PolicyCenter 8.0.3, in `AutomaticRenewalCap` rate table definition, click **Edit**, then **Update**. Do this before performing the upgrade to PolicyCenter 8.0.4.

Support

For assistance, visit the Guidewire Community.

Guidewire Customers

- `https://community.guidewire.com`

Guidewire Partners

- https://partner.guidewire.com

Major Issues and Changes

This topic describes major issues and changes that can affect your installation.

New and Changed Features

For information on new features and other changes, see the topic "New and Changed in PolicyCenter 8.0.4" in the *New and Changed Guide*.

Base PCF File Changes

The link below requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 8.0.3 to 8.0.4

To view a report of the changes to the base PCF files, `click here`.

Base Resource Changes

The link below requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 8.0.3 to 8.0.4

To view a report of the changes to the base resources, `click here`.

Changes in This Release Provided in Upgrade Diff Report

Guidewire provides a report detailing certain differences between the current release and your prior release. This report describes changes in display keys, entities, typelists, and the Gosu API. To obtain your custom Upgrade Diff Report, visit the Guidewire Community.

Improvements and Resolved Issues

This topic describes improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This list is not intended to be comprehensive.

- "PolicyCenter improvements and resolved issues" on page 353
- "Product Designer improvements and resolved issues" on page 357
- "Platform improvements and resolved issues" on page 357

PolicyCenter improvements and resolved issues

The following are the primary improvements and issues corrected in this release:

ID	Description
Account	
PC-21925	With this fix, you cannot delete a producer code if that code is used as a producer of record on any policy period or producer of service on any policy.

ID	Description
PC-23489	Fixed the **Related To** drop-down list on the **Account File Documents** search screen to display the full descriptive text instead of just the identifier.
PC-24811	On the **New Account** screen, the producer code list is now refreshed after changing the producer organization.
PC-26201	Account locations are automatically assigned an identifying location number when created. Under certain circumstances these numbers can be duplicated which is not desirable. Now when an account's locations are updated, duplicate location numbers are renumbered with an effort to preserve the location numbers of the current account locations.
Archiving/Purging	
PC-26996 PL-32941	With this change, the `Purger` class now purges temporary policy period branches. The `PCDependencies` and `PCDependenciesGateway` classes give you access to a purger (`getPurger`). If you purge policy periods using the purger, temporary policy period branches will now be purged.
Contacts	
PC-25520	Fixed an issue with removing additional interest. Contact is now removed from policy contacts.
PC-25549	Fixed an issue where `PolicyAdditionalInterest` objects were not deleted when the last associated `AddlInterestDetail` was deleted.
PC-25832	Fixed an issue where the Secondary Named Insured on a policy (`SecondaryNamedInsured`) triggered Database Consistency Check errors if the `SecondaryNamedInsured` was committed to the database and then removed.
PC-26599	Fixed an issue on the **Search→Contact** screen which resulted in error message **Cannot convert value "ABUserContact"**. PolicyCenter now excludes user contacts created in ContactManager.
Data Model / Core Entities	
PC-25592	Fixed an issue where copying a policy period resulted in a database consistency check error.
Financials	
PC-26107	Fixed an issue in the `CostEnhancement` class where `SubjectToAccrual` for flat rated costs returned `true` rather than `false`. (Flat rate premium is earned when charged and is not accrued.) This could have resulted in an error in the earned premium which is displayed on the policy file **Summary** screen.
General	
PC-25086	The `PolicyImpl.startConversionRenewalWithBasedOn` method now logs exceptions to the `Application.JobProcess` logging category.
PC-25844	Fixed `TransactionUtil.runAtomically` to prevent bundle commits until the block has completed. This change to `runAtomically` enables the Job code to run more atomically. In configuration, ensure that changes to the policy, its status and message creation are part of a single atomic action.
PC-25854	When an error occurs during the creation of a new branch from an existing one (copy submission, for example) an improved error message is displayed. Log entries with debugging information are also created.
PC-26108	Added new `config.xml` parameter `WorkQueueBulkInsertIdAllowance` that creates an allowance, or buffer, for IDs as part of the bulk insert process in some work queues, such as the Policy Exception work queues. This parameter sets the limit for the number of IDs that can be inserted into query results before the insert select compares the row count. The default value is 10.
PC-26230	Changed "Policy Center" to "PolicyCenter" in `BillingSystemPlugin.Error.MismatchedPaymentMethod` display key.
PC-26492	Added validation for null state on **New Submission** screen.
PC-30025	Fixed a `RangeCell` issue that resulted in HTML characters that were not properly escaped.
PC-30026	Resolved an issue that caused `EffDatedBaseProxy#changeNonDefaultLocaleValue` to create duplicate slices.
Internationalization	

ID	Description
PC-26327	Fixed the `AbstractUnderwriterEvaluator#maybeAddUWIssueForProducer` so that underwriting issue short and long description display keys resolve to the correct locales.
Integration	
PC-25436	Added a new version of the `GWClaimSearchPlugin` in the `gw.plugin.claimsearch.cc706` package. This plugin uses the ClaimCenter 7.0.6 version of the Claim Search API. The ClaimCenter 7.0.6 API provides multicurrency support when integrating between ClaimCenter 7.0.6 and later and PolicyCenter 8.0.4 and later.
PC-25437	Fixed an issue in the integration between PolicyCenter and ContactManager which caused exceptions when ContactManager merged duplicate PolicyCenter account contacts.
Jobs	
PC-22823	Fixed an issue where clicking on secondary named insured in a submission caused a null pointer exception.
PC-26063	Fixed an issue where policy term numbers were reset to zero during a legacy conversion. With this fix, conversion on renewals increment from the term number provided from the legacy system. Note: The BillingCenter limitation on the maximum term number is 63. This issue is logged as BC-15168.
PC-26271	Fixed an issue which resulted in an `IllegalArgumentException` with the message "Cannot promote preempted branch" when trying to issue a policy change.
PC-29652	Fixed an issue which allowed a user to bind a policy ignoring changes in a prior policy transaction by another user. Transaction arrays are now owned by the `PolicyPeriod`, causing the `PolicyPeriod` to be marked dirty whenever one of the arrays is modified.
Line of Business - Businessowners, Commercial Property	
PC-24789	Implemented `EffDatedLogicalMatcher` for `BuildingImprovement` and `BuildingSide` to help protect against the creation of duplicate rows while executing out-of-sequence jobs. For `BuildingImprovement`, matching is performed against the improvement type, such as Heating, and the parent building. For `BuildingSide`, matching is performed against the side type, such as Front, and the parent building. Although there is a metadata change, no tables were modified so there is no upgrade impact.
Line of Business - Commercial Auto, Personal Auto	
PC-25782	In the base configuration, the Short Rate calculation of the penalty in Commercial Auto and Personal Auto was calculated incorrectly. Previously, the calculation summed the premium for the uncanceled portion of the policy (effectively the earned premium), and then calculated the penalty as a percentage of that amount. The penalty is now calculated as a percentage of the unearned premium. Short rate is now calculated using this formula: sum the premium to date, sum the premium for the full term. The penalty is the short rate factor multiplied by (premium full term minus premium to date).
Line of Business - Commercial Property	
PC-25771	In Commercial Property, fixed an issue on the **Buildings and Locations** step that prevented exporting from a bound submission or renewal policy transaction.
Line of Business - Personal Auto	
PC-25804	Disabling the demonstration `IMotorVehicleRecordPlugin` will now correctly disable the MVR checks during Personal Auto policy issuance. This was previously a known issue.
Line of Business - Workers' Compensation	
PC-23965	Fixed an issue in Workers' Compensation where rows in **Covered Employees** with the same class code were removed during an out-of-sequence job.
Performance Improvement	
PC-26208	In systems using producer code security with large numbers of producer codes, improved performance of policy searches by first name and last name.
PC-26313	Improved PolicyCenter user interface performance in configurations with large numbers of users assigned to large numbers of producer codes.
PC-26324	Improved performance of permission checks for users with producer code security.

ID	Description
Product Model	
PC-23397	Fixed an issue where the `DirectCovTermPattern` methods `getMinVal` and `getMaxVal` caused exceptions in single currency mode.
PC-25566	The `ProductModel.getClausePatternsForPolicyLinePattern` method now returns the correct result (all clause patterns for the given policy line).
PC-25861	In PolicyCenter 8.0.3, the `ClausePattern.GRANDFATHER_STATES` property was removed. The new `GrandfatherStates` property on all `ClausePattern` instances provides access to grandfathering information. For example, accessing the `GrandfatherStates` property on a coverage pattern will return an equivalent result to `ClausePattern.GRANDFATHER_STATES.get(PACollisionCov)`. The same applies to `Offerings`, which now has the `GrandfatherStates` property on an offering instance instead of the static `Offerings.GRANDFATHER_STATES` getter.
PC-26511	Fixed an issue where an array of jurisdictions in state lookups caused an exception. Though this fix is most likely to apply to Workers' Compensation, it applies to any line of business that has line level clauses whose availability is potentially based on any jurisdiction on the policy, not just on the base state of the policy.
PC-26886	Fixed an issue where PolicyCenter was not explicitly validating the `publicID` maximum length of 64 characters for all product model pattern objects.
Rating	
PC-20627	Fixed locking issue that occurred with Impact Testing on a SQLServer database. This issue caused failures in the **Create Baseline** step, and `DEADLOCK` messages in the PolicyCenter log file.
PC-22803	In rating worksheets, currency conversion is now appended to the worksheets when the rating currency differs from the settlement currency.
PC-22851	New navigation allows you to navigate back from a rate routine to the search or rate book screen you came from. This uses a button rather than a link.
PC-25056	Fixed an issue where a rate table whose `QueryStrategy` had been edited could not be removed from a rate book.
PC-25326	In conditional blocks of rate routines, fixed an issue which trimmed whitespace added for code indentation. This whitespace is now preserved.
PC-25686	Fixed an issue which caused a `DBNullConstraintException` when editing a rate table definition and setting the parameter set to **<none>**.
PC-26266	A numeric constant entered directly into the Rate Routine editor is now correctly categorized as decimal or integer.
PC-26814	Fixed sample code in `CPSysTableRatingEngine.gs` to process premium overrides correctly.
PC-26815	On the **Rate Routine Details** screen, replaced the **Go Back** button with a link.
PC-26961	On the **Parameter Sets** screen, the **Wrapper/Coverage** field now displays the name of the selected coverage or wrapper. Formerly, it displayed the public ID of a coverage or the type of a wrapper.
Reinsurance	
PC-15905	With this change, the list of reinsurance agreements in the `RIRiskInfo` object will not contain agreements in draft status. In the `RICoverageAPI` web service, the `findRIRiskByCoverableID` and `findRIPolicyRisk` methods return the `RIRiskInfo` object. This change affects both the `700` and `800` versions of the web service.
PC-29700	Fixed an issue where ceded premium transactions were missing on cancellations when there was a policy change effective after the cancellation date.
Revisioning	
PC-29636	Fixed an issue where a query against an `EffDated` entity incorrectly included instances in temporary branches. To include `EffDated` entities in temporary branches, call `withFindTemporaryBranches(true)` on the query object.
UI Configuration	

ID	Description
PC-27001	The **Valid until** column on the **Risk Approval Details** history screen now includes the **Invalid From** type description if no expiration date is set. If you set the expiration date, then PolicyCenter will display it.
Underwriting	
PC-22766	Changed the sample data so that the Underwriter 1 and Underwriter 2 authority profiles can approve reinsurance issues.
PC-25830	Fixed an issue which caused PolicyCenter to throw a `Not expecting blocking point` exception when an underwriting issue was rejected.

Product Designer improvements and resolved issues

The following are the primary improvements and issues corrected in this release:

ID	Description
Availability	
PC-28483	This change ensures that pending changes are not lost when the Product Designer server is restarted.
Offerings	
PC-28021	To prevent data loss, Product Designer no longer allows simultaneous edits to the same pattern from two different screens. To make edits from the second screen, commit your changes before making additional edits.
PC-28285	Fixed an issue where deleting a product question set did not remove all of the associated information from the product model XML. Specifically, the question set selection entry associated with the product question set was not removed.
Product Model (Overall)	
PC-28274	In a multiple user scenario, **Revert** is now consistent with **Revert All** and **Validate**. Like **Revert All** and **Validate**, **Revert** now reflects the changes done by a different user in another session.
System Tables	
PC-28860	Fixed an issue where edits made to the `underwriting_Companies.xml` system file reordered the entity type nodes, thereby causing server start-up failure.

Platform improvements and resolved issues

The following are the primary improvements and issues corrected in this release:

ID	Description
Appserver Support - WebSphere	
PL-31805	EAR built for WebSphere now contains an IBM-specific descriptor that sets `classloading` mode to `parent-last` and WAR `classloader` policy to `single`. Setting these parameters in WebSphere manually was required for prior releases. Verify that `classloading` mode and WAR `classloader` policy are set correctly after the installation.
Configuration Upgrade	
PL-32082	The configuration upgrade tool would report fatal errors with case issues in display keys. The tool now corrects these issues and proceeds with the configuration upgrade.
Database Instrumentation	
PL-25165	Guidewire has modified the Database Table Info download to include the XML database configuration files. To access the files, download the table information file from the **Server Tools→Database Table Info** screen, then open the `index.html` file, and then click **config_files: Directory with config files**.

ID	Description
PL-31967	Added two new options to the `table_import` command to list and download load history reports.
Database Support - Oracle	
PL-31514	LOB objects associated with spatial columns are now stored in a LOB tablespace, if so configured for the table.
Database Support - SQL Server	
PL-31729	LOB objects associated with spatial columns are now stored in a LOB tablespace, if so configured for the table.
PL-32898 PC-30119	Guidewire has improved the performance of the production of the (Server Tools Info Pages) SQL Server DMV Snapshot by restricting the data extract to the top 400 queries.
Database Upgrade	
PL-32115	A SQL Server upgrade that runs only the version trigger that converts integers to big integers for ID and foreign keys now runs without error.
Document Management	
PL-32605	Resolved an issue that caused a `ClassCastException` exception in document generation if you set `IDocumentTemplateSource.gwp` plugin parameter `cacheDescriptors` to `true`.
Gosu	
PL-30881	Resolved an issue with GosuProgram threads retaining type system locks unnecessarily, which resulted in deadlock.
PL-32490	Fixed an issue where Gosu does not automatically downcast if the left side of the `typeis` or `typeof` expression uses deprecated members.
IntelliJ IDE - Typelist Editor	
PL-30524	Fixed an issue where Studio would encode files containing non-English characters as ANSI.
Messaging	
PL-31680	Resolved a thread safety issue that occurred in rare cases when initializing event messages.
Other	
PLWEB-161	Fixed an issue where printing list views would throw an exception.
PLWEB-1997	Fixed an issue where an ampersand or other special character would break auto-filling other data.
PLWEB-2032	Added a `width` attribute to `ToolbarFilter` to specify a width in pixels. This allows you to increase the width of a `ToolbarFilter` if the default width does not allow all filter options to be displayed
PLWEB-4568	Restored the ability to hide the sidebar.
PLWEB-5	Fixed an issue where the progress bar could be triggered multiple times and cause an incorrect status to be shown.
Other - Cloud	
PL-27758	CSV export from a list view now correctly handles newline characters (CR, LR) in the source data.
PL-32043	Upgraded the Joda-Time library to version 2.5 for updated time zone information.
Other - Integration Pod	
PL-31798	The utility class instance for the production data change utility methods changed packages. If you use the access syntax that the documentation recommends (`DataChange.util`), your code requires no changes.
PCF	
PLWEB-2017	In a `RowTree`, a `Row` widget that has its `highlighted` property set to `true` now displays as highlighted.
PLWEB-4431 PC-30024	Fixed an issue where incorrect text was displayed in the QuickJump box while typing a command.

ID	Description
Staging Tables	
PL-30911	(Oracle) Guidewire has added a `<loader>` element to `database-config.xml` that you can use to configure parallel operations during staging table load.
PL-31931	If an integrity check and load fails after the check, another load operation on the same database is not performed.
Web - Configuration	
PLWEB-2000	Fixed an issue where the regional calendar for Portuguese displayed extraneous characters.
PLWEB-2035	Resolved an issue where the combination date-time widget would not auto-fill with a standard time when only the date was entered, if the date contained leading zeroes in the month or day.
PLWEB-4251	Added a `visible` attribute to the `TabBarUnsavedWork` and the `MenuActions` button to allow dynamic enabling and disabling of their visibility based on Gosu logic.
Web - IE Support	
PLWEB-18	As a security precaution, added an explicit header to prevent MIME sniffing.
Web - Styling	
PLWEB-1993	Fixed an issue where dividers were not appearing in the **Actions** menu.
PLWEB-4289	Fixed an issue where the `labelStyleClass` was not being applied properly to inputs in the `GlobalAddressInputSet`.
Web - UI/Runtime	
PLWEB-140	Fixed an alignment issue with check boxes in input groups.
PLWEB-15	Fixed an issue where dates with leading zeroes in the month and year would not autoformat correctly for all localization options.
PLWEB-1728	Improved keyboard navigation of the **Actions** menu.
PLWEB-1918	Fixed an issue where sorting a tree table would throw an exception.
PLWEB-2003	Fixed an issue where an unsecured GET request could be exposed.
PLWEB-2005	Fixed an issue where pressing `Enter` on a check box would both toggle the check box and submit the page.
PLWEB-2010	Fixed an issue where `numCols` did not appropriately scale for the font size. For example, year fields specified with `numCols = 4` did not show all four digits. This may affect the layout of screens where `numCols` has previously been set.
PLWEB-2034	Fixed an issue with the `target` property of the `ExitPoint` widget that prevented it from functioning properly.
PLWEB-2039	Pressing `Enter` no longer changes the value of a check box. Now only pressing `Space` changes it.
PLWEB-2040	Fixed an issue where a button with the `download` attribute set to `true` lost the chosen values from the multi-select shuttle widget.
PLWEB-2047	Fixed an issue where help text would not appear for boolean radio button inputs.
PLWEB-2079	Fixed the info bar to allow for better copying and pasting of individual text snippets.
PLWEB-2191	Modified info bar styling to better reflect which items are links instead of text.
PLWEB-3042	Modified styling of input groups to better delineate expansion and separation.
PLWEB-32	Fixed an issue where inputs would not reset to the default value after enabling and disabling.
PLWEB-37	A date and time entered without separators, such as "11272014 1035am", is now accurately translated and validated. Note that a space must separate the date from the time.
PLWEB-4199	Fixed an issue where the `objFocusID` was not escaping HTML properly.
PLWEB-4292	Fixed a `RangeCell` issue that resulted in HTML characters that were not properly escaped.

ID	Description
PC-30025	
PLWEB-4521	Fixed an issue where the HTML in a `RangeCell` was not escaping properly.
Web Services - WSI (New)	
PL-32548	Fixed a security vulnerability to XML external entity (XXE) attacks. For more information about this issue, or to learn about temporary workarounds while you deploy this fix, visit the Guidewire Community and search for Knowledge article "*Are Guidewire PolicyCenter, BillingCenter, ClaimCenter, and ContactManager Vulnerable to XML External Entity (XXE) Attacks in Their Web Service Layers?*".

Known Issues and Limitations

This section describes known issues with this release of Guidewire PolicyCenter.

- "PolicyCenter Known Issues" on page 360
- "Product Designer Known Issues" on page 364
- "Studio/Platform Known Issues" on page 365

  Note: For maintenance releases, Guidewire often defers fixing configuration issues that require merging files during the upgrade. Workarounds to many of these issues are listed in the following topics. The goal of this policy is to make upgrades as straightforward as possible.

PolicyCenter Known Issues

Provide a way to filter out added and removed subentities in a policy differences tree (PC-7929)

Issue – Need a way to filter out the children of an added or removed entity in a policy differences tree.
Workaround – Guidewire is aware of this issue.

When using parallel sessions, approvals made by underwriter are not updated for second user simultaneously viewing same work order (PC-12322)

Issue – Approvals made by an underwriter are not updated for a second user simultaneously viewing the same work order. This occurs in the following sequence of events:

- An agent processes a submission and issues arise
- The agent calls an underwriter, and remains in the policy
- The underwriter approves the issue or issues
- The agent refreshes the **Risk Analysis** screen by going to a different wizard step and returning to the **Risk Analysis** screen

After this sequence of events:

- The new approvals do not appear on the agent's screen
- However, the agent can bind or issue the policy (based on the new approvals by the underwriter)

Workaround – The agent can leave the policy and return to it. The approvals appear as expected. Alternatively, the agent can see the approvals by clicking the **Details** button when issuing the policy.

Must enter VIN when moving from quick quote multi-version or side-by-side to full application (PC-13099)

Issue – In a quick quote, you do not have to enter the VIN or other fields that are required for a full application submission. When converting a quick quote with multi-version or side-by-side quoting to full application, to resolve validation issues, you must enter the VIN for each of the versions. In a job with side-by-side quoting, this issue can occur with fields that are not base data.

Workaround – Add code that copies this type of data to the other versions.

PolicyCenter does not send renewal billing instruction to BillingCenter (PC-15082)

Issue – On receiving payment notification from BillingCenter, PolicyCenter tries to bind. If the policy is in a draft status, PolicyCenter does not bind and send the Renewal Billing Instruction to BillingCenter. The policy might be in draft status after an underwriter edited something on the policy. This is an issue with BillingCenter 3.0.1 and later.

Workaround – Guidewire is aware of this issue.

BillingCenter integration: Automatic payment instrument is used to create invoice stream even if final selection is manual (PC-15307)

Issue – If an automatic payment instrument is setup and selected, the policy will create a new invoice stream with this payment instrument even if the user changes to Manual before issuing the policy.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Guidewire is aware of this issue.

BillingCenter integration: Removing alternate billing account when payment instrument selected generates WsdlFault (PC-15308)

Issue – If an automatic payment instrument is selected when using an Alt Billing Account to create a new policy, PolicyCenter throws a `WsdlFault` if the Alt Billing Account gets removed.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Guidewire is aware of this issue.

BillingCenter integration: New invoice streams not created when issuing policy with alternate billing account on new account (PC-15309)

Issue – Assume an account has an existing invoice stream. When this account is used as an **Alt Billing Account** for new policies on new accounts, a new invoice stream is not created when the new policy's new invoice stream periodicity matches the invoice stream on the **Alt Billing Account**.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Guidewire is aware of this issue.

Renewal processing for renewal conversion assumes BasedOn is populated (PC-15851)

Issue – The `startNewRenewal` method in `gw.webservice.pc.pc700.job.PolicyRenwalAPI` assumes the existence of a prior term or a `basedOn` version. Unless customized, this method can cause problems in certain convert-on-renewal scenarios. This method has been deprecated.

Workaround – If you are developing convert-on-renewal flows, use the `gw.webservice.pc.pc700.job.startConversionRenewal` method.

SQL Query performance in free-text search (PC-18794)

Issue – There are open issues related to the performance of SQL query in free-text search. This occurs the first time that you load the PolicyCenter data into the Solr index database.

Workaround – Guidewire is aware of this issue.

Cannot quote a renewal if the expiring policy term has a handled preemption (PC-19416)

Issue – In Commercial Package Policy in the base configuration, if a policy term has a preemption that has been handled, when a renewal job is created, the **Edit Workorder** button is not always available on the renewal. When this occurs, the only options are to bind or not take the renewal.

Workaround – Guidewire is aware of this issue.

Copy-data with an out-of-range date (PC-20279)

Issue – When working with the copy data feature, if you specify a specific date to copy data from and that date is out of range for the policy, PolicyCenter presents a warning that no applicable policy version could be found.

However, the date selector still indicates the requested date, not the date that PolicyCenter will use for the copy if you choose to proceed.

Workaround – As appropriate, modify the user interface to reset the date selector or to hide the ability to select data to copy.

Error when quoting after adding new location (PC-25559)

Issue – In the Commercial Auto, General Liability, and Commercial Property lines, users sometimes receive an error message after clicking **Quote** on the **Locations** screen indicating that there are no locations even when there are. This error could potentially occur in other lines of business and on other list views, depending on how they are configured.

Workaround – Click **Next** and then **Back** to navigate away from and back to the **Locations** screen before quoting. Alternately, remove the client side validation on the number of rows.

Reinsurance warnings not refreshed even after saving a draft (PC-21720)

Issue – When working with reinsurance, PolicyCenter displays validation errors and warnings both at the top and in the worksheet at the bottom of the screen depending on the action that caused validation to run (typically **Save**, or leaving the screen). Depending on the user flow, one or both of these areas may display stale messages, and the messages may not be the same.

Workaround – Refresh the messages in either or both locations before taking the action again.

Gosu classes not intended for use (PC-23009)

Issue – Some Gosu classes are visible in Studio, but are not intended for use. You can distinguish these Gosu classes because they have no visibility annotations (neither `@Export` nor `@ReadOnly`) and they are not in a `gw` package.

Workaround – Do not use these methods in configuration. The methods are unsupported and may change or be withdrawn without notice.

Billing system integration startup error when creating carrier organization (PC-23211, PC-15682)

Issue – When PolicyCenter is first started it creates a carrier organization as part of the start up process. The carrier organization by definition is also a producer. PolicyCenter sends a message to the billing system when a producer is created so that the producer can also be created in the billing system. However at the point in start up when the carrier organization is created, the messaging system has not been started. Therefore no create producer message is sent. When subsequent changes are made to the carrier organization an update message is sent to the billing system to update the carrier organization. This may result in errors being generated in the billing system.

PolicyCenter reports the following error:

```
Retryable Error at startup: Cannot find producer to update with publicid: systemTables: 1
```

In the case of PolicyCenter to BillingCenter integration BillingCenter responds to the update message from PolicyCenter with the error message:

```
BadIdentifierException: Producer with publicId systemTables:1 was not found
```

Workaround – Guidewire is aware of this issue.

PolicyCenter allows owner and overriding-payer accounts to be the same (PC-23570)

Issue – In PolicyCenter, you can set the overriding payer on a list bill policy to the same account as that account which owns the policy. You set the overriding on the **Account**→**Billing** screen. Systems which receive data from PolicyCenter may report this as an error. For example, BillingCenter requires an overriding payer account to be different than the owner account.

Workaround – Add validation or other logic to PolicyCenter that ensures that, if present, an overriding-payer account is different than the owner account.

Unbound renewal policy transactions can provide invalid dates (PC-25288)

Issue – If a policy renewal is in the renewal process and the user goes from the renewal job wizard to the policy file summary, the as-of date default in the **Date** field of the **Actions** panel will not be entirely valid, as it will be the `editEffectiveDate` of an unbound period from an incomplete job.The quote, reinsurance, and audit schedule, if enabled, screens display exceptions because appropriate periods do not exist on the specified dates.

Workaround – Edit `JobWizardInfoBar.pcf` to filter out any `PolicyPeriod` objects that cannot provide the `editEffectiveDate`.

Billing integration error when final audit is waived from Audit job (PC-25500)

Issue – When the user waives a final audit from within a running Audit job, the BillingCenter integration sends an incomplete message to BillingCenter. The message does not include the term number of the policy period. BillingCenter needs the term number to identify the BillingCenter policy period that needs the change.

Workaround – Guidewire is aware of this issue.

Insurance Suite integration issues (PC-25732, PC-25733, PC-23873)

Issue – There are issues with creating and sharing contact information between PolicyCenter, BillingCenter, and ContactManager if the applications are different versions. For example, these issues arise if you have PolicyCenter 8.0 integrated with BillingCenter 7.0. These issues include:

1. In the `BillingMessageTransport` Gosu class, the commented-out code in the `send` method is invalid. This commented code handles the case when ContactManager is unavailable and the message is ready to be sent. The commented code begins with:

```
/*
    It is not recommended to wait for ContactManager to process the contact messages ...
```

   The `BillingContactUnsyncedException` method does not exist and the code does not compile.

2. When PolicyCenter 8.0 is integrated with BillingCenter 7.0, contacts sent to BillingCenter have a null `AddressBookUID` even though PolicyCenter has called ContactManager with the suggested `UniqueID`. This results in BillingCenter also creating the contact in ContactManager, leading to duplicate contacts.

3. When BillingCenter creates a new Contact for an Account that had been created in PolicyCenter, PolicyCenter does not recognize the Contact and throws a `BadIdentifierException`.

Workaround – Guidewire is aware of these issues. In the Knowledge Base, see article 3183.

In addition:

1. Do not uncomment the code in the `BillingMessageTransport` class. Write code that handles the case when ContactManager is unavailable and the message is ready to be sent.

2. In the `bc700` package, update the sync method in the `PCContactInfoEnhancement` class to match the version in the `bc800` package.

Rate routine editor message needs notification of parameter change (PC-30084)

Issue – The rate routine editor does not display message if rate table definition parameter is modified.

A rate table definition is referenced in a rate routine. You modify a parameter in the rate table definition. When you return to the rate routine, the rate routine editor does not display a message warning that it is unable to reconcile the definition of rate table.

If you select a rate table, the rate routine editor does not warn you that parameters have been modified in the rate table. You are not given the option to accept this change or cancel.

Workaround – Guidewire is aware of this issue.

Businessowners quote blocked when BOPConditionalSurcharge modifier is non-zero (PC-30334)

Issue – In the base configuration, businessowners line of business, quote validation requires a justification if the `BOPConditionalSurcharge` modifier has a non-zero credit/debit. However, the screen does not include a justification entry field.

Workaround – Deactivate the validation on the `BOPConditionalSurcharge` modifier or add another column to the list view to capture the justification.

Upgrade to PolicyCenter after running ETL Product Model Loader extension pack (DOC-10285)

Issue – Upgrading to PolicyCenter 8.0.3 or later after running ETL Product Model Loader extension pack may require a manual drop of some tables.

In PolicyCenter 8.0.3, the ETL Product Model Loader Extension Pack functionality moved into the core PolicyCenter product. The extension pack created ETL database tables with `pcx_` as the prefix. PolicyCenter 8.0.3 and later create ETL database tables with `pc_` as the prefix. If you ran this extension pack with a prior version of PolicyCenter, you will have stale tables after upgrading to PolicyCenter 8.0.3 or later. The upgrade does not fail, but the server detects extra tables in the database.

Workaround – To remove these stale tables, manually drop the old `pcx_` tables created by the ETL Product Model Loader extension pack. The ETL Product Model Loader Extension Pack created these tables:

- `pcx_etlclausepattern`
- `pcx_etlcovtermpattern`
- `pcx_etlcovtermoption`
- `pcx_etlcovtermpackage`
- `pcx_etlpackterm`
- `pcx_etlmodifierpattern`
- `pcx_etlratefactorpattern`

Product Designer Known Issues

Cannot use Product Designer to edit an extended object with no base object (PC-27763)

Issue – If you attempt to edit a part of the product model that is based on an entity extension file (`.etx`), and that extension has no base entity file (`.eti`), Product Designer displays errors and refuses to display the contents of the file.

Although Guidewire best practices recommend that you never create an entity file extension (`.etx` file) when no base entity (`.eti` file) exists, Guidewire Studio did not enforce this recommendation. However, the current version of Product Designer does not allow the use of an entity extension without a corresponding entity base file. If such an entity extension exists, Product Designer displays a **NullPointerException** when you attempt to access the affected objects. You then can use the Back command in the browser to view other pages, but you cannot configure the affected objects.

Workaround – Locate all `.etx` files that do not have corresponding `.eti` files, and rename them with an `.eti` extension.

PolicyCenter does not start in single currency display mode if the product model is configured to use multiple currencies (PC-27829)

Issue – Product Designer allows users to configure multicurrency features without regard for the setting of the `MultiCurrencyDisplayMode` in the PolicyCenter `config.xml` file. If you configure multiple currencies in a product model object, and the value of the `MultiCurrencyDisplayMode` property is `SINGLE`, PolicyCenter does not start.

For example, if you add clauses that specify multiple coverage currencies and the PolicyCenter `MultiCurrencyDisplayMode` is set to `SINGLE`, the server fails to start and displays messages similar to the following:

```
ERROR ... PolicyLinePattern : ... [field : AvailableCoverageCurrencies] : - Multiple Coverage
Currencies defined for product not allowed in single Currency mode
ERROR ... [ExclusionPattern] : ... [DirectCovTermPattern] : ... [LimitsSet] : - The pattern specifies
limits for different currencies in SingleCurrencyMode
```

Workaround – If you intend to deploy a product model on PolicyCenter servers that are not configured for multiple currency display mode, do not configure multiple currencies in any part of the product model. However, you can test

multicurrency products on a staging server configured for multicurrency while the production system is configured for a single currency.

An exception might appear when you run Product Designer (PC-28531)

Issue – An `IncompatibleRemoteServiceException` might appear if you run Product Designer 8.0.3 or later after running a previous version of Product Designer.
Workaround – Refresh the browser cache by selecting `Ctrl+F5`.

Cannot edit a row in the question choices list (PC-28498)

Issue – If you use Google Chrome you cannot position the mouse to select a row in the **Question Choices** list.
Workaround – Use the keyboard to select the row. You can then edit the text.

System table cells display public IDs when referencing another system table (PC-28368)

Issue – If the cells in a particular column of a system table are referencing values from another system table, only the public IDs of those values are shown, not the values, when the system table is edited in Product Designer. In addition, users cannot search for values in these system table cells. The referenced system table, however, does not display the public ids so they cannot be viewed from within Product Designer.
Workaround – View the XML directly for the system table being referenced to determine the public IDs associated with the values that need to be added, edited or deleted. You can then use those public IDs directly.

Invalid audit schedules when audit schedule properties file is changed while user is editing audit schedule (PC-28146)

Issue – If you edit an audit schedule in Product Designer at the time that the associated audit schedule properties file is updated, you may not see the properties file changes reflected in Product Designer. This could result in an invalid audit schedule (that matches the previous audit schedule properties file).
Workaround – Perform the following steps:
- In Product Designer, revert the changes to affected audit schedules.
- To load the audit schedule properties file changes into Product Designer, restart Product Designer, logout, or change the changelist.
- Reopen the affected audit schedules.

Studio/Platform Known Issues

Renaming method or property throws ParseResultsException (PL-16633)

Issue – If you rename a property or a method or you change a method signature, and a workflow references that property or method in a Gosu field, PolicyCenter throws `ParseResultsException`. This is the intended behavior.
Workaround – Restart the workflow engine. To do so:
1. Log in to PolicyCenter using an administrative account.
2. Access **Internal Tools→Reload**.
3. Click **Reload Workflow Engine**.

Long text in table cells can add white space to the right of the page (PL-28288)

Issue – In an editable list view, extremely long text entered in a single cell can cause additional white space on the right side of the page. Long text is text that occupies approximately the entre width of the screen. This issue occurs primarily in Chrome.
Workaround – If you expect users to enter large amounts of text into the cells of a column, configure the column to support text wrapping.

Administrative command-line tools cannot refresh WSDL (PL-29021)

Issue – Administrative command-line tools rely on web service implementation classes such as `MaintenanceToolsAPI.gs`. If you change the web service implementation classes, administrative tools might fail

because the tool's WSDL does not match the server WSDL. Some changes do not affect the WSDL. For example, adding a `@WsiPermission annotation`.

Workaround – Do the following:

From the `PolicyCenter/bin` directory, at a command prompt type the command:

```
gwpc regen-soap-api
```

In Windows Explorer, copy the WSDL files from the location:

```
PolicyCenter/soap-api/wsi/wsdl
```

to:

```
PolicyCenter/admin/res/wsi
```

No entity changed event fired through a one-to-one entity relationship (PL-26224)

Issue – The existence of a one-to-one entity relationship between two entities prevents an entity change event from being fired, even if it is supposed to. For example, in ContactManager there is a foreign key from `ABContact` to `Address` for the primary address, and a one-to-one relationship from `Address` to `ABContact`. Because of the one-to-one link, no event changed event is fired for any change in the primary address.

More generally, if entity E1 has a one-to-one relationship to entity E2, no entity changed event is fired for E2 if a property on E1 changes.

Workaround – You might be able to take advantage of a secondary change caused by the initial change, with the secondary change triggering the event to be fired instead. In the previous ContactManager example, making sure that changes to all properties in `Address` cause a `History` record to be generated ensures that the entity changed event will be fired.

Free-text Search indexes cannot be upgraded from 7.0 to 8.0 (PL-29062)

Issue – To upgrade free-text search in PolicyCenter from 7.0 to 8.0, you install the newer version of the Guidewire Solr Extension in 8.0 on top of the version from 7.0. Because the PolicyCenter data model also changes, the free-text search indexes from 7.0 become obsolete. Guidewire does not support migrating free-text search indexes from 7.0. to 8.0.

Workaround – For Oracle, upgrade the Guidewire Solr Extension by following instructions in the *Upgrade Guide*. Then, run the new free-text batch load command to rebuild the indexes based on the new data model, after you complete your database upgrade. For SQL Server, Guidewire is aware of potential performance problems running the free-text batch load command after you complete your database upgrade.

Chrome browser cannot display product documentation in HTML format (DOC-7251)

Issue – If you use the Google Chrome browser, you can view the HTML Guidewire product documentation only if it is served by an HTTP server using the `http://` protocol. The Chrome browser cannot load HTML product documentation from your local disk by using the `file://` protocol.

Workaround – Use a different browser, such as Microsoft Internet Explorer. If you use the HTML documentation only to access the PDF files, you can open them directly in the `pdf` subdirectory of the `doc` directory.

New mechanism for reloading Gosu classes (DOC-8218)

Issue – In past releases, you could modify your PCF files and Gosu classes in Studio, and then reload the changes into the running server by pressing `Alt+Shift+L` in the application user interface. This shortcut no longer loads Gosu classes.

Workaround – To have the server reload your Gosu classes, in Studio, click **Build→Make Project**. When Studio is finished compiling your project, the changes will be loaded. You can also restart your server to load the Gosu classes.

Alignment Property Does Not Work for Inputs in a DetailView (PL-29429)

Issue – Inputs in a detail view do not align to the right or center. They always align to the left, regardless of the value of the `align` property.

Workaround – If you need to right align a series of items such as a set of monetary values, put them into a list view instead of a detail view. You can then set the alignment as needed.

JMS inbound integration implementations must be OSGi or Java plugins (PL-32054)

Issue – Writing JMS inbound integration handlers requires casting an object to the type `javax.jms.Message`. In Gosu, for some app servers this fails with the message: "Caused by: gw.lang.parser.exceptions.ErrantGosuClassException: GosuClass test.GosuJMSMessageReplyHandler has errors, and cannot be used at runtime."

Workaround – Implement JMS handler code for inbound integration in Java. Guidewire recommends that for inbound integration Java code, implement your code as an OSGi plugin, not a standard (non-OSGi) Java plugin.

Guidewire PolicyCenter 8.0.3 Release Notes

Release 8.0.3

Release Notes Update

IMPORTANT These release notes replace the release notes that were included with the official product release. Please disregard the earlier version of the release notes.

Overview

These release notes contain the following:

- "Release Information" on page 369
- "Installing This Release" on page 370
- "Support" on page 370
- "Major Issues and Changes" on page 370
- "Improvements and Resolved Issues" on page 371
- "Known Issues and Limitations" on page 382

Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

IMPORTANT If you skipped one or more upgrade releases to PolicyCenter, be sure to read the release notes for those releases to learn about changes and fixed issues.

Version Number

This release of Guidewire PolicyCenter is 8.0.3.

- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 7.0.1 or later. The latest maintenance release is preferred.

If you are implementing multiple settlement currencies, BillingCenter 8.0.2 or later is preferred.

- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 7.0.0 or later. The latest maintenance release is preferred.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 7.0.7, 8.0.1, or later. The latest maintenance release is preferred.

Installing This Release

Refer to the following:

- The *PolicyCenter Installation Guide* for general installation information
- Prior PolicyCenter release notes for any versions that you have skipped

Refer to the following:

- The *PolicyCenter Installation Guide* for general installation information
- Prior PolicyCenter release notes for any versions that you have skipped

Upgrade Information

Incorrect warning message when running Configuration Upgrade Tool (PL-28723)

When upgrading from one PolicyCenter 8 release to another PolicyCenter 8 release, running the Configuration Upgrade Tool may produce following warning message:

```
WARN cannot find Emerald base configuration zip, this could indicate a problem...
```

If the `PolicyCenter/modules/base.zip` file does exist, then you can safely ignore this warning.

Support

For assistance, visit the Guidewire Community.

Guidewire Customers

- https://community.guidewire.com

Guidewire Partners

- https://partner.guidewire.com

Major Issues and Changes

This topic describes major issues and changes that can affect your installation.

New and Changed Features

For information on new features and other changes, see the topic "New and Changed in PolicyCenter 8.0.3" in the *New and Changed Guide*.

Product Model File Restructuring

Restructured the product model files in the `config/resources/productmodel` directory to facilitate adding and removing lines of business and state deliverables. The restructuring does not affect the Product Designer user interface. The restructuring included:

- New directory structure that separates files by policy line, product, and jurisdiction
- Lookup files organized into jurisdiction-specific and non-jurisdiction-specific files

Improvements to List View Navigation (PL-26518)

The following improvements have been made to the behavior of selecting and navigating within a list view:

- Selection now favors selecting a cell rather than a row. A row is still selected where required, such as when selecting a row in a list detail view.
- Use the arrow keys, `Tab` (next cell), and `Shift+Tab` (previous cell) to navigate within in a list view.
- With an editable cell selected, press `Enter` to begin editing it. To leave edit mode, press `Esc` or click outside of the list view.
- When editing a list view cell, press `Tab` or `Shift+Tab` to remain in edit mode and navigate to the next or previous editable cell.
- For cells that have special controls, use standard keyboard actions to change the cell value. For example, press `Space` or `Enter` to select a check box or radio button, press `Down Arrow` to activate a drop-down list, and so on.

Base PCF File Changes

The link below requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 8.0.2 to 8.0.3

To view a report of the changes to the base PCF files, *click here*.

Base Resource Changes

The link below requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 8.0.2 to 8.0.3

To view a report of the changes to the base resources, *click here*.

Changes in This Release Provided in Upgrade Diff Report

Guidewire provides a report detailing certain differences between the current release and your prior release. This report describes changes in display keys, entities, typelists, and the Gosu API. To obtain your custom Upgrade Diff Report, visit the Guidewire Community.

Improvements and Resolved Issues

This topic describes improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This list is not intended to be comprehensive.

- "PolicyCenter improvements and resolved issues" on page 372
- "Product Designer Improvements and Resolved Issues" on page 374
- "Platform improvements and resolved issues" on page 376
- "Documentation improvements and resolved issues" on page 382

PolicyCenter improvements and resolved issues

ID	Description
Data Model / Core Entities	
PC-24839	Create a database consistency check for `BuildingImprovement` and `BuildingSide` to ensure that no duplicates exist. For example, the check ensures that: • A building never has more than one Front side at any given time, by checking `BuildingSide.BuildingSideType`. • A building never has more than one Heating at any given time, by checking `BuildingImprovement.BuildingImprType`.
Diffs	
PC-23910	The differences display in the **Policy Review** screen and between transactions now includes two additional fields: `RateAsOfDate` and `WrittenDate`. These two fields are configurable and can be removed by modifying the `PolicyPeriodDiffPlugin#addPolicyPeriodDiffItems` method.
Financials	
PC-25505	Added loss ratio fields on the **Summary** screen.
General	
PC-20135	Fixed an issue where the `Account.createConversionRenewalWithBasedOn` method altered the `BasisAmount`.
PC-23811	Added a display key for a hard coded string in `AccountClaimLV.pcf`.
PC-23907	In previous versions, it was necessary to load the Tiny sample data before the Small or Large sample data when integrated with BillingCenter. This is no longer necessary. This was previously known issue DOC-9605.
PC-23988	Fixed a known issue where rewrite jobs incorrectly calculated the expiration date in some cases.
PC-24255	The fields and values to be held constant in a `DocumentSearchCriteria` are now explicitly callable by sending them into the `setFixedContextWith` method. These values will not be cleared by calls to the `setRelatedTo` method, but they can be changed.
PC-24383	Fixed an issue where additional Account Contacts cannot be removed on the **Account Contacts** screen after being copied to the Account during a Rewrite New Account policy transaction.
PC-24440	With this change, quote purging, which prunes and purges policy periods and policy transactions, is disabled by default in the base configuration. To enable quote purging, set the `PruneAndPurgeJobsEnabled` and `PurgeOrphanedPolicyPeriodsEnabled` parameters to true in `config.xml`. These parameters must be enabled for the Purge and Purge Orphaned Policy Periods batch processes to remove policy transaction data from the database. In the base configuration, these parameters are set to false. You must also uncomment the Purge and Purge Orphaned Policy Periods batch process schedule entries in `scheduler-config.xml`. The entries are Purge and `PurgeOrphanedPolicyPeriod`. In the base configuration, these entries are commented out.
PC-24658 PC-24837	With this change, the Purge Rating Worksheets batch process is disabled by default in the base configuration. To enable this batch process, set the `PurgeWorksheetsEnabled` parameter to `true` in `config.xml`. This parameter must be enabled for the Purge Rating Worksheets batch process to remove rating worksheets from the database. In the base configuration, this parameter is set to `false`. You must also uncomment the Purge Rating Worksheets batch process schedule entry in `scheduler-config.xml`. The entry is `purgeworksheets`. In the base configuration, this entry is commented out.
PC-24959	Fixed an issue which caused the `JobNumberGenPlugin.getNewJobNumber` method to generate negative numbers.
PC-25053	Fixed a previously known issue where rewrite jobs did not correctly calculate the end of a the new policy term in certain circumstances.
PC-25241	Introduced `PCNumberFormatUtil` utility class which provides Gosu accessibility for locale aware conversions. Updated several instances of `String` to `BigDecimal` conversion to be locale aware.

ID	Description
PC-25388	Fixed an issue that caused the exception "`DB Exception - database attempted to divide by 0`" when performing a calculation on a date range that contained a leap day.
PC-25421	Fixed a issue in the **Team** screens. Filters now use the current date as a reference date instead of the date when the filter was first used. Removed caching of the first used date.
PC-25540	Modified the `PolicyPeriod` GXModel sample XML so that costs can easily be associated with the coverage and coverable. This improvement was made for processing High Volume Quote requests. This simplifies your implementation of the PolicyCenter system of record retrieving the GX model of the quote from the external quoting database. The following changes were made, specifically for the Personal Auto line: • At the `PolicyPeriod` level, instead of all the costs, only the non-premium costs (such as tax) are listed. All premium costs (such as coverage costs) are listed under the `PersonalAutoLine` section. • For every coverage cost, the vehicle VIN is now specified. • `PolicyContactRoles` and `PolicyLocations` are now listed at the `PolicyPeriod` level. • Added additional properties in various objects.

Integration

ID	Description
PC-23869	Fixed an issue on the **Payment Screen** where a Policy Renewal did not retain the audit reporting plan (audit schedule pattern selection).
PC-25438	Fixed an issue with **Search→Account** after ContactManager merged duplicate PolicyCenter account contacts.
PC-25605	Fixed an issue in the integration between PolicyCenter and ContactManager which caused both applications to throw exceptions when ContactManager merged duplicate PolicyCenter account contacts.

Jobs

ID	Description
PC-24034	Fixed an issue which caused an illegal argument exception in base data copy in side-by-side quoting.
PC-24928	In **Side-by-Side Quoting** screen, fixed layout of multiple Coverage Terms and problems with checkboxes.

Rating

ID	Description
PC-24304	Fixed an issue in the rate routine editor where, in certain circumstances involving an IF statement, a correct rating algorithm was considered invalid.
PC-24435	Fixed an issue where garbage collection followed by a type system refresh caused rating to give invalid quotes. Type system refresh only occurs when the system is in development mode, so this did not impact production systems.
PC-24823	Reduced memory requirements for storing rate table data.
PC-24889	Fixed an issue which caused a difference in costs between Cancellation and Rewrite policy transactions. Added `StartOfRatedTerm` property to `PolicyPeriod` for use in prorating policies rewritten to end of term. Rewrite to end of term and policies based on it had been using the period start date for proration, but now they use the beginning of the term instead.
PC-25229	Fixed an issue where a rate book would not validate even after correcting the problem. The message was **Only Exact or Range matches are allowed to be higher Priority than Interpolated match**.
PC-25301	In rate routines, fixed an issue in conditional expressions where the option to clear the **Comparison** operator was available but difficult to detect. With this release, you can clear the **Comparison** by selecting **<none selected>**.
PC-25313	In rate routines, fixed an issue in conditional expressions where the right operand was editable even though Constant was disabled. In this release, the right operand is no longer editable in this situation.
PC-25324	The Rate Routine editor now keeps track of the type of local variables and only presents options that are type-compatible.
PC-25390	For added error detection, PolicyCenter now re-validates rate table definitions when rate books are promoted or imported.
PC-25427	On the **Rate Routine Details** screen, if the parameter set used by that rate routine has been deleted, PolicyCenter now displays an alert message.

ID	Description
PC-25445	In previous releases, it was possible to create a rate book where the content and the book had differing values for `PolicyLinePatternCode`. This occurred if you first added appropriate content and then changed the `PolicyLinePatternCode`. This is now treated as invalid and prevents promotion of the rate book.
Revisioning	
PC-24121	Fixed an issue which caused notes and documents to point to an incorrect policy period after handling preemption. With this fix, notes and documents are linked to the new policy period after handling preemption.
Security	
PC-23890	Improved the searching of documents by enabling users with producer code security to use account-based security when the policy is not specified on the document.
Underwriting	
PC-23947	Fixed an issue on the **Side by Side Quoting** screen where clicking on **Resolve** did not take the user to **UW Activity/Risk Analysis** screen.
PC-25217	Add an `EstimatedPremium` column to `PolicyPeriod`. For commercial lines of business, the user can provide the **Estimated Premium** on the **Policy Info** screen on Submission, Renewal or Rewrite policy transactions. For Renewals and Rewrites, the value is populated from the value on the prior period. PolicyCenter updates the value with the **Total Premium** when the policy quote is released (**Print Quote** button). It is also updated when the policy is bound or issued. The value also appears on the **Policy Review** screen.
Upgrade	
PC-23914	In this release, the `EmeraldPopulateUWIssueValueVersionTrigger` upgrade trigger, which appended the default currency to the value column of the `pc_uwissue` table, has been removed. Now for `MonetaryAmount` objects created in previous releases, PolicyCenter automatically uses the default currency to create the `MonetaryAmount` object if the currency is missing from the value. This happens at run-time.
PC-24844	Added upgrade version check to ensure that (`PolicyPeriod`, `BillingId`) is unique within `pc_paymentplansummary`. This check is to avoid a runtime error when PolicyCenter detects multiple matching rows in `pc_paymentplansummary`. In PolicyCenter 8.0.2, the relationship between `PolicyPeriod` and `PaymentPlanSummary` was changed to a one-to-one. However, the upgrade trigger to handle this change did not previously detect duplicate non-retired rows in `pc_paymentplansummary` with the same (`PolicyPeriod`, `BillingId`) pair. This could cause runtime errors after upgrading to PolicyCenter 8.0.2. This fix adds a version check that aborts the upgrade when detecting duplicate rows of this type. The version check prints out a list of duplicate cases, and you must either retire or remove duplicate rows before the upgrade can proceed. This fix also changes the archive document upgrade trigger such that duplicate rows are marked as retired after restoring from archive. This prevents runtime issues with the new one-to-one relationship.
PC-25015	In `config.xml`, the `WorkingsheetContainerAgeForPurging` parameter has been renamed `RatingWorkingsheetContainerAgeForPurging`. This fix adds a configuration upgrade trigger that renames the parameter.

Product Designer Improvements and Resolved Issues

ID	Description
Changelist Management	
BCC-2273	Fixed an issue which caused an exception to occur when the user navigates to another page after the current workspace has become invalid.
Conflict Handling	
BCC-1972	Fixed an issue that caused an exception when one user edited a coverage category that another user had already deleted.
BCC-2175	Fixed an issue which caused duplicate items to appear in the **Product Offering Selections** page after a conflict had been resolved.

ID	Description
BCC-2258	Fixed an issue where adding a product with a policy line removed by another user in a concurrent session caused an exception.
BCC-2277	Fixed an issue where all system table rows were shown after multiple users made conflicting changes to the same rows.

Internationalization

ID	Description
BCC-2215	Fixed an issue which prevented Product Designer from handling whitespace values correctly in localized system table columns.
BCC-2224	Fixed an issue where adding but not committing a display key value and then changing the language moved the value to the new language in the **Display Keys by Language** popup. The display key value now correctly remains associated with the original language.
BCC-2424	Fixed an issue which caused a multi-language version of Product Designer to lose display key values when the user logs out and then logs back in.
BCC-2485	Validation for required fields has been made stricter. A new or existing required field cannot be set to an empty space (" "). This behavior was introduced to ensure that system tables with required fields are populated correctly for PolicyCenter. Note that a system table field configured as `trimWhitespace=false` in the metadata may still require users to enter non-whitespace values in Product Designer.

Policy Lines

ID	Description
BCC-2127 BCC-2402	Added validation to ensure that there is no mismatch in currency between the default coverage term option and the selected coverage term option.

Product Model (Overall)

ID	Description
BCC-2195	Fixed an issue where Product Designer failed to detect a user's attempt to commit duplicated, invalid pattern data.
BCC-2206	Fixed an issue that caused an exception to occur when the **StartEffectiveDate** field was cleared and the user then clicked the **Today** button in the **Calendar** popup.
BCC-2401	Improved the handling of situations where a typekey option's typelist is changed when a default typelist value already exists.

Server & Backing Model

ID	Description
BCC-2411	Fixed an issue where entering an empty or invalid value in the **Typelist** field on the **Coverage Term** page caused an exception to occur.
BCC-2469	Fixed an issue which occurs in a Perforce workspace when a user's actions in Product Designer result in a product model file being deleted and then added back. In previous versions of Product Designer, the file would incorrectly have been deleted when the changes were committed. In this release the file is not deleted but is marked for edit in Perforce.

Sync With PC

ID	Description
BCC-2295	Fixed an issue which caused unnecessary warning messages to be generated during product model synchronization.

System Tables

ID	Description
BCC-2276	Fixed an issue which prevented system tables from being sorted by column.

User Management

ID	Description
BCC-1553	Increased the length of the title for the **Reset Password** dialog box.

Validation

ID	Description
BCC-2418	Added an error indicator for an empty **Typelist** field when the coverage term type is **Typekey**.

Workspace Management

ID	Description
BCC-2421	Fixed an issue which prevented a new workspace from being created in Internet Explorer 10.
BCC-2444	Fixed the error message that appears when an invalid folder is specified in the **Add Workspace** dialog.

Platform improvements and resolved issues

The following are the primary improvements and issues corrected in this release:

ID	Description
Archiving	
PL-30622	Quote purging no longer incorrectly deletes data from bound periods.
Authentication	
PL-30775	Removed the unused configuration parameter `ShouldSynchUserRolesInLDAP`.
Batch Processes	
PL-30718	The free-text batch load command now works correctly with JNDI data sources.
Build Infrastructure	
PL-27679	The `gwpc regen-java-api` command now generates Java API Javadoc into the `java-api/doc/api` directory.
Command Line Tools	
PL-27608	Removed extraneous files from the output of the command `zip-changed-config`. The list used is in `com.guidewire.tools.upgrade2.merge.ConfigMergeList#EXCLUDE_DIRS`. Also added new option `-e`, which enables you to specify a semicolon-separated list of directories to exclude.
Consistency Checker	
PL-30098	Fixed an issue where Database Consistency Checks that checked date parameters would compare against an incorrect date.
Database Support - DB2	
PL-30469	Replaced parameter `identifyQueriesViaComments` in `config.xml` with new parameters `IdentifyQueryBuilderViaComments` and `IdentifyORMLayerViaComments`. The `IdentifyQueryBuilderViaComment` parameter instruments high-level database objects constructed by using the query builder APIs. `IdentifyQueryBuilderViaComment` is set to `true` in the base configuration. The `IdentifyORMLayerViaComments` parameter instruments lower level objects, such as beans, typelists, and other database building blocks. `IdentifyORMLayerViaComments` is set to `false` in the base configuration.
Database Support - Oracle	
PL-29153	The AWR report will now only report activity for the targeted schema (Guidewire database) on an Oracle server hosting multiple schemas.
PL-29868	The configuration of Oracle range and hash partitioned indexes has changed from the last release. If you have any indexes configured for Oracle partitioning, read the topics on partitioning under "Configuring the Database" in the *Installation Guide* and make changes accordingly.
PL-30770	Updated upgrade options in `database-config.xml` files to show default values. Removed an unneeded setting.
PL-31513	Fixed an issue where Oracle LOBs were mistakenly set with the `nologging` attribute.
PL-31548	If the AWR report experiences an XML parsing error while reading from `DBMS_SQLTUNE.REPORT_SQL_MONITOR`, the read of the data will be skipped and an explanatory message printed in the **Errors** section of the report.
PL-31810	Fixed a problem with Guidewire AWR download when the database is Oracle 12.1.0.2 patchset or higher.
Database Support - SQL Server	

ID	Description
PL-30578	Two pages have been added to the SQL Server DMV performance report that aggregate active session statistics by user and by action.
Document Management	
PL-31555	Fixed an issue so that you can now use asynchronous document storage (`AsyncDocumentContentSource`) even if the Plugins registry has an enabled implementation for the `IDocumentMetadataSource` plugin interface.
Email	
PL-27557	Added an additional constructor in `EmailTemplateSearchCriteria` to add a callback interface and set a variable when it does not exist in the current evaluation context. This is to cover cases where the environment in which the templates are retrieved (filtered) and the environment in which they are evaluated are not the same.
Entities/Metadata	
PL-30357	Fixed an issue where the `gwpc regen-java-api` command would throw an exception when processing a typelist that had more than 4,000 typecodes. Now 8,000 typecodes are supported.
External Entities	
PL-28992	Resolved an issue that caused the `gwpc regen-java-api` command to sometimes fail if the flag `-Ddeprecated=true` was specified.
Gosu	
PL-29715	Fixed an issue that could cause runtime exceptions if you set the `config.xml` parameter `WarnOnImplicitCoercion` to the value false.
IntelliJ IDE - Compiler	
PL-29534	Compilation errors for a widget template are now shown against the containing PCF file rather than against the template.
PL-30918	Fixed an exception that would occur when a PCF file contained a widget template that contained errors.
IntelliJ IDE - Entity Editor	
PL-29124	Fixed an issue where comment elements such as `<!-- Comment -->` inside an entity or typelist definition file were removed when edited in Studio.
IntelliJ IDE - Localization	
PL-30209	You can now switch the language in Studio without needing to restart. To do so, navigate to **File→Settings→Guidewire Studio**, and then under **Language Settings**, click the language.
IntelliJ IDE - PCF Editor	
PL-30102	Removed a non-functional link in the pop-up help text when hovering over a PCF widget in Studio.
IntelliJ IDE - Plugins	
PL-29776	If you configure Guidewire Studio to work with IDEA IntelliJ Ultimate, now the OSGi Plugin Editor also uses Ultimate.
IntelliJ IDE - Product Model Editor	
PL-29996	Fixed an issue that prevented creating Gosu enhancements for product model coverages.
IntelliJ IDE - Webservices Editor	
PL-28462	Fixed an issue with the Web Services Collection Editor in Studio where it was unable to download imported schemas correctly.
PL-29920	Fixed an error that would occur in the Web Services editor in Studio when clicking Fetch Updates when the `suite-config.xml` file contained multiple values for the env attribute.
Messaging	

ID	Description
PL-27870	For messaging use, there are Gosu APIs that provide a `HashMap` that exists across multiple rule set executions. The APIs are methods on the result object of Gosu expression `messageContext.SessionMarker`. Older methods `addToTempMap` and `getFromGetMap` manipulate a `HashMap` that exists uniquely for each messaging destination. For a hash map that exists across all rules for all destinations, use the new methods `addToSessionMap` and `getFromSessionMap`.
PL-28769	Destination ID, message ID, sender ID, and requester user name of retried and skipped messages are now logged.
PL-28952	On Oracle with this release, PolicyCenter includes Oracle SQL hints in queries that select messages for a message destination. Oracle SQL hints can improve query performance. You can use the new parameter `UseOracleHintsOnMessageQueries` in `config.xml` to enable and disable the use of Oracle SQL hints at run time.
PL-29139	Several changes for inbound integration APIs: • The `InboundIntegrationPlugin` plugin interface is now called `InboundIntegrationStartablePlugin` • For messaging use, the new plugin interface `InboundIntegrationMessageReply` shares features of the inbound integration plugin but is a message reply plugin not a startable plugin • Existing file and JMS integrations are now available also as `MessageReply` variants. See the *Integration Guide* in the "Multi-threaded Inbound Integration" chapter.
PL-30119	Fixed an issue with the JMS inbound integration APIs that caused the exception "JMS Integration Service not supported on server runtime".
PL-30689	The application no longer performs redundant locking and unlocking of entities if a message destination does not use the optional `MessageRequest` plugin.
Other - Developer IDE	
PL-27489	Reduced the number of unnecessary library files that `regen-java-api` generates into the `lib` folder.
Other - Integration	
PL-27845	Custom implementations of the inbound integration plugin interface (`InboundIntegrationPlugin`) now support Gosu implementations.
PL-30148	For use with the `DataChangeAPI` web service, new Gosu APIs for your data change code configure logging to the data change user interface. To log field-level entity changes, call `DataChange.util.setDetailResultWriting(bundle)`. To log arbitrary text, call `DataChange.util.ResultsWriter.append("your message")`.
PL-30566	The multi-threaded inbound integration `inbound-integration-config.xml` file now supports configurations that vary based on server environment set with the env system configuration setting. See the *Integration Guide* in the "Inbound Integration" chapter.
PL-30910	For custom implementations of the `InboundIntegrationStartablePlugin` plugin interface, all method signatures of the start and stop methods must call methods of `gw.api.integration.inbound.CustomWorkAgent`. As a method argument, pass the plugin name that you used in your `inbound-integration-config.xml` file with the name attribute on the `<custom-integration>` element. In each start method, call `CustomWorkAgent.startCustomWorkAgent(pluginName)`. In each stop method, call `CustomWorkAgent.stopCustomWorkAgent(pluginName)`. This new API does not apply to writing handlers for the built-in file or JMS integrations.
Other - Persistence	
PL-28950	This release improves the performance of the Populate Search Columns batch process.
PL-30644	Added configuration parameter `BatchProcessPopulateSearchColumnsParallelDML` to enable or disable parallelism in the populate search columns batch process. By default parallelism is enabled.
Other - Web UI	
PL-29767	There was an issue with localized date formats that used a single-digit day or month, which sometimes resulted in swapping the day and month. Single digits are now supported for these values. In addition, the following date separators are supported by default: / (slash; example: 18/2/2015)

ID	Description
	. (dot; example: 18.2.2015) - (dash; example: 18-2-2015) `<space>` (example: 18 02 2015) `<nospace>` (example: 18022015)
PL-31050	Fixed an issue where the `DateTime` input widget was not navigable with the keyboard.
Queries	
PL-29695	Fixed the `forEmpty` method on `Query` objects to work properly on root queries and have no effect on subqueries.
Search	

PL-29222	This release introduces the env attribute to the `<document>` element in `solrserver-config.xml`. You can use the value you set in `solrserver-config.xml` as a Java VM environment variable to run PolicyCenter with that Guidewire Solr Extension instance without editing the `solrserver-config.xml`. For example, your `solrserver-config.xml` file contains the following document elements.
	`<PC-only content>`
	`<document name="policy" archive="false" servername="embedded"/>`
	`<document name="policy" archive="false" servername="localhttp" env="local"/>`
	`</PC-only content>`
	`<CC-only content>`
	`<document name="claimcontact" archive="true" servername="embedded"/>`
	`<document name="claimcontact" archive="true" servername="localhttp" env="local"/>`
	`</CC-only content>`
	By default, PolicyCenter runs with the Solr server found in the first document element in `solrserver-config.xml`. In the preceding example, PolicyCenter runs with the embedded server by default. If you stop the PolicyCenter application server and restart it with `-Dgw.pc.env=sql` as an option, PolicyCenter runs with the Solr server found in the document element with the attribute env="sql". In the preceding example, PolicyCenter runs with an external instance of the Guidewire Solr Extension hosted locally.

PL-29476	Added the `zkhosts` property to Guidewire Solr Extension server configuration to specify in a comma separated list the hosts and ports of all members of the Zookeeper ensemble. Deprecated the `host` and `port` properties, which specify only a single Zookeeper instance.
PL-29911	Upgraded the Apache Solr free text search engine to version 4.7.2.
PL-31446	Configuration of application servers changed due to changes in logging. See "Free-text Search Setup" in the *Installation Guide* for the modified setup procedures.
Web - Configuration	
PL-29869	Fixed an issue where, in the `RowTree` widget, a `Cell` with a currency format type or a `CurrencyCell` would not show the currency format properly.
PL-30310	Fixed an issue where validation on a `PrivacyInput` was applying to the mask value instead of the actual value.
PL-30475	Fixed an issue where the `labelWidth` property in an `InputColumn` was not being applied for specific widgets.
Web - IE Support	
PL-29681	Fixed an error where the server would report that the file `toolbar-top-info-bg.gif` was missing.
Web - ListViews	
PL-26386	Fixed an issue when `postOnChange` is enabled where tabbing through menu items or inputs with a menu icon would not show focus on those fields correctly.
PL-29294	Fixed an issue with a newly added list view column appearing at the extreme right until the layout preferences are reset.
PL-29657	Improved the behavior of tabbing through list view cells.
PL-29769	Fixed an issue where header menus were appearing incorrectly for list views with the `colspan` attribute set.

ID	Description
PL-30595	Fixed an issue where the screen would reset its horizontal scroll position while editing a list view in Internet Explorer.
PL-30989	Fixed an issue where the helper icon in a cell would not be visible when the cell had focus.
PL-31015	Fixed an issue where a `LinkCell` was not accessible with the keyboard.
PL-31038	List views now reposition themselves into view horizontally if you use the keyboard to navigate to an element off the screen.
PL-31152	Fixed an issue where you could not use the keyboard to invoke actions on text in list view cells.
PL-31201	Fixed an issue where the `PrivacyInput` menu remained expanded when navigating the page using the keyboard.
PL-31298	Fixed an issue where you could not open the menus of a `ButtonCell` with a `MenuItem` by using the `Alt+Down Arrow` keyboard shortcut.
PL-31354	Fixed an issue in list views where `postOnChange` would cause the focus to be set incorrectly.
PL-31380	Fixed an issue where focus would be lost if a `postOnChange` occurred in the last cell of a list view.
PL-31403	Fixed an issue where `postOnChange` was not triggered in list views with only one editable cell in Internet Explorer.
Web - Other	
PL-28929	Fixed an error message that referenced a missing "perf-analyzer". The analyzer is an internal Guidewire tool that is not used in customer releases.
PL-30507	Fixed an issue in the `RunBatchProcessCommand` to display an error instead of a crash if a work queue contains zero workers.
PL-31341	Fixed an issue where the `no-store` value was omitted from the `Cache-control` HTTP response header in application responses.
PL-31813	Fixed an issue where calling an `ExitPoint` would fail.
Web - Styling	
PL-28575	The workspace panel had a property that enabled it to float while hovering. This behavior led to a confusing user experience because the panel would appear to pop up and down on user actions. The hover behavior has been disabled, and a larger button for expand/collapse has been added.
PL-30112	Fixed an issue where the `PreFormattedTextInput` PCF widget did not properly render HTML.
PL-30534	Fixed an issue where headings in the **Actions** menu were not appearing in a bold font.
Web - UI/Runtime	
PL-28186	Fixed an issue where the `PrivacyInput` field would be exposed if used during `postOnChange`.
PL-28224	Fixed an issue where the `PrivacyInput` widget was not masking values that were changed by client-side reflection.
PL-28581	Fixed an issue where actions in the `TitleBar` were not displayed properly.
PL-28784	Fixed an error that occurred in scatter charts after data axes were hidden.
PL-29118	To prevent CSRF (Cross-Site Request Forgery) attacks PolicyCenter now generates a unique Cross-Site Request Forgery (CSRF) token for each user session. The CSRF token is included in each request and used by the server to verify the legitimacy of the user request. See "Configuring Single Sign-on Authentication" in the *Installation Guide*.
PL-29230	Fixed a localization issue where the shadow placeholder text in the QuickJump box did not immediately refresh after the language was changed.
PL-29650	Fixed `RangeValueInput` to handle values that require HTML encoding (for example, an ampersand).

ID	Description
PL-29653	Fixed an issue where the screen would freeze or become blank when using typeahead in an `AddressAutoFillInput` widget.
PL-29689	Fixed an issue where detail views would lose focus after a `postOnChange`.
PL-29805	Fixed an issue where pressing `Alt+Z` to log out would do so even if there was unsaved work.
PL-29858, PL-30039	Fixed an issue with list views unexpectedly scrolling to the top after interacting with a radio button, check box, or combo box in a row.
PL-30055	Addressed an issue where keyboard input entered while a `postOnChange` event was in progress disappeared after the event completed. Keyboard input is now disabled until the event is complete.
PL-30072	Fixed an issue where `postOnChange` in the date widget required an additional user action before it would be triggered.
PL-30160	Fixed an issue where opening and then closing the `DateTime` widget would reset the minute value to the current minute of the user's machine.
PL-30207	Fixed an issue where you could not navigate out of a list view by using the `Tab` key.
PL-30492	Fixed an issue where using the **Unsaved Work** menu failed when a page title contained certain special characters (such as an ampersand).
PL-30586	Pressing the `Backspace` key no longer acts as a Back navigation command in the web browser.
PL-30772	Fixed an issue where the focus was not being set into a list view after a row was added.
PL-31001	Fixed an issue where the focus in a list view did not go to a newly added row.
PL-31027	Fixed an issue where list view rows with `colspans` specified did not handle tabbing correctly.
PL-31040	Fixed an issue where list view rows were highlighted incorrectly after columns were rearranged.
PL-31042	Fixed an issue where a list detail view configured to have an editable list view would create conflicting edits that would override each other and cause the data to be discarded. An impact of this change is that you now single-click a row to select it and double-click to edit it.
PL-31512	Fixed an issue where the calendar widget did not properly display Russian months.
PL-31610	Fixed an issue where reflection on the PCF widgets `RangeRadioInput`, `BooleanRadioInput`, and `TypekeyRadioInput` were not working properly.
Web Services - WSI (New)	
PL-31055	Fixed an issue in WSDL generation for services with multiple @Throws annotations for the same exception but different reasons. This caused an exception in some cases for WSDL consumers. The WSDL now merges related faults into one fault with all explanations concatenated in one comment. The fix prevents an exception that some WSDL consumers experienced.
Work Queues	
PL-31032	Improved performance of the work queue framework in finding available work items for work workers based on work item priority. The improvement introduces a new parameter, `WorkItemPriorityMultiplierSecs`, in `config.xml`, with a default value of 600 seconds.
PL-31043	Added `BulkInsertWorkQueueBase` for work queues with writers that typically select large volumes of work items. Instead of developing the work queue writer by implementing the `findTargets` method, implement the `buildBulkInsertSelect` method.
XMLElement (and XSD types)	
PL-28345	XSD types based on `XMLNode` have been deprecated, and will be removed in a future release.
PL-30793	Fixed an issue with WSDL import relating to handling simple data if a `complexType` is a `complexContent` extension of a `complexType` with a `simpleContent` extension.

Documentation improvements and resolved issues

ID	Description
DOC-2765	Added documentation on developing custom work queues. See "Custom Batch Processing" in the *Integration Guide*.
DOC-7022	Concepts and configuration of address autofill and autocompletion are now described in the topic '"Address Autocompletion and Autofill" in the *Globalization Guide*.
DOC-8027	Added a topic that has steps for adding a new address field. This example covers the classes, configuration files, and PCF fields that are involved in adding a new address field. Additionally, the example shows how to configure autofill and autocomplete functionality for the new field. See "Example: Adding a Country with a New Address Field" in the *Globalization Guide*.
DOC-8646	Added documentation describing how to localize `Contact` name information to "Configuring Name Information" in the *Globalization Guide*.
DOC-8869	Added documentation for how to edit, configure, and load zone data. See "Configuring Zone Information" in the *Globalization Guide*.
DOC-8672	Enhanced documentation on debugging the application within Studio. See "Debugging and Testing Your Gosu Code" in the *Configuration Guide*.
DOC-8873	Added documentation on viewing entity property values during debugging. See "Debugging and Testing Your Gosu Code" in the *Configuration Guide*.

Known Issues and Limitations

This section describes known issues with this release of Guidewire PolicyCenter.

- "PolicyCenter Known Issues" on page 382
- "Product Designer Known Issues" on page 386
- "Studio/Platform Known Issues" on page 387

Note: For maintenance releases, Guidewire often defers fixing configuration issues that require merging files during the upgrade. Workarounds to many of these issues are listed in the following topics. The goal of this policy is to make upgrades as straightforward as possible.

PolicyCenter Known Issues

Provide a way to filter out added and removed subentities in a policy differences tree (PC-7929)

Issue – Need a way to filter out the children of an added or removed entity in a policy differences tree.
Workaround – Guidewire is aware of this issue.

When using parallel sessions, approvals made by underwriter are not updated for second user simultaneously viewing same work order (PC-12322)

Issue – Approvals made by an underwriter are not updated for a second user simultaneously viewing the same work order. This occurs in the following sequence of events:
- An agent processes a submission and issues arise
- The agent calls an underwriter, and remains in the policy
- The underwriter approves the issue or issues
- The agent refreshes the **Risk Analysis** screen by going to a different wizard step and returning to the **Risk Analysis** screen

After this sequence of events:
- The new approvals do not appear on the agent's screen
- However, the agent can bind or issue the policy (based on the new approvals by the underwriter)

Workaround – The agent can leave the policy and return to it. The approvals appear as expected. Alternatively, the agent can see the approvals by clicking the **Details** button when issuing the policy.

Must enter VIN when moving from quick quote multi-version or side-by-side to full application (PC-13099)

Issue – In a quick quote, you do not have to enter the VIN or other fields that are required for a full application submission. When converting a quick quote with multi-version or side-by-side quoting to full application, to resolve validation issues, you must enter the VIN for each of the versions. In a job with side-by-side quoting, this issue can occur with fields that are not base data.

Workaround – Add code that copies this type of data to the other versions.

PolicyCenter does not send renewal billing instruction to BillingCenter (PC-15082)

Issue – On receiving payment notification from BillingCenter, PolicyCenter tries to bind. If the policy is in a draft status, PolicyCenter does not bind and send the Renewal Billing Instruction to BillingCenter. The policy might be in draft status after an underwriter edited something on the policy. This is an issue with BillingCenter 3.0.1 and later.

Workaround – Guidewire is aware of this issue.

BillingCenter integration: Automatic payment instrument is used to create invoice stream even if final selection is manual (PC-15307)

Issue – If an automatic payment instrument is setup and selected, the policy will create a new invoice stream with this payment instrument even if the user changes to Manual before issuing the policy.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Guidewire is aware of this issue.

BillingCenter integration: Removing alternate billing account when payment instrument selected generates WsdlFault (PC-15308)

Issue – If an automatic payment instrument is selected when using an Alt Billing Account to create a new policy, PolicyCenter throws a `WsdlFault` if the Alt Billing Account gets removed.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Contact Guidewire Support for assistance with this issue.

BillingCenter integration: New invoice streams not created when issuing policy with alternate billing account on new account (PC-15309)

Issue – Assume an account has an existing invoice stream. When this account is used as an **Alt Billing Account** for new policies on new accounts, a new invoice stream is not created when the new policy's new invoice stream periodicity matches the invoice stream on the **Alt Billing Account**.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Guidewire is aware of this issue.

Renewal processing for renewal conversion assumes BasedOn is populated (PC-15851)

Issue – The `startNewRenewal` method in `gw.webservice.pc.pc700.job.PolicyRenwalAPI` assumes the existence of a prior term or a `basedOn` version. Unless customized, this method can cause problems in certain convert-on-renewal scenarios. This method has been deprecated.

Workaround – If you are developing convert-on-renewal flows, use the `gw.webservice.pc.pc700.job.startConversionRenewal` method.

Create Baseline button not visible after navigating to the Create Baseline screen (PC-17809)

Issue – When you start the **Create Baseline** step, PolicyCenter kicks off a batch job to create the baseline quotes. When you click **Cancel**, the cancel request is sent to the batch job, but the current quote may take some time to stop.

If you click **Cancel**, go back and return to the **Create Baseline** screen. You may see the progress bar from the prior cancellation.

Workaround – Navigate to another screen and wait about a minute before returning to the **Create Baseline** screen.

SQL Query performance in free-text search (PC-18794)

Issue – There are open issues related to the performance of SQL query in free-text search. This occurs the first time that you load the PolicyCenter data into the Solr index database.

Workaround – Guidewire is aware of this issue.

Cannot quote a renewal if the expiring policy term has a handled preemption (PC-19416)

Issue – In Commercial Package Policy in the base configuration, if a policy term has a preemption that has been handled, when a renewal job is created, the **Edit Workorder** button is not always available on the renewal. When this occurs, the only options are to bind or not take the renewal.

Workaround – Guidewire is aware of this issue.

Copy-data with an out-of-range date (PC-20279)

Issue – When working with the copy data feature, if you specify a specific date to copy data from and that date is out of range for the policy, PolicyCenter presents a warning that no applicable policy version could be found. However, the date selector still indicates the requested date, not the date that PolicyCenter will use for the copy if you choose to proceed.

Workaround – As appropriate, modify the user interface to reset the date selector or to hide the ability to select data to copy.

Error when quoting after adding new location (PC-21656)

Issue – In the Commercial Auto, General Liability, and Commercial Property lines, users sometimes receive an error message after clicking **Quote** on the **Locations** screen indicating that there are no locations even when there are. This error could potentially occur in other lines of business and on other list views, depending on how they are configured.

Workaround – Click **Next** and then **Back** to navigate away from and back to the **Locations** screen before quoting. Alternately, remove the client side validation on the number of rows.

Reinsurance warnings not refreshed even after saving a draft (PC-21720)

Issue – When working with reinsurance, PolicyCenter displays validation errors and warnings both at the top and in the worksheet at the bottom of the screen depending on the action that caused validation to run (typically **Save**, or leaving the screen). Depending on the user flow, one or both of these areas may display stale messages, and the messages may not be the same.

Workaround – Refresh the messages in either or both locations before taking the action again.

Gosu classes not intended for use (PC-23009)

Issue – Some Gosu classes are visible in Studio, but are not intended for use. You can distinguish these Gosu classes because they have no visibility annotations (neither `@Export` nor `@ReadOnly`) and they are not in a `gw` package.

Workaround – Do not use these methods in configuration. The methods are unsupported and may change or be withdrawn without notice.

PolicyCenter allows owner and overriding-payer accounts to be the same (PC-23570)

Issue – In PolicyCenter, you can set the overriding payer on a list bill policy to the same account as that account which owns the policy. You set the overriding on the **Account→Billing** screen. Systems which receive data from PolicyCenter may report this as an error. For example, BillingCenter requires an overriding payer account to be different than the owner account.

Workaround – Add validation or other logic to PolicyCenter that ensures that, if present, an overriding-payer account is different than the owner account.

Unbound renewal policy transactions can provide invalid dates (PC-25288)

Issue – If a policy renewal is in the renewal process and the user goes from the renewal job wizard to the policy file summary, the as-of date default in the **Date** field of the **Actions** panel will not be entirely valid, as it will be the `editEffectiveDate` of an unbound period from an incomplete job.The quote and reinsurance, if enabled, screens display exceptions because appropriate periods do not exist on the specified dates.

Workaround – Edit `JobWizardInfoBar.pcf` to filter out any `PolicyPeriod` objects that cannot provide the `editEffectiveDate`.

Billing integration error when final audit is waived from Audit job (PC-25500)

Issue – When the user waives a final audit from within a running Audit job, the BillingCenter integration sends an incomplete message to BillingCenter. The message does not include the term number of the policy period. BillingCenter needs the term number to identify the BillingCenter policy period that needs the change.

Workaround – Guidewire is aware of this issue.

Insurance Suite integration issues (PC-25732, PC-25733, PC-23873)

Issue – There are issues with creating and sharing contact information between PolicyCenter, BillingCenter, and ContactManager if the applications are different versions. For example, these issues arise if you have PolicyCenter 8.0 integrated with BillingCenter 7.0. These issues include:

1. In the `BillingMessageTransport` Gosu class, the commented-out code in the `send` method is invalid. This commented code handles the case when ContactManager is unavailable and the message is ready to be sent. The commented code begins with:

   ```
   /*
      It is not recommended to wait for ContactManager to process the contact messages ...
   ```

   The `BillingContactUnsyncedException` method does not exist and the code does not compile.

2. When PolicyCenter 8.0 is integrated with BillingCenter 7.0, contacts sent to BillingCenter have a null `AddressBookUID` even though PolicyCenter has called ContactManager with the suggested `UniqueID`. This results in BillingCenter also creating the contact in ContactManager, leading to duplicate contacts.

3. When BillingCenter creates a new Contact for an Account that had been created in PolicyCenter, PolicyCenter does not recognize the Contact and throws a `BadIdentifierException`.

Workaround – Guidewire is aware of these issues. In addition:

1. Do not uncomment the code in the `BillingMessageTransport` class. Write code that handles the case when ContactManager is unavailable and the message is ready to be sent.

2. In the `bc700` package, update the sync method in the `PCContactInfoEnhancement` class to match the version in the `bc800` package.

Upgrade to PolicyCenter after running ETL Product Model Loader extension pack (DOC-10285)

Issue – Upgrading to PolicyCenter 8.0.3 or later after running ETL Product Model Loader extension pack may require a manual drop of some tables.

In PolicyCenter 8.0.3, the ETL Product Model Loader Extension Pack functionality moved into the core PolicyCenter product. The extension pack created ETL database tables with `pcx_` as the prefix. PolicyCenter 8.0.3 and later create ETL database tables with `pc_` as the prefix. If you ran this extension pack with a prior version of PolicyCenter, you will have stale tables after upgrading to PolicyCenter 8.0.3 or later. The upgrade does not fail, but the server detects extra tables in the database.

Workaround – To remove these stale tables, manually drop the old `pcx_` tables created by the ETL Product Model Loader extension pack. The ETL Product Model Loader Extension Pack created these tables:

- `pcx_etlclausepattern`
- `pcx_etlcovtermpattern`
- `pcx_etlcovtermoption`
- `pcx_etlcovtermpackage`
- `pcx_etlpackterm`
- `pcx_etlmodifierpattern`
- `pcx_etlratefactorpattern`

Product Designer Known Issues

When using Google Chrome, Changes and Help panels "bounce" open and closed (BCC-926)

Issue – Clicking the **Changes** button or **Help** button opens a panel on the right side of the Product Designer window. In some cases, when using Google Chrome, the panels open, then immediately close, then open again. This "bouncing" behavior has been observed when certain Chrome extensions are active, and appears to be related to certain mail notification and ad blocking extensions.

Workaround – Disable browser extensions one at a time until the "bouncing" window behavior stops, or use a different browser.

Cannot edit a Series audit schedule (BCC-1704)

Issue – Two types of checking audits are in common use: *single* checking audits and *series* checking audits. The base configuration does not support either type of checking audit. However, you can configure checking audits through configuration, which is made easier if you use the PolicyCenter Checking Audits Accelerator.

Product Designer provides the user interface that enables you to edit single checking audits when they have been enabled through customer configuration. However, it does not provide a user interface to edit series checking audits.

Workaround – Follow the instructions in the *PolicyCenter Checking Audits - Solution Guide* to manually create and edit an appropriate XML file with the required checking audit series patterns.

Cannot use Product Designer to edit an extended object with no base object (BCC-1759)

Issue – If you attempt to edit a part of the product model that is based on an entity extension file (`.etx`), and that extension has no base entity file (`.eti`), Product Designer displays errors and refuses to display the contents of the file.

Although Guidewire best practices recommend that you never create an entity file extension (`.etx` file) when no base entity (`.eti` file) exists, Guidewire Studio did not enforce this recommendation. However, the current version of Product Designer does not allow the use of an entity extension without a corresponding entity base file. If such an entity extension exists, Product Designer displays a **NullPointerException** when you attempt to access the affected objects. You then can use the Back command in the browser to view other pages, but you cannot configure the affected objects.

Workaround – Locate all `.etx` files that do not have corresponding `.eti` files, and rename them with an `.eti` extension.

PolicyCenter does not start in single currency display mode if the product model is configured to use multiple currencies (BCC-1800)

Issue – Product Designer allows users to configure multicurrency features without regard for the setting of the `MultiCurrencyDisplayMode` in the PolicyCenter `config.sys` file. If you configure multiple currencies in a product model object, and the value of the `MultiCurrencyDisplayMode` property is `SINGLE`, PolicyCenter does not start.

For example, if you add clauses that specify multiple coverage currencies and the PolicyCenter `MultiCurrencyDisplayMode` is set to `SINGLE`, the server fails to start and displays messages similar to the following:

```
ERROR ... PolicyLinePattern : ... [field : AvailableCoverageCurrencies] : - Multiple Coverage
Currencies defined for product not allowed in single Currency mode
ERROR ... [ExclusionPattern] : ... [DirectCovTermPattern] : ... [LimitsSet] : - The pattern specifies
limits for different currencies in SingleCurrencyMode
```

Workaround – If you intend to deploy a product model on PolicyCenter servers that are not configured for multiple currency display mode, do not configure multiple currencies in any part of the product model. However, you can test multicurrency products on a staging server configured for multicurrency while the production system is configured for a single currency.

An exception may be generated when you run 8.0.3 after a previous version of Product Designer (BCC-2154)

Issue – An `IncompatibleRemoteServiceException` may appear if you run Product Designer 8.0.3 after running a previous version of Product Designer.

Workaround – Refresh the browser cache by selecting `Ctrl+F5`.

Cannot edit a row in the question choices list (BCC-2188)

Issue – If you use Google Chrome you cannot position the mouse to select a row in the **Question Choices** list.

Workaround – Use the keyboard to select the row. You can then edit the text.

System table cells display public IDs when referencing another system table (BCC-2242)

Issue – If the cells in a particular column of a system table are referencing values from another system table, only the public IDs of those values are shown, not the values, when the system table is edited in Product Designer. In addition, users cannot search for values in these system table cells.

Workaround – View the XML directly for the system table being referenced to determine the public IDs associated with the values that need to be added, edited or deleted. You can then use those public IDs directly.

Multiple changes to offerings are not committed by Product Designer in the intended order (BCC-2392)

Issue – When uncommitted changes are made to an **Offering Page** and then the **Offering Selections Page**, the order in which Product Designer actually commits those changes is not guaranteed. This could mean that the committed changes do not reflect the user's original intentions in Product Designer.

Workaround – Validation has been added to Product Designer. After a change has been made in an **Offering Page**, it will not be possible to add or duplicate new offerings or edit any existing offerings until the original change has been committed.

Availability changes are lost when Product Designer is restarted (BCC-2472)

Issue – When you create an object and then make changes to the first availability row, the availability changes are lost when you restart Product Designer. This issue affects all objects in Product Designer.

Workaround – Commit your changes before you restart Product Designer.

Studio/Platform Known Issues

Renaming method or property throws ParseResultsException (PL-16633)

Issue – If you rename a property or a method or you change a method signature, and a workflow references that property or method in a Gosu field, PolicyCenter throws `ParseResultsException`. This is the intended behavior.

Workaround – Restart the workflow engine. To do so:

1. Log in to PolicyCenter using an administrative account.
2. Access **Internal Tools**→**Reload**.
3. Click **Reload Workflow Engine**.

Long text in table cells can add white space to the right of the page (PL-28288)

Issue – In an editable list view, extremely long text entered in a single cell can cause additional white space on the right side of the page. Long text is text that occupies approximately the entire width of the screen. This issue occurs primarily in Chrome.

Workaround – If you expect users to enter large amounts of text into the cells of a column, configure the column to support text wrapping.

Administrative command-line tools cannot refresh WSDL (PL-29021)

Issue – Administrative command-line tools rely on web service implementation classes such as `MaintenanceToolsAPI.gs`. If you change the web service implementation classes, administrative tools might fail because the tool's WSDL does not match the server WSDL. Some changes do not affect the WSDL. For example, adding a `@WsiPermission annotation`.

Workaround – Do the following:

1. From the `PolicyCenter/bin` directory, at a command prompt type the command:

    ```
    gwpc regen-soap-api
    ```

2. In Windows Explorer, copy the WSDL files from the location:

    ```
    PolicyCenter/soap-api/wsi/wsdl
    ```

    to:

    ```
    PolicyCenter/admin/res/wsi
    ```

No entity changed event fired through a one-to-one entity relationship (PL-26224)

Issue – The existence of a one-to-one entity relationship between two entities prevents an entity change event from being fired, even if it is supposed to. For example, in ContactManager there is a foreign key from `ABContact` to `Address` for the primary address, and a one-to-one relationship from `Address` to `ABContact`. Because of the one-to-one link, no event changed event is fired for any change in the primary address.

More generally, if entity E1 has a one-to-one relationship to entity E2, no entity changed event is fired for E2 if a property on E1 changes.

Workaround – You might be able to take advantage of a secondary change caused by the initial change, with the secondary change triggering the event to be fired instead. In the previous ContactManager example, making sure that changes to all properties in `Address` cause a `History` record to be generated ensures that the entity changed event will be fired.

Free-text Search indexes cannot be upgraded from 7.0 to 8.0 (PL-29062)

Issue – To upgrade free-text search in PolicyCenter from 7.0 to 8.0, you install the newer version of the Guidewire Solr Extension in 8.0 on top of the version from 7.0. Because the PolicyCenter data model also changes, the free-text search indexes from 7.0 become obsolete. Guidewire does not support migrating free-text search indexes from 7.0 to 8.0.

Workaround – For Oracle, upgrade the Guidewire Solr Extension by following instructions in the *Upgrade Guide*. Then, run the new free-text batch load command to rebuild the indexes based on the new data model, after you complete your database upgrade. For SQL Server, Guidewire is aware of potential performance problems running the free-text batch load command after you complete your database upgrade.

Chrome browser cannot display product documentation in HTML format (DOC-7251)

Issue – If you use the Google Chrome browser, you can view the HTML Guidewire product documentation only if it is served by an HTTP server using the `http://` protocol. The Chrome browser cannot load HTML product documentation from your local disk by using the `file://` protocol.

Workaround – Use a different browser, such as Microsoft Internet Explorer. If you use the HTML documentation only to access the PDF files, you can open them directly in the `pdf` subdirectory of the `doc` directory.

New mechanism for reloading Gosu classes (DOC-8218)

Issue – In past releases, you could modify your PCF files and Gosu classes in Studio, and then reload the changes into the running server by pressing `Alt`+`Shift`+`L` in the application user interface. This shortcut no longer loads Gosu classes.

Workaround – To have the server reload your Gosu classes, in Studio, click **Build**→**Make Project**. When Studio is finished compiling your project, the changes will be loaded. You can also restart your server to load the Gosu classes.

Alignment Property Does Not Work for Inputs in a DetailView (PL-29429)

Issue – Inputs in a detail view do not align to the right or center. They always align to the left, regardless of the value of the `align` property.

Workaround – If you need to right align a series of items such as a set of monetary values, put them into a list view instead of a detail view. You can then set the alignment as needed.

JMS inbound integration implementations must be OSGi or Java plugins (PL-32054)

Issue – Writing JMS inbound integration handlers requires casting an object to the type `javax.jms.Message`. In Gosu, for some app servers this fails with the message: "Caused by: gw.lang.parser.exceptions.ErrantGosuClassException: GosuClass test.GosuJMSMessageReplyHandler has errors, and cannot be used at runtime."

Workaround – Implement JMS handler code for inbound integration in Java. Guidewire recommends that for inbound integration Java code, implement your code as an OSGi plugin, not a standard (non-OSGi) Java plugin.

DateTime widget does not auto-fill the time when the date contains leading zeros (PL-32116)

Issue – The `DateTime` widget, a variation of the `DateInput` widget, configures both `dateFormat` and `timeFormat` for display. If you enter just a date into this field without entering a time, the application is supposed to auto-fill the time to 12:00 a.m. However, if the date specified contains leading zeros in both the month and day, such as 01/01/2014 for January 1, 2014, the time field is not auto-filled.

Workaround – A few workaround options are available for this issue:
- If users are not expected to enter the time, configure the widget to display only the date.
- Configure the display to show two separate date inputs, one for date and one for time.
- Inform users to not use leading zeros when entering the date. For example, 1/1/2014. In that case, time auto-fill works correctly.
- Instruct users to manually supply a time along with the date.

chapter 43

Guidewire PolicyCenter 8.0.2 Release Notes

Release 8.0.2

Release Notes Update

IMPORTANT These release notes replace the release notes that were included with the official product release. Please disregard the earlier version of the release notes.

Overview

These release notes contain the following:
- "Release Information" on page 391
- "Critical Action Required" on page 392
- "Installing This Release" on page 392
- "Support" on page 393
- "Major Issues and Changes" on page 393
- "Improvements and Resolved Issues" on page 394
- "Known Issues and Limitations" on page 403

Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

IMPORTANT If you skipped one or more upgrade releases to PolicyCenter, be sure to read the release notes for those releases to learn about changes and fixed issues.

Version Number

This release of Guidewire PolicyCenter is 8.0.2.
- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 7.0.1 or later. The latest maintenance release is preferred.

If you are implementing multiple settlement currencies, BillingCenter 8.0.2 or later is preferred.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 7.0.0 or later. The latest maintenance release is preferred.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 7.0.7, 8.0.1, or later. The latest maintenance release is preferred.

Critical Action Required

An issue that potentially causes data corruption was discovered in the quote purging batch processes. This issue can occur whether these processes have been intentionally enabled or not. You must explicitly disable the quote purging batch processes. For guidance and instructions, see Knowledge Base article 2811, `Potential for data corruption when using Purge Batch Processes`, on the Guidewire Resource Portal.

This issue will be addressed in a future release.

Installing This Release

Refer to the following:
- The *PolicyCenter Installation Guide* for general installation information
- Prior PolicyCenter release notes for any versions that you have skipped

Upgrade Information

Incorrect warning message when running Configuration Upgrade Tool (PL-28723)

When upgrading from one PolicyCenter 8 release to another PolicyCenter 8 release, running the Configuration Upgrade Tool may produce following warning message:

```
WARN cannot find Emerald base configuration zip, this could indicate a problem...
```

If the `PolicyCenter/modules/base.zip` file does exist, then you can safely ignore this warning.

You can now nest block comments in Gosu code (PL-28663)

Block (multiline) comments in Gosu code can now be placed within other block comments. This is often useful for commenting out blocks of code that already have existing comments within them. For example:

```
/*
  if(x) {
    print("x is true") // success
  } else {
    /* failure */
      print("x is not true")
  }
*/
```

The ability to nest block comments also allows you to comment out methods that contain Javadoc, which is written within comment blocks.

This change may have an unexpected impact on existing code that contains comment symbols within comments. In previous releases, those symbols were ignored. With this release, they may now be interpreted as new comments. For example, consider the following function:

```
1  function test(){
2      /*
3          /*
4      */
5      print("hello")
6  }
```

In previous releases, line 3 was ignored because it appears within a comment block. It is now interpreted as the opening of an additional comment block, nested within the outer comment. However, because the nested comment block has no close, the above code now results in an error such as "unclosed comment".

As another example, lines such as the following may be used to visually separate methods or sections of code:

```
1  /*
2     //*****************************************************
3  */
```

In previous releases, line 2 was ignored because it appears within a comment block. Now the character sequence /* in line 2 is interpreted as the opening of an additional comment block, one that has no close. The single-line comment sequence // is ignored within a block comment.

To avoid this issue, make sure that the body of a code comment does not contain the character sequence /* unless there is also a corresponding */ character sequence.

Product Designer Requires Committing All Change Lists Before Upgrading PolicyCenter

Before upgrading PolicyCenter to this release, commit changes in all open Product Designer change lists. The upgrade process discards uncommitted changes.

Support

For assistance, visit the Guidewire Community.

Guidewire Customers

- https://community.guidewire.com

Guidewire Partners

- https://partner.guidewire.com

Major Issues and Changes

This section contains major issues and changes that may affect your installation. For information on new features and major changes, see the topic "New and Changed in PolicyCenter 8.0.2" in the PolicyCenter New and Changed Guide.

Base PCF File Changes

The link below requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 8.0.1 to 8.0.2

To view a report of the changes to the base PCF files, *click here*.

Base Resource Changes

The link below requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 8.0.1 to 8.0.2

To view a report of the changes to the base resources, *click here*.

Changes in This Release Provided in Upgrade Diff Report

Guidewire provides a report detailing certain differences between the current release and your prior release. This report describes changes in display keys, entities, typelists, and the Gosu API. To obtain your custom Upgrade Diff Report, visit the Guidewire Community.

Improvements and Resolved Issues

This topic describes improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This list is not intended to be comprehensive.

PolicyCenter improvements and resolved issues

ID	Description
Account	
PC-23127	Made changes to AccountPlugin.gs so that you can configure the validation logic that occurs before merging personal accounts with company accounts.
Data Model	
PC-23391	With this fix, `TerritoryCode` creation and updates can be overridden per `PolicyLine` in `XXPolicyLineMethods.gs`.
Internationalization	
PC-17829	For languages with grammatical gender distinctions, added display keys for feminine **New Contact**, **New Company**, and **New Person**.
PC-21516	**User Language**, **Regional Format**, **Default Country**, and **Default Phone Region** are now editable in the **User Profile Admin** screen. Previously, only **User Language** was available.
PC-22620	Fixed an issue which caused the English word **Unassigned** to appear on the **Notes** screen in localized releases of PolicyCenter.
Integration	
PC-22595	For integration with a billing system, added `ConversionOnRenewalPlugin` to handle exceptions while performing Conversion on Renewal.
PC-22649	Fixed an issue where `IssuePolicyInfoEnhancement` serialized the `Currency` as `period.PreferredSettlementCurrency.toString()` - this was a localized value. The code now serializes the Currency with `PreferredSettlementCurrency#Code`.
PC-22654	When entering screen to view or edit producer codes on an account, PolicyCenter synchronizes with its billing system to refresh the commission plan identifiers for the commission plans on the current producer code. If the billing system cannot be reached, PolicyCenter uses the values it currently knows about. This is an asynchronous call, so if the billing system cannot be reached, you can include a short wait before contacting the billing system again.
PC-23058	The Commission Plan input field in single currency mode is now visible when producer code already exists. The field is not editable.
PC-23162	Fixed an issue in the PolicyCenter and ContactManager integration which resulted in `CEDEX` field being set to `false` in PolicyCenter instead of `null`, the value from ContactManager. If you are upgrading from PolicyCenter 8.0.0 or 8.0.1, run an SQL update statement that synchronizes the `CEDEX` field in PolicyCenter to the value in ContactManager.

ID	Description
	For more information, see Knowledge Base article 2796, *SQL update statement required for PC-23659 / PC-23162 when upgrade from PolicyCenter 8.0.0 or 8.0.1*, on the Guidewire Resource Portal.
PC-23172	Previous versions of the default PolicyCenter-BillingCenter integration may have created unneeded rows in the `pc_paymentplansummary` table. One of the upgrade triggers in PolicyCenter 8.0.2 should remove these unnecessary rows.
PC-23305	The `getProducerInfo` and `getProducerCodeInfo` methods were added to `BillingAPI` to support update from BillingCenter of `Producer AgencyBillPlan` value and `ProducerCode CommissionPlan` value.

Jobs/Policy Transactions

ID	Description
PC-17871	This issue was fixed in PolicyCenter 8.0.1. Further changes have been made for this release.
	A product becomes unavailable for a particular producer code when there is a bound submission of that product type associated with a different producer code. This is enforced by filtering the product selection on the **New Submission** screen.
	With this fix, if you change the Producer Code on a Submission or Issuance job, PolicyCenter now raises an underwriting issue to alert you to the change, but you can bind the policy. Previously, you could not bind the policy.
PC-22958	Fixed an issue where multiple renewal workflows were created for a single renewal job.
PC-23181	Fixed issue where if you had a canceled future term and then reinstated the current term, you could not renew the reinstated term.
PC-23216	Removed restrictions that prevented changes to Organization and Producer Code in policy transactions such as certain renewals and submissions. You can now change organization or producer code. The producer codes refer to either the producer of record or producer of service, as appropriate. If you create a new Submission with a different Organization or Producer Code from what was previously used on the account, PolicyCenter may raise an Underwriting Issue.
PC-23576	In **Side-by-Side Quoting** screen, fixed layout of multiple Coverage Terms and problems with checkboxes.
PC-23943	Fixed an issue where in several circumstances the expiration date of a policy would not update correctly. Those circumstances included, but were not limited to:
	• When the initial policy term was a short term and following terms were annual or half yearly
	• Changing the effective date on a policy renewal (resulting in a lapse)
	• Changing the term type of the policy

Lines of Business

ID	Description
PC-16051	Fixed an issue with importing buildings in Commercial Property. Previously, buildings on new locations that were imported did appear on the screen comparing previous and imported information for the location.
PC-16438	An Inland Marine submission now quotes successfully when a policy hold applies to that submission. Previously, the policy hold was checking the locations on the line, but now it is checking the locations on the period.
PC-23040	Fixed an issue in Commercial Property with importing building and locations. Previously, coverages that had not been modified were shown as imported in the **Comparing Previous and Imported** tree in the **Import From Spreadsheet** screen.
PC-23117	Fixed an issue in Commercial Property which prevented issuing a policy due to inadequate facultative reinsurance. As part of initializing the approval screen for an underwriting issue, PolicyCenter now puts the value needed to approve the underwriting issue is put into the approval.
PC-23573	Fixed a null pointed exception when exporting or importing buildings or locations for Commercial Property. This issue occurred in a system with a language pack.

ID	Description
PC-23653	Changes to the schedule interface enable you to create complex schedule data models with one-to-many relationships from a schedule item to many coverages. Previously, schedule data models were limited to a one-to-one relationship. As a result of this change, a new property called `ScheduledItemMultiPatterns` was added and the existing `ScheduledItemPattern` was deprecated. The deprecated property is fully backward-compatible, but will eventually be removed. Guidewire recommends that you convert your code to use the new property. If you use the new property, modify your code to handle the possibility that multiple patterns can be returned.

Product Model

ID	Description
PC-15808	Fixed an issue that caused an exception in Studio when calling `ProductModelAPI.getAvailableQuestions`.
PC-23286	Added a `removeUnusedAnswers` method to the `QuoteProcess` class. A required question can now have a null answer if and only if the question is invisible to the user at the time of Quote. This can happen when one question is dependent upon another. This is a change. Sometimes a dependent question is visible to the user, and sometimes it is invisible. Previously, the answers (default, null, or filled in by the user) to invisible dependent questions (required or not) were being committed with the other answers. Now, the answers to invisible questions are nulled during Quote. This fixes the previously Known Issue PC-10645.
PC-23681	If you create an audit schedule pattern of type Premium Report, you must now specify **Payment Plan Code** and **Reporting Default Deposit %** in Product Designer. You must also specify `<patternName>.Name` in `productmodel.display.properties`. If you do not specify these, product model validation throws errors at server startup. These new checks prevent you from creating a premium report audit schedule pattern that has missing critical information.

Rating

ID	Description
PC-16985	You can now expand the Rating Worksheet list view columns to can view the worksheet data.
PC-17907	Fixed an issue in impact testing, whereby if you searched for policies then changed the criteria, you received a message that the new criteria were used for the test when in fact the old criteria were being used. Now you are not allowed to proceed with the changed criteria until you search with the new criteria.
PC-22302	Added a page size control to `ImpactTestingSearchCriteria.gs` which affects the `ImpactTestingTestCase.populatePeriods` method. You can adjust the page size to optimize performance. Default is no paging.
PC-22533	Fixed an issue in the `SysTableRatingPlugin` where the plugin did not handle rating overrides for Commercial Property.
PC-22927	Previously, compressed rating worksheet data did not have a file extension. Now the filename includes the gz extension to make it clear that it is a gzipped file.
PC-23355	Added a **Continue with Interrupted Baseline Creation** button to the **Create Baseline** screen and a **Continue with Interrupted Test Periods Quote** to the **Testing Period** screen. These buttons enable you to restart a canceled job.
PC-23448	Fixed an issue where changing a Schedule modifier value allowed user to Quote without providing a Justification for the modifier value.
PC-23670	Fixed an issue which caused a `NullPointerException` for the default value `null` for a rate book in a rate routine during validation.
PC-23671	Fixed issue where mismatched data type values resulted in `ConversionException: Unparseable number: "null"` and you had to remove the row to continue.
PC-23763	Fixed a issue where assigning a typekey value into a writable parameter created an unreadable worksheet.
PC-23800	In PolicyCenter 8.0.1, the `AbstractRatingEngineBase` class calculated term end dates slightly differently than the `PolicyTermPlugin`. In this release, the `AbstractRatingEngineBase` uses the `PolicyTermPlugin` to calculate term end dates, thus ensuring consistency in term length for rating.
PC-23810	PolicyCenter now reports a validation warning instead of an error when a rate book includes a rate routine that references a rate table that does not exist.

Security

ID	Description
PC-21619	Fixed an issue with permission checking on the account **Work Orders** screen where links to some policy transactions were clickable but should not have been.
PC-23649	Added a new permission that allows users to assign activities from a queue. In this release, this permission has not been assigned to any roles. To allow users to assign activities from a queue, add this new permission, `actqueueassign` or `SystemPermissionType.TC_ACTQUEUEASSIGN`, to a role.
User Interface	
PC-22916	The page configuration file, `GlobalAddressInputSet`, can display addresses in two different ways: the concatenated formatted address and the individual address items. The former should be visible when read-only, and the latter when the user enters data. Properties on the address Owner, for example `addressOwner.ShowAddressSummary`, control the display. These properties were not being set in all cases. So in places, such as **Contact** screen and the **Location** screen on the **Account Summary**, the address appeared more than once.
Upgrade	
PC-22962	The `gw.api.web.util.TransactionUtil` method called `runAtomically` is now deprecated because it allowed intermediate bundle commits. If you used this, instead explicitly call the bundle `commit()` method only as needed. The method is now unused in PolicyCenter code. As part of this change, PolicyCenter refactored some entity data changes from intermediate bundle commits. All related job changes and new event messages commit atomically.

Product Designer improvements and resolved issues

ID	Description
BCC-1794	Fixed an issue which prevented Product Designer from running on WebLogic in PolicyCenter in 8.0.0 and 8.0.1.
BCC-2095	Fixed an issue which caused the language setting in one Product Designer (PD) session to affect the display keys in another session. Example: 1. First PD session starts with the default language `ja_JP`. 2. First PD session adds a workspace, creates a change list, creates `user1`. 3. First PD session switches language to `en_US`. 4. Second PD session (with `user1`) starts with default language `ja_JP`. 5. Second PD session creates change list. 6. First PD session clicks on a new page in the tree. This session incorrectly shows `ja_JP` display keys for some fields.
BCC-2116 BCC-2114	Fixed issues where edits to offerings were lost on commit.
BCC-2117	Fixed an issue which causes an exception to occur if the correct answer was removed for a question of type **Choice Radio**.
BCC-2119	Fixed an issue which caused an exception to occur when the label was removed for a question which was as **Dependent On** for another question.
BCC-2130	A validation message did not appear when no offering was added for a new product that has **Offering Required** selected. This has been fixed and a validation message now appears.
BCC-2176	Fixed an issue that allowed a local user to search for a workspace folder on a remote Product Designer server. In the Create Workspace dialog, the file navigation selection option only appears for a local instance of Product Designer.
BCC-2207	Fixed an issue that caused an exception to be generated when Product Designer is re-started after the workspace folder is renamed.
BCC-2209	Fixed an issue that allowed you to set an active change list for a deleted or renamed workspace folder.

ID	Description
BCC-2240	Fixed an issue which caused Product Designer to show blank text when a **Question** label or the associated help text was removed.

Platform improvements and resolved issues

The following are the primary improvements and issues corrected in this release:

ID	Description
Activities, Assignment	
PL-29201	A new permission has been added to allow users to assign activities from a queue. This permission is not assigned to any roles by default. To allow users to assign activities from a queue, add the new permission `actqueueassign` or `SystemPermissionType.TC_ACTQUEUEASSIGN` to the necessary role.
Archiving	
PL-28043	Fixed an archiving bug that caused improper handling of a data model change during restoration of an entity from the archive. If the data model was changed so that a property was moved from a subtype to its supertype, the related data on the property could have been lost from the archive document during restoration.
PL-28098	Improved archiving to support links to objects outside the current graph of archived objects. In your `IDatamodelUpgrade` plugin implementation, every `IArchivedEntitySet` object has two new methods. The `addCrossGraphRelationship` method adds a cross-graph relationship between a `publicID` field and a link field. The `removeCrossGraphRelationship` method removes the relationship.
PL-29160	Improved archiving graph validation error message related to overlap tables to simplify error messages to only include entities that are the root of issues and also include the involved links in the error message.
Build Infrastructure	
PL-24111	When generating a WAR file using the build-war command and passing `-Dconfig.war.dictionary=true`, the data dictionary was not being generated. This issue is resolved.
Clustering	
PL-29135	In the event of a serious network failure (such as a network switch failure) when a cluster is split into many parts, JGroups might have problems merging these parts into a single cluster after the network is restored. To solve this potential problem Guidewire implemented a JGroups watchdog timer that resets the JGroups channel if it thinks that the node is running outside of the cluster for a long time. The JGroups watchdog uses the following algorithm: • Coordinator of the cluster sends periodic heartbeat messages. Each message contains the list of the cluster members from the coordinator's point of view. • The watchdog resets the channel if the node does not receive messages from its coordinator for a long time. Note that an incorrect heartbeat (one that does not include the node in the list of members) is equal to a missing heartbeat. This feature can be configured by adjusting two new parameters in `config.xml`: • `JGroupsWatchdogHeartbeatIntervalSecs` - Coordinator node sends a heartbeat each interval. Other nodes reset JGroups channel if they do not receive the heartbeat from their coordinator for several intervals. The default value is 30 seconds. • `JGroupsWatchdogMissedHeartbeatsBeforeReset` - The number of missed heartbeat messages after which a node resets its JGroups channel. The default is 10.
PL-29240	Updated the default value for `ClusterProtocolStack` to a configuration that is more reliable in case of a major network failure. FD protocol was replaced with `FD_ALL`, MERGE2 was replaced with `MERGE3`.
Command Line Tools	
PL-27608	Removed extraneous files from the output of the command `zip-changed-config`. The list used is in `com.guidewire.tools.upgrade2.merge.ConfigMergeList#EXCLUDE_DIRS`.

ID	Description
	Also added new option -e, which lets you specify a semicolon-separated list of directories to exclude.

Configuration Upgrade

ID	Description
PL-19432	The configuration upgrade tool now accepts "Y" or "N" keyboard input for dialog boxes.
PL-28639	The configuration upgrade tool now formats metadata extension files in a more consistent way.

Consistency Checker

ID	Description
PL-26560	Added a progress bar to the consistency check page.
PL-26841	Fixed an issue where a check constraint error might not show the description of the check that failed.
PL-28361	If one or more consistency checks result in an SQL failure, then the run will be flagged to allow a rerun of just the checks that failed, in a single thread.
PL-28579	Consistency checks are now executed by the worker threads individually instead of being grouped by table and consistency check type.
PL-28819	If an application server crashes while consistency checks are being run, when the server restarts, the worker threads will restart, but some of the consistency checks won't be run.

Consistency Checker, Other - Database Pod

ID	Description
PL-28612	Fixed an issue where a work queue could wait a long time if the server is restarted when an item is being processed.

Data Distribution

ID	Description
PL-28620	Fixed an issue that would cause the data distribution process to fail.

Database Configuration, Database Instrumentation

ID	Description
PL-28478	Database debug logging including SQL statements has been enabled for JNDI configurations.

Database Configuration, Database Support - General, Database Support - Oracle, Database Upgrade

ID	Description
PL-28601	Resolved an issue in which the database statistics configuration in config.xml was not honored during upgrade. Statistics for tables that were specified to keep their statistics were being erroneously deleted.

Database Configuration

ID	Description
PL-28660	The validationQuery attribute has been removed from the database configuration of the DBCP connection pool.

Database Configuration, Database Support - Oracle

ID	Description
PL-28730	Updated the **Database Parameters/Guidewire Database Config Statistics Settings** page to show only the configured items.

Database Instrumentation

ID	Description
PL-28236	The **Server Tools Database Statistics** page has been simplified. In addition, the code that compared the actual row counts with the calculated row counts, and used that information to calculate the statistics, has been eliminated. Statistics can be run unconditionally on all or a set of tables, or you can run "incremental" statistics, which uses the incrementalupdatethresholdpercent attribute of the databasestatistics configuration element, which defaults to 10 percent, to decide which statistics to run. Now the database statistics report reports all the update statistics statements that would be executed.
PL-28294	Fixed an issue where system_tools and maintenance_tools could be run from the command line without proper permissions.

Database Support - General

ID	Description
PL-28655	Improved the performance of proximity searches.

Database Support - Oracle

ID	Description
PL-29300	The enable-all setting of the `ora-parallel-dml` attribute on the upgrade element no longer forces parallel execution in Oracle for an `InsertSelectBuilder` that is used outside of upgrade. The `BeforeUpgradeInsertSelectBuilder` is still forced to use parallel execution if the `ora-parallel-dml` attribute is set to `enable-all`.

Database Support - SQL Server, Database Upgrade

ID	Description
PL-28779	Fixed an issue that prevented upgrade to a PolicyCenter 8.0 maintenance release when the database type is SQL Server.

Database Support - SQL Server

ID	Description
PL-28885	Fixed an issue which allowed users to create the PolicyCenter schema on a SQL Server system database.

Database Upgrade

ID	Description
PL-28302	The Upgrade Info report is now generated even when the upgrade fails, if the upgrade is run in development mode. The report is generated in the default temporary-file directory specified by the system property `java.io.tmpdir`. On UNIX systems the default value of this property is typically `/tmp` or `/var/tmp`. On Microsoft Windows systems it is typically `c:\temp`.
PL-28447	If data model files were changed but the extension.properties version number was not updated, PolicyCenter would report an error when starting the server. This error did not include the file name of the extension file that included the data model change. The error has been updated to include the extension file name.
PL-28463	Resolved an issue that caused a `ClassCastException` when trying to delete a column using the method `dropColumns` inside an `AfterUpgradeVersionTrigger`.
PL-28572	Improved the performance of the upgrader when typecodes have been removed.
PL-28613	Fixed the ordering of typelist IDs, which could create a race condition during schema verification while upgrading the database.
PL-28773	We now support database upgrade version trigger configuration options to cause select statements, particularly in `VersionChecks`, to use an Oracle hint to cause parallel execution in query (SELECT) statements.
PL-29200	If the upgrade configuration attribute `deferCreateArchiveIndexes` is not explicitly specified in the `database-config.xml` file, then its default value has been changed to `false`. See the *Upgrade Guide* topic "Deferring Creation of Archive Indexes" for information about this feature.

Entities/Metadata

ID	Description
PL-29932	Added the following attributes to the `MonetaryAmount` entity: `createhistogram`, `createAmountHistogram`, `createCurrencyHistogram`.

Gosu

ID	Description
PL-28349	Previous versions of Gosu reported a warning on ambiguous method calls. Ambiguous method calls can hide a logical bug in your code. Previously, the Gosu compiler selected the best matching method to remove ambiguity. This release changes behavior in two ways (1) Ambiguous calls are now an error instead of a warning. (2) Studio now has a code inspection to identify and optionally fix any ambiguous code to previous Studio behavior. This inspection is disabled by default. To find and fix potential logical errors, Guidewire recommends that you run the inspection and carefully individually analyze every ambiguous call before applying any proposed fix.
PL-28663	You can now nest comments in Gosu code.

Gosu, IntelliJ IDE - Gosu Editor

ID	Description
PL-28777	Fixed an issue where Studio would display invalid "implicit conversion to integer" warnings on PCF code that used Gosu reflection.

IntelliJ IDE - Customer Build

ID	Description
PL-29861	The version number reported by Guidewire Studio now reflects the particular Studio version rather than the version of its companion Guidewire application. These numbers may be different.

IntelliJ IDE - Entity Editor

ID	Description
PL-28783	Fixed an issue where changes to a delegate containing `MonetaryAmount` would not be saved correctly.
PL-29211	Fixed an error that would occur when trying to open several typelist editors at the same time.

IntelliJ IDE - Entity Editor, IntelliJ IDE - Typelist Editor

PL-28999	Removed the file name suffix option from the entity extension and typelist extension dialog boxes, since this is not a common operation, and removing the option simplifies the creation process. To restore this option, click **File→Settings**, then navigate to the **Guidewire Studio→Metadata Editor** page, and then set **Show filename suffix on new extension dialog**.

IntelliJ IDE - Entity Editor, IntelliJ IDE - Typelist Editor

PL-29592	Fixed an error that would occur if you tried to create an entity or typelist extension when one already exists.

IntelliJ IDE - Formatting

PL-29310	When viewing PCFs, entities, and typelists as XML in Studio, unicode characters now display according to their language native symbol, rather than as a unicode escape sequence.

IntelliJ IDE - Other

PL-27965	The **Compare to Base** and **Show Diff** windows can now be resized and maximized.
PL-29172	Guidewire Studio is bundled with IntelliJ IDEA Community Edition. If you would like to run Studio with your own installation of IntelliJ IDEA Ultimate Edition, you can now do so. Only version 12.1.7 of IntelliJ IDEA Ultimate Edition is supported. To enable this, in the Guidewire product root directory, create a text file named `studio.ultimate` that contains the path to the IntelliJ IDEA Ultimate installation directory.

IntelliJ IDE - Typelist Editor

PL-28289	Fixed a problem when, after installing a language pack, typelists within the language pack were not recognized as valid metadata files.

Internal/Server Tools Pages, Work Queues

PL-28725	Implemented the Work Queue management bean. The **Management Beans** page now shows a management bean for each work queue running locally.

Localization

PL-27731	The maximum length of a phone number extension can now be set in the Plugin Registry editor. The default length is 4. To set this value: 1. Open Guidewire Studio and navigate in the **Project** window to **configuration→config→Plugins→Registry→IPhoneNormalizerPlugin.gwp**. 2. Add a new parameter by clicking the + button to the right of the **Parameters** pane. 3. For the **Name**, enter `extensionLength`. 4. For the **Value**, enter a number representing the length of the phone number extension.
PL-29815	There was an issue with out of memory exceptions with large data sets when running the phone normalizer upgrade batch process writer. This issue has been fixed. The writer now creates batch process work items in chunks, reducing total heap space required.

Other - Database

PL-28631	Fixed an issue where `SystemToolsAPI.getUpdateStatsState` would return the wrong state.

Other - Developer IDE

PL-27186	Studio now shows warnings for improper usages of internal types when the usage is in an `extends` or `implements` clause.
PL-28718	Improvements in IntelliJ with OSGi Editor when you upgrade (or move) a Guidewire product to a new disk location. In IntelliJ with OSGi Editor, open your module settings. Click **OSGi Bundle Facet**. To the right of **Guidewire product directory** text field, click the **Change** button and set the new disk path. The tool updates IDE library dependencies and build.properties.

Other - Integration

ID	Description
PL-29002	This release adds a new servlet utility class `gw.servlet.ServletUtils`, which includes methods to get the signed in User object. Use any of three authentication types: 1) the session token from a logged-in user linking from a PCF page; 2) HTTP Basic authentication headers; 3) or name/password pair from custom headers. You can use more than one API in your servlet, for example check the session token, and if it is not available use HTTP Basic authentication. See the Gosu Reference Guide book in the Servlets chapter for details.

Other - Persistence

ID	Description
PL-28762	Updated the exception for bean overlaps to include more information about the bean including the type of the bean and the fixed id and date range.

Plugins

ID	Description
PL-28621	Failing to start a startable plugin during server start now prevents the server from advancing to the next run level.

Queries

ID	Description
PL-27790	Resolved an issue that table aliases in subqueries were not generated with unique values.
PL-28194	With this release, the query builder APIs throw an exception whenever an effdated entity participates in a query through a join statement.

Revisioning

ID	Description
PL-27210	Fixed an incorrect error that was generated by the "Verifies foreign key reference to a subtype is to correct subtype" consistency check.
PL-28586	Fixed a bug relating to creating a new `PolicyPeriod` from a `PolicyPeriod` that had effective-dated entity instances with matching effective date and expiration date, which are also called 0-width beans. For an unlocked `PolicyPeriod`, the method `createDraftMultiVersionJobBranch` now copies these entity instances to the new `PolicyPeriod`.

Templates

ID	Description
PL-27589	The WS-I web service called `TemplateToolsAPI` changed to expand functionality to note and email templates, not just document templates: (a) The four methods `validateAllTemplates`, `validateTemplate`, `validateTemplateInLocale`, and `listTemplate` have new names `validateAllDocumentTemplates`, `validateDocumentTemplate`, `validateTemplateDocumentInLocale`, `listDocumentTemplate`. The old method names are deprecated. (b) New methods for note templates: `validateAllNoteTemplates`, `validateAllNoteTemplatesInLocale`, `validateNoteTemplate`, `validateNoteTemplateInLocale`, `listNoteTemplate` (c) New methods for email templates: `validateAllEmailTemplates`, `validateAllEmailTemplatesInLocale`, `validateEmailTemplate`, `validateEmailTemplateInLocale`, `listEmailTemplate`

Web - Configuration

ID	Description
PL-28965	In PCF files, when the LinkCell widget is used under the RowTree widget, the `id` attribute of the LinkCell widget is now required.

Web - ListViews

ID	Description
PL-25265	In the PCF widget RowTree, removed support for the `pageSize` attribute, since the widget is able to resize dynamically. The default value for this attribute is now 0, and any other value produces a validation error.
PL-26720	Added validation to RowTree to prevent using Cell types that cannot be rendered inside the RowTree. For example, CheckBoxCell cannot show check boxes inside the tree. If needed, convert existing usages to the type Cell.
PL-26900	Fixed an issue in ListViews where hidden columns that were redisplayed were not becoming editable when triggered.
PL-28169	Fixed an issue where user-hidden columns caused layout problems by disabling the ability to show/hide columns for columns when a colspan is specified.
PL-28782	Fixed an issue with the RangeValueWidget where RowTree cells were not displaying the proper localized value.

ID	Description
PL-29483	Fixed an issue where grouping a ListView displayed a generic "object" string instead of the correct value.
Web - UI/Runtime	
PL-27761	Fixed an issue where pressing Enter did not properly trigger a search or navigation.
PL-28593	Fixed an issue where the text in the InfoBar was not selectable for copying.
PL-28666	Added validation to the DetailViewPanel configuration to ensure that Input widgets are enclosed by an Input-Column.
PL-28770	Resolved an issue that MonetaryAmountCell did not handle some currency formats correctly.
PL-28854	Fixed an issue with rendering of LinkCells in the RowTree.
PL-28860	You can now localize watermarks in the user interface for date and time fields.
PL-28946	Fixed the DateInput widget for the time to reflect the user's settings for 12-hour versus 24-hour format.
PL-29052	Resolved a critical security vulnerability to non-persistent cross-site scripting attacks.
PL-29475	You can no longer select **Group By This Field** in a list view when it is in edit mode.
Web Services - WSI (New)	
PL-27467	Fixed an issue where WsiAuthenticationException was being declared in generated WSDLs, but was not actually being returned.
PL-28179	Fixed an issue where WebserviceServlet was waiting for a return to multmode once the runlevel was dropped to maintenance mode before responding to requests.
PL-28267	Web services now throw HTTP 500 for SOAP faults instead of HTTP 200.

Known Issues and Limitations

This section describes known issues with this release of Guidewire PolicyCenter.

- "PolicyCenter Known Issues" on page 403
- "Product Designer Known Issues" on page 406
- "Studio/Platform Known Issues" on page 408

> **Note:** For maintenance releases, Guidewire often defers fixing configuration issues that require merging files during the upgrade. Workarounds to many of these issues are listed in the following sections. The goal of this policy is to make upgrades as straightforward as possible.

PolicyCenter Known Issues

Account Billing screen may erroneously indicate Unsaved Work (DOC-9628)

Issue – On navigating to the **Account Billing** screen, the Unsaved Work list may become active when no work has been done on that screen.

Workaround – Guidewire is aware of this issue.

Additional information about Converting Money to MonetaryAmount (DOC-9164)

Issue – The following statement in the *Upgrade Guide*, "Converting Money to MonetaryAmount" topic is not quite complete: "If you change the definition of your extension properties to use the MonetaryAmount datatype rather than the Money data type, upgrade will correctly convert your entity extension data."

Workaround – The more complete information is: "If you change the definition of your extension properties to use the MonetaryAmount datatype rather than the Money data type, upgrade will correctly convert your entity extension data if you are coming from a single-currency instance (including any local extensions or business practices to manage currency) with the instance currency set properly. If you already manage multiple currencies, either

explicitly or implicitly (such as through special notations on the account or the like), you must write an appropriate upgrade trigger as well as examine the supplied single-currency upgrade triggers."

Rewrite jobs incorrectly calculate expiration date in some cases (PC-23988)

Issue – Rewrite jobs do not correctly calculate the end of a the new policy term correctly in certain circumstances. Specifically, if the rewrite is full term or new term (that is, a full term not a partial term), the effective date of the rewritten term is the last day of the month. If that same day of the month is not the last day of the expiration month, PolicyCenter incorrectly uses that day for the expiration date, not the last day of the month. For example:

• Policy term is annual, effective date is 2/28 and the policy ends in a leap year. Expiration date is set to 2/28, but should be 2/29.

• Policy term is semi-annual, effective date is 4/30. Expiration date is set to 10/30 but should be 10/31.

Workaround – Guidewire is aware of this issue.

Exception in PolicyCenter when adding an Account Contact in BillingCenter (PC-23873)

Issue – When BillingCenter creates a new Contact for an Account that had been created in PolicyCenter, PolicyCenter does not recognize the Contact and throws a `BadIdentifierException`.

Workaround – Guidewire is aware of this issue.

Reporting policies may not retain reporting schedule on renewal (PC-23869)

Issue – Certain policies that use reporting schedules, such a Workers' Compensation, allow you to select various reporting plans and audit schedules. A manual renewal on these policies can result in a different reporting plan and audit schedule being applied to the new term.

Workaround – When manually renewing a reporting policy, review the reporting plan and make sure the correct reporting plan is being used.

PolicyCenter allows owner and overriding-payer accounts to be the same (PC-23570)

Issue – In PolicyCenter, you can set the overriding payer on a list bill policy to the same account as that account which owns the policy. You set the overriding on the **Account→Billing** screen. Systems which receive data from PolicyCenter may report this as an error. For example, BillingCenter requires an overriding payer account to be different than the owner account.

Workaround – Add validation or other logic to PolicyCenter that ensures that, if present, an overriding-payer account is different than the owner account.

Gosu classes not intended for use (PC-23009)

Issue – Some Gosu classes are visible in Studio, but are not intended for use. You can distinguish these Gosu classes because they have no visibility annotations (neither `@Export` nor `@ReadOnly`) and they are not in a `gw` package.

Workaround – Do not use these methods in configuration. The methods are unsupported and may change or be withdrawn without notice.

Reinsurance warnings not refreshed even after saving a draft (PC-21720)

Issue – When working with reinsurance, PolicyCenter displays validation errors and warnings both at the top and in the worksheet at the bottom of the screen depending on the action that caused validation to run (typically **Save**, or leaving the screen). Depending on the user flow, one or both of these areas may display stale messages, and the messages may not be the same.

Workaround – Refresh the messages in either or both locations before taking the action again.

Error when quoting after adding new location (PC-21656)

Issue – In the Commercial Auto, General Liability, and Commercial Property lines, users sometimes receive an error message after clicking **Quote** on the **Locations** screen indicating that there are no locations even when there are. This error could potentially occur in other lines of business and on other list views, depending on how they are configured.

Workaround – Click **Next** and then **Back** to navigate away from and back to the **Locations** screen before quoting. Alternately, remove the client side validation on the number of rows.

Copy-data with an out-of-range date (PC-20279)

Issue – When working with the copy data feature, if you specify a specific date to copy data from and that date is out of range for the policy, PolicyCenter presents a warning that no applicable policy version could be found. However, the date selector still indicates the requested date, not the date that PolicyCenter will use for the copy if you choose to proceed.

Workaround – As appropriate, modify the user interface to reset the date selector or to hide the ability to select data to copy.

SQL Query performance in free-text search (PC-18794)

Issue – There are open issues related to the performance of SQL query in free-text search. This occurs the first time that you load the PolicyCenter data into the Solr index database.

Workaround – Guidewire is aware of this issue.

Create Baseline button not visible after navigating to the Create Baseline screen (PC-17809)

Issue – When you start the **Create Baseline** step, PolicyCenter kicks off a batch job to create the baseline quotes. When you click **Cancel**, the cancel request is sent to the batch job, but the current quote may take some time to stop. If you click **Cancel**, go back and return to the **Create Baseline** screen. You may see the progress bar from the prior cancellation.

Workaround – Navigate to another screen and wait about a minute before returning to the **Create Baseline** screen.

Renewal processing for renewal conversion assumes BasedOn is populated (PC-15851)

Issue – The `startNewRenewal` method in `gw.webservice.pc.pc700.job.PolicyRenwalAPI` assumes the existence of a prior term or a `basedOn` version. Unless customized, this method can cause problems in certain convert-on-renewal scenarios. This method has been deprecated.

Workaround – If you are developing convert-on-renewal flows, use the `gw.webservice.pc.pc700.job.startConversionRenewal` method.

BillingCenter integration: New invoice streams not created when issuing policy with alternate billing account on new account (PC-15309)

Issue – Assume an account has an existing invoice stream. When this account is used as an **Alt Billing Account** for new policies on new accounts, a new invoice stream is not created when the new policy's new invoice stream periodicity matches the invoice stream on the **Alt Billing Account**.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Guidewire is aware of this issue.

BillingCenter integration: Removing alternate billing account when payment instrument selected generates WsdlFault (PC-15308)

Issue – If an automatic payment instrument is selected when using an Alt Billing Account to create a new policy, PolicyCenter throws a `WsdlFault` if the Alt Billing Account gets removed.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Contact Guidewire Support for assistance with this issue.

BillingCenter integration: Automatic payment instrument is used to create invoice stream even if final selection is manual (PC-15307)

Issue – If an automatic payment instrument is setup and selected, the policy will create a new invoice stream with this payment instrument even if the user changes to Manual before issuing the policy.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Guidewire is aware of this issue.

PolicyCenter does not send renewal billing instruction to BillingCenter (PC-15082)

Issue – On receiving payment notification from BillingCenter, PolicyCenter tries to bind. If the policy is in a draft status, PolicyCenter does not bind and send the Renewal Billing Instruction to BillingCenter. The policy might be in draft status after an underwriter edited something on the policy. This is an issue with BillingCenter 3.0.1 and later.

Workaround – Guidewire is aware of this issue.

Must enter VIN when moving from quick quote multi-version or side-by-side to full application (PC-13099)

Issue – In a quick quote, you do not have to enter the VIN or other fields that are required for a full application submission. When converting a quick quote with multi-version or side-by-side quoting to full application, to resolve validation issues, you must enter the VIN for each of the versions. In a job with side-by-side quoting, this issue can occur with fields that are not base data.

Workaround – Add code that copies this type of data to the other versions.

When using parallel sessions, approvals made by underwriter are not updated for second user simultaneously viewing same work order (PC-12322)

Issue – Approvals made by an underwriter are not updated for a second user simultaneously viewing the same work order. This occurs in the following sequence of events:

• An agent processes a submission and issues arise
• The agent calls an underwriter, and remains in the policy
• The underwriter approves the issue or issues
• The agent refreshes the **Risk Analysis** screen by going to a different wizard step and returning to the **Risk Analysis** screen

After this sequence of events:

• The new approvals do not appear on the agent's screen
• However, the agent can bind or issue the policy (based on the new approvals by the underwriter)

Workaround – The agent can leave the policy and return to it. The approvals appear as expected. Alternatively, the agent can see the approvals by clicking the **Details** button when issuing the policy.

Provide a way to filter out added and removed subentities in a policy differences tree (PC-7929)

Issue – Need a way to filter out the children of an added or removed entity in a policy differences tree.

Workaround – Guidewire is aware of this issue.

Product Designer Known Issues

PolicyCenter does not start in single currency display mode if the product model is configured to use multiple currencies (BCC-1800)

Issue – Product Designer allows users to configure multicurrency features without regard for the setting of the `MultiCurrencyDisplayMode` in the PolicyCenter `config.sys` file. If you configure multiple currencies in a product model object, and the value of the `MultiCurrencyDisplayMode` property is `SINGLE`, PolicyCenter does not start.

For example, if you add clauses that specify multiple coverage currencies and the PolicyCenter `MultiCurrencyDisplayMode` is set to `SINGLE`, the server fails to start and displays messages similar to the following:

```
ERROR ... PolicyLinePattern : ... [field : AvailableCoverageCurrencies] : - Multiple Coverage
Currencies defined for product not allowed in single Currency mode
ERROR ... [ExclusionPattern] : ... [DirectCovTermPattern] : ... [LimitsSet] : - The pattern specifies
limits for different currencies in SingleCurrencyMode
```

Workaround – If you intend to deploy a product model on PolicyCenter servers that are not configured for multiple currency display mode, do not configure multiple currencies in any part of the product model. However, you can test multicurrency products on a staging server configured for multicurrency while the production system is configured for a single currency.

PolicyCenter exception when synchronizing the product model while a submission is in progress (PC-21708)

Issue – PolicyCenter throws a `ClassCastException` and stops responding if you synchronize the product model from Product Designer while a submission policy transaction is in progress. To recover from this state, you must restart the PolicyCenter server.

Workaround – Do not synchronize the product model while a submission policy transaction is in progress. Instead, navigate to a screen outside any policy transaction, such as the **Account Summary** screen, before synchronizing the product model.

Alternatively, edit `config.xml` to enable internal debug tools, and then after synchronizing the product model and prior to performing any operation in PolicyCenter, press `Alt+Shift+L` to reload PCF files.

Cannot use Product Designer to edit an extended object with no base object (BCC-1759)

Issue – If you attempt to edit a part of the product model that is based on an entity extension file (`.etx`), and that extension has no base entity file (`.eti`), Product Designer displays errors and refuses to display the contents of the file.

Although Guidewire best practices recommend that you never create an entity file extension (`.etx` file) when no base entity (`.eti` file) exists, Guidewire Studio did not enforce this recommendation. However, the current version of Product Designer does not allow the use of an entity extension without a corresponding entity base file. If such an entity extension exists, Product Designer displays a **NullPointerException** when you attempt to access the affected objects. You then can use the Back command in the browser to view other pages, but you cannot configure the affected objects.

Workaround – Locate all `.etx` files that do not have corresponding `.eti` files, and rename them with an `.eti` extension.

Cannot edit a Series audit schedule (BCC-1704)

Issue – Two types of checking audits are in common use: *single* checking audits and *series* checking audits. The base configuration does not support either type of checking audit. However, you can configure checking audits through configuration, which is made easier if you use the PolicyCenter Checking Audits Accelerator.

Product Designer provides the user interface that enables you to edit single checking audits when they have been enabled through customer configuration. However, it does not provide a user interface to edit series checking audits.

Workaround – Follow the instructions in the *PolicyCenter Checking Audits - Solution Guide* to manually create and edit an appropriate XML file with the required checking audit series patterns.

When using Google Chrome, Changes and Help panels "bounce" open and closed (BCC-926)

Issue – Clicking the **Changes** button or **Help** button opens a panel on the right side of the Product Designer window. In some cases, when using Google Chrome, the panels open, then immediately close, then open again. This "bouncing" behavior has been observed when certain Chrome extensions are active, and appears to be related to certain mail notification and ad blocking extensions.

Workaround – Disable browser extensions one at a time until the "bouncing" window behavior stops, or use a different browser.

An exception may be generated when you run 8.0.2 after 8.0.1 (BCC-2154)

Issue – An `IncompatibleRemoteServiceException` may appear if you run Product Designer 8.0.1, stop it, and then run Product Designer 8.0.2.

Workaround – Refresh the browser cache by selecting `Ctrl+F5`.

System table cells display public IDs when referencing another system table (BCC-2242)

Issue – If the cells in a particular column of a system table are referencing values from another system table, only the public IDs of those values are shown, not the values, when the system table is edited in Product Designer. In addition, users cannot search for values in these system table cells.

Workaround – View the XML directly for the system table being referenced to determine the public IDs associated with the values that need to be added, edited or deleted. You can then use those public IDs directly.

Cannot edit a row in the question choices list (BCC-2188)

Issue – If you use Google Chrome you cannot position the mouse to select a row in the **Question Choices** list.
Workaround – Use the keyboard to select the row. You can then edit the text.

Warnings appear when the product model is synchronized with PolicyCenter (BCC-2295)

Issue – Warning messages may appear in the Product Designer console when you synchronize the product model with a PolicyCenter instance.

Workaround – No action is required. Your committed product model changes are unaffected and will be deployed to the PolicyCenter instance.

Studio/Platform Known Issues

Renaming method or property throws ParseResultsException (PL-16633)

Issue – If you rename a property or a method or you change a method signature, and a workflow references that property or method in a Gosu field, PolicyCenter throws `ParseResultsException`. This is the intended behavior.
Workaround – Restart the workflow engine. To do so:
1. Log in to PolicyCenter using an administrative account.
2. Access **Internal Tools→Reload**.
3. Click **Reload Workflow Engine**.

Javadoc command does not generate index file in expected location (PL-27679)

Issue – When you run `gwXX regen-java-api`, an `index.html` file is not created in `PolicyCenter/java-api/doc`.
Workaround – The command now generates Javadoc JAR files in `PolicyCenter/java-api/doc`. To view the Javadoc, add the JAR files to the Studio project.

Long text in table cells can add white space to the right of the page (PL-28288)

Issue – In an editable list view, extremely long text entered in a single cell can cause additional white space on the right side of the page. Long text is text that occupies approximately the entre width of the screen. This issue occurs primarily in Chrome.

Workaround – If you expect users to enter large amounts of text into the cells of a column, configure the column to support text wrapping.

Build tool regen-java-api can fail with -Ddeprecated=true (PL-28992)

Issue – If you run the `gwpc regen-java-api` command with the flag `-Ddeprecated=true`, the command can fail.
Workaround – Specify `dev-deploy` as the value of the `depends` attribute, as follows:
1. Open the `build.xml` file in `PolicyCenter/modules/ant` in an editor.
2. Find the entry that starts as follows:

```
<target name="regen-java-api" depends="init"
```

3. Change the value of the `depends` attribute to `dev-deploy`, as follows:

```
<target name="regen-java-api" depends="dev-deploy"
```

4.   Save the file and then run the `gwpc regen-java-api` command.

Administrative command-line tools cannot refresh WSDL (PL-29021)

Issue – Some administrative command-line tools rely on web service implementation classes, such as `MaintenanceToolsAPI.gs`. Source files for these classes use the `@Export` annotation, which allows you to edit the file. In this release, the administrative command-line tools cannot refresh the WSDL for these classes. Any change to the web service implementation class that changes the WSDL can prevent the administrative tools from working. Therefore, the only changes you can make to these classes are changes that do not affect the WSDL. For example, you can add `@WsiPermission` annotations to change the permissions without changing the WSDL.

Workaround – Guidewire is aware of this issue.

No entity changed event fired through a one-to-one entity relationship (PL-26224)

Issue – The existence of a one-to-one entity relationship between two entities prevents an entity change event from being fired, even if it is supposed to. For example, in ContactManager there is a foreign key from ABContact to Address for the primary address, and a one-to-one relationship from Address to ABContact. Because of the one-to-one link, no event changed event is fired for any change in the primary address.

More generally, if entity E1 has a one-to-one relationship to entity E2, no entity changed event is fired for E2 if a property on E1 changes.

Workaround – You might be able to take advantage of a secondary change caused by the initial change, with the secondary change triggering the event to be fired instead. In the previous ContactManager example, making sure that changes to all properties in Address cause a History record to be generated ensures that the entity changed event will be fired.

Free-text Search indexes cannot be upgraded from 7.0 to 8.0 (PL-29062)

Issue – To upgrade free-text search in PolicyCenter from 7.0 to 8.0, you install the newer version of the Guidewire Solr Extension in 8.0 on top of version from 7.0. Because the PolicyCenter data model also changes, free-text search indexes from 7.0 become obsolete. Guidewire does not support migrating free-text search indexes from 7.0. to 8.0.

Workaround – For Oracle, upgrade the Guidewire Solr Extension by following instructions in the *Upgrade Guide*. Then, run the new free-text batch load command to rebuild the indexes based on the new data model, after you complete your database upgrade. For SQL Server, Guidewire is aware of potential performance problems running the free-text batch load command after you complete your database upgrade.

Chrome browser cannot display product documentation in HTML format (DOC-7251)

Issue – If you use the Google Chrome browser, you can view the HTML Guidewire product documentation only if it is served by an HTTP server using the `http://` protocol. The Chrome browser cannot load HTML product documentation from your local disk by using the `file://` protocol.

Workaround – Use a different browser, such as Microsoft Internet Explorer. If you use the HTML documentation only to access the PDF files, you can open them directly in the `pdf` subdirectory of the `doc` directory.

New mechanism for reloading Gosu classes (DOC-8218)

Issue – In past releases, you could modify your PCF files and Gosu classes in Studio, and then reload the changes into the running server by pressing Alt+Shift+L in the application user interface. This shortcut no longer loads Gosu classes.

Workaround – Have the server reload your Gosu classes, in Studio, click **Build→Make Project**. When Studio is finished compiling your project, the changes will be loaded. You can also restart your server to load the Gosu classes.

Database Consistency Checks - The date parameter to check against is not correct (PL-30098)

Issue – Database Consistency Checks that check date parameters compare against an incorrect date.

Steps to reproduce the problem:

1.   Start PolicyCenter 8.0.2 on a fresh database.

2. Run database consistency checks. This reports zero failures.

3. Create records for entities that store the datetime when they were created.

4. From server tools, run consistency checks again. Consistency check failures related to the timestamp are reported.

The creation date of an object cannot be after the time. The consistency checks fail because the creation time is compared to the time of the first consistency check run of the application server session instead of to the current database time.

Workaround – Ignore database consistency check errors caused due to timestamp. Or, restart the application server and run consistency checks for the results to be accurate.

Some PCF widgets in ClaimCenter are miscategorized in Studio (PL-23454)

Issue – Some PCF widgets in ClaimCenter are miscategorized in Studio. This means that on certain screens (such as wizard screens), widgets that you may want to use are not available in the toolbox.

Workaround – Find a usage of the missing widget on a different screen, and then copy and paste it to the screen that you are editing.

Validation for WS-I Web Services not invoked by Studio (PL-29136)

Issue – Studio does not validate WS-I web service implementation classes.

Workaround – Before deploying new web service code, run the "gwxx verify-resources" tool from the command line.

Custom inbound integration implementations require OSGi (PL-28956)

Issue – Custom implementations of the `InboundIntegrationStartablePlugin` or `InboundIntegrationMessageReply` plugin interface must use OSGi.

Workaround – Implement the plugin interface as an OSGi plugin.

JMS inbound integration connectivity problem (PL-30199)

Issue – The JMS inbound integration APIs throw exception "JMS Integration Service not supported on server runtime".

Workaround – Guidewire is aware of this issue.

Screen elements might disappear when using the AddressAutoFillInput widget (PL-29653)

Issue – Quick use of the `AddressAutoFillInput` widget might lead to a JavaScript error, which causes screen elements to disappear.

Workaround – Refresh the page, and the elements will reappear.

Data might disappear while typing while a PostOnChange is firing (PL-30055)

Issue – While a PostOnChange is occurring, the screen may continue to appear responsive, but action or keystrokes captured during that time will be erased when the post returns. This is most noticeable on slow networks or with a large amount of processing occurring during the post.

Workaround – Evaluate your users' network speeds to determine how likely it is that this issue will impact them. This is most likely to affect screens with quick, expert data entry. To improve the speed of the post, set a narrow target for `postOnChange`, or set it to `DATA_ONLY` if the user interface does not need to change.

Minute selection in time picker resets to current time (PL-30160)

Issue – When you open a date/time picker, the minutes are reset to the current time's minutes. For example, if the current computer time is 12:37 and the currently set time in the picker is 5:50, then opening and closing the picker changes the time to 5:37 and marks it as an edit.

Workaround – Guidewire is aware of this issue.

Guidewire PolicyCenter 8.0.1 Release Notes

Release 8.0.1

Release Notes Update

> **IMPORTANT** These release notes replace the release notes that were included with the official product release. Please disregard the earlier version of the release notes.

Overview

These release notes contain the following:

- "Release Information" on page 411
- "Critical Action Required" on page 412
- "Installing This Release" on page 412
- "Support" on page 412
- "Issues and Major Changes" on page 412
- "Improvements and General Issues" on page 413
- "Known Issues and Limitations" on page 428

Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

> **IMPORTANT** If you skipped one or more upgrade releases to PolicyCenter, be sure to read the release notes for those releases to learn about changes and fixed issues.

Version Number

This release of Guidewire PolicyCenter is 8.0.1.
- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 7.0.1 or later. The latest maintenance release is preferred.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 7.0.0 or later. The latest maintenance release is preferred.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 7.0.7, 8.0.1, or later. The latest maintenance release is preferred.

Critical Action Required

An issue that potentially causes data corruption was discovered in the quote purging batch processes. This issue can occur whether these processes have been intentionally enabled or not. You must explicitly disable the quote purging batch processes. For guidance and instructions, see Knowledge Base article *"Potential for data corruption when using Purge Batch Processes"* on the Guidewire Resource Portal.

This issue will be addressed in a future release.

Installing This Release

Refer to the following:
- The *PolicyCenter Installation Guide* for general installation information
- Prior PolicyCenter release notes for any versions that you have skipped

Support

For assistance, visit the Guidewire Community.

Guidewire Customers

- https://community.guidewire.com

Guidewire Partners

- https://partner.guidewire.com

Issues and Major Changes

This section contains issues and major changes that may affect your installation. For information on new features and major changes, see the topic "New and Changed in PolicyCenter 8.0.1" in the PolicyCenter New and Changed Guide.

Product Designer Localization

Product Designer can now be localized by installing a Guidewire language pack or by custom configuration. Localization of Product Designer display strings and online help requires that you unpack the `productdesigner.war` file, add translations, and then create a new `.war` file. The unpack and repack operations are automated by new scripts located in the `productdesigner/language-config` directory. A new `language.properties` file enables customers to configure available and default languages. The process is documented in a `readme.html` file in the same location.

Product Designer support language hierarchy when displaying read-only product model fields. Therefore, when working with a Product Model that has fields derived both from a locale-specific language and a generic language, Product Designer displays read-only values from both locations. For example, if your product model contains both Japanese (`ja`) and Japanese (Japan) (`ja_JP`) fields, Product Designer displays values from both locations. It first checks for a locale-specific field. If a locale-specific field exists, Product Designer displays it. If no locale-specific field exists, Product Designer displays the equivalent field from the appropriate generic language.

Product Designer Requires Committing All Change Lists Before Upgrading PolicyCenter

Before upgrading PolicyCenter to this release, you must commit all changes in all open Product Designer change lists. Uncommitted changes are discarded during the upgrade process.

Base PCF File Changes

The link below requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 8.0.0 to 8.0.1

To view a report of the changes to the base PCF files, *click here*.

Base Resource Changes

The link below requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 8.0.0 to 8.0.1

To view a report of the changes to the base resources, *click here*.

Changes in This Release Provided in Upgrade Diff Report

Guidewire provides a report detailing certain differences between the current release and your prior release. This report describes changes in display keys, entities, typelists, and the Gosu API. To obtain your custom Upgrade Diff Report, visit the Guidewire Community.

Improvements and General Issues

This topic describes improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This is not a comprehensive list.

- "PolicyCenter improvements and general issues" on page 413
- "Product Designer improvements and general issues" on page 420
- "Platform improvements and general issues" on page 421

PolicyCenter improvements and general issues

ID	Description
Account	
PC-16669	As of this release, the `Lookups.gs` class is deprecated. See the class for suggested replacement methods.
PC-22022	The `ILocationPlugin` was renamed `AccountLocationPlugin`.
Administration	
PC-22323	Fixed an intermittent issue with the **New Producer Code→Roles** tab. In some instances, selecting the **Name** drop-down list resulted in an "Invalid server log" message.

ID	Description
Archiving	
PC-17887	The associated policy transactions card on the `AccountFile` contacts section now shows data for the `PrimaryNamedInsured` contacts of an archived policy period.
PC-21557	Fixed an issue in the policy **Summary** PCF file which resulted in an error in the console log when trying to access an archived policy.
PC-21873	Added an upgrade trigger that updates monetary underwriting issue values from big decimal to include currency, such as USD.
Contacts	
PC-23017	Contacts in PolicyCenter and ContactManager now synchronize correctly after a contact merge in ContactManager.
Financials	
PC-17321	Added **Earned Premium** to the policy **Summary** screen.
Infrastructure	
PC-20234	Fixed an issue where `PCSolrMessageTransportPlugin`, `PCSearchPlugin`, and `PCSolrMessageTransport` were initialized during PolicyCenter start up when free-text search, which uses an integration with the full-text search engine Solr, was not enabled.
PC-21895	Fixed an issue with the Solr batch loader that prevented Solr from running correctly.
PC-21908	Fixed an issue which caused the Solr batchload to generate `InvalidSlice` differences when running **Consistency Check** from **Internal Tools** > **Free-text Search**.
PC-21965	Fixed an issue which prevented the user from starting rewrite jobs if the rewrite effective date was same as any anniversary date or other user split that resulted in a split period.
PC-22023	The following methods were deprecated: • `IPolicyHoldJobEvalPlugin.findJobsToEvaluate` • `PCArchivingUtilInternal.findArchivingCandidates` • `PCArchivingUtilInternal.findRestoreCandidates`
PC-22048	Fixed an issue where changes in a rewrite were not correctly carried forward for the whole term when the cancellation/rewrite was out-of-sequence.
PC-22412	Removed HTML occurences from labels in PCF files and replaced with /n. HTML is not allowed in labels.
Integration	
PC-14630	The `IPolicyPortalAPI.gs` file has been removed.
PC-15755	You can now add and remove territory codes on locations. You can add multiple territory codes to locations and to lines of business. In previous releases, new territory codes were added when creating a location and you could not add or remove them. However, the PolicyCenter user interface allows for only one territory code per line of business. Thus you cannot subsequently add new territory codes to a location. This is a user interface limitation which can be modified through configuration.
PC-15836	In this release, system permission checks have been added to many of the PolicyCenter APIs. Refer to the specific API classes for information about required permissions.
PC-17810	Fixed an issue with multicurrency and the `gw.plugin.billing.impl.StandAloneBillingSummaryPlugin` plugin implementation. Previously, the plugin returned values with the default currency. Now it searches for the specified period and returns values in that settlement currency if the period can be found. Otherwise it reverts back to the default currency.
PC-20813	Prior implementations of the *conversion on renewal process* did not set the `MostRecentTerm` on the converted policy. The renewal batch process can only operate on policies containing a `PolicyTerm` with the `MostRecentTerm` bit set to true. This issue prevented the policy from being picked up by the Renewal batch process and from being renewed.

ID	Description
PC-21657	Fixed issues that occurred when integrated with BillingCenter. Previously you could issue a policy submission when no existing unapplied funds or existing invoice stream was selected.
PC-21658	Fixed an issue on the **Payment** screen. Previously, if you selected **Plan Type→Reporting Plan**, there were no radio buttons to select an existing invoice. This occurred when integrated with BillingCenter.
PC-21679	In the previous version, the **Unapplied Funds** field on the **Account Billing** screen displayed only the **Default Unapplied** value from BillingCenter. BillingCenter now sends the total of all unapplied funds.
PC-21715	Made improvements to code and code comments in the following integration plugins: • `gw.plugin.contact.ab800` – ContactManager integration • `gw.contactmapper.ab800` – ContactManager integration supporting classes • `gw.plugin.claimsearch.cc800` – ClaimCenter integration • `gw.plugin.billing.bc800` – BillingCenter integration
PC-22008	Renamed the `selectFinalAuditCancellationSchedulePattern` method to `selectFinalAuditSchedulePatternForCancellation`. Renamed the `selectFinalAuditExpirationSchedulePattern` method to `selectFinalAuditSchedulePatternForExpiredPolicy`. The old names are deprecated.
PC-22016	Previously, the methods in the billing summary plugin (`IBillingSummaryPlugin`) plugin interface returned special data types defined by marker interfaces. The methods in the billing summary plugin return instances of a classes defined by Java interfaces. The new Java interfaces define methods, primarily property getter methods. If you previously implemented this plugin, you must create classes that implement these new interfaces and update the billing summary plugin. See the documentation of the billing summary plugin for more information.
PC-22019	The `IBillingSystemPlugin` plugin interface now returns more strongly typed entities. These entities represent the data based on PolicyCenter and BillingCenter integration, but can be configured as needed.
PC-22020	Made changes to the `IBillingSystemPlugin` plugin interface. The `isAccountExist` method is now `accountExists`. The `isProduceExist` method is now `producerExists`. The `isAccountExist` and `isProduceExist` methods are deprecated. Changed parameter of `updatePolicyPeriodTermConfirmed` method from `Boolean` to `boolean` so that it no longer allows `null`. The previous method signature is deprecated.
PC-22021	Deprecated `ClaimDetail getClaimDetailByClaimNumber(Claim claim)` since it referred to `ClaimNumber` which is not a part of the API signature. Use `ClaimDetail getClaimDetail(Claim claim)` in its place.
PC-22062	Plugins that communicate with WSI web service APIs are now thread-safe.
PC-22132	Previously, the *conversion on renewal process* did not set the `IssueDate` when issuing the renewal job. This prevented subsequent jobs from being taken on the Policy because the default `PolicyPlugin` checks for `Policy.Issued` in `canStart<job>`.
PC-22263	Previously on the **Policy Billing** screen, the **Period** drop-down showed both bound and unbound periods even though BillingCenter only has bound periods. Now the **Period** drop-down only shows bound periods.
PC-22337	To support multicurrency, the `IBillingSummaryPlugin` plugin interface changed arguments and return values that represent money (for example, `BigDecimal` types) to the new `MonetaryAmount` type.
PC-22594	Producer codes can be configured with multiple currencies and each currency configured has a commission plan associated with it.
PC-22661	The `BundleEnhancement.turnOffValidation` method has been deprecated.
PC-22694	In `IBillingSystemPlugin`, removed the `CompatibilityMode` flag which provided backwards-compatibility with PC 4.0.
PC-22711	Fixed a potential endless loop that occurred when PolicyCenter waited for ContactManager responses before sending messages to BillingCenter. It is not recommended to wait for ContactManager, however, if you must ensure that ContactManager processes the request before sending messages to BillingCenter, you can uncomment the code in the send method of the `BillingMessageTransport` plugin.

ID	Description
PC-22814	Previously in the base integration with BillingCenter, you could not add new currencies to an existing producer code in PolicyCenter.
PC-22840	Previously in single currency mode on the **Create New Producer** screen, the **Currency** field was not editable and did not default to the system's default currency. As a result the producer was not authorized to bind policies in any currency.
PC-22862	With this fix, PolicyCenter no longer sends `NewInvoiceStreams` regardless of selected options in the user interface but rather honors the user selection.
PC-22870	In the base integration with BillingCenter, when a user updates an existing organization in PolicyCenter and adds agency bill, that information is now sent and persisted in BillingCenter.
Internationalization	
PC-21541	Fixed a `DBNullConstraintException` that occurred while creating new organization in a locale that was not the default.
PC-21671	Added additional display keys so that the PolicyCenter to ClaimCenter web service integration exceptions are localizable.
PC-21848	In a multicurrency system, you must specify a **Currency Selection** for a **Producer Code**.

PC-21897	Web service interfaces were modified to transmit the following international name and address fields:
	• Name — `FirstNameKanji`, `LastNameKanji`, `Particle`, and `CompanyNameKanji`
	• Address — `AddressLine1Kanji`, `AddressLine2Kanji`, `CityKanji`, `CEDEX`, and `CEDEXBureau`
	The following web services were modified:
	• `gw.webservice.pc.pc800.account`
	  ◦ `AccountAPI.gs`, `AccountSearchInfo.gs`
	• `gw.webservice.pc.pc800.community.datamodel`
	  ◦ `AddressDTO.gs`
	  ◦ `ProducerCodeDTO.gs`
	• `gw.webservice.pc.pc800.contact`
	  ◦ `AddressData.gs`
	  ◦ `AddressDataCopier.gs`
	• `gw.webservice.pc.pc800.gxmodel`
	  ◦ `AccountLocationModelEnhancement.gsx`
	  ◦ `ContactModelEnhancement.gsx`
	  ◦ `AddressModel.gx` and `AccountLocationModel.gx`
	• `gw.plugin.contact.ab800`
	  ◦ `ABContactAPISearchCriteriaEnhancement.gsx`
	  ◦ `DemoContactSystemPlugin*.gs`
	  ◦ `AddressInfoEnhancement`
	  ◦ `AccountLocationModelEnhancement.gsx`
	• `gw.webservice.pc.pc800.ccintegration`
	  ◦ `CCContactGenerator.gs`
	  ◦ `CCPolicyGenerator.gs`
	  ◦ `CCAddressEnhancement`
	  ◦ `PolicyLocationInfoEnhancement.gsx`

PC-21951	Database indexes for Japanese Kanji columns on PolicyCenter entities are provided but commented out. If you wish to index the Kanji columns, uncomment the entity delegates referenced on the `AffinityGroup`, `Person`, and `PolicyContactRole` entity extensions.
PC-22534	Fixed an issue with importing French and Japanese address book entries from ContactManager.
PC-22597	In the previous version, the CEDEX field was visible on the **Search** screen when it should not have been.

ID	Description
PC-22600	The `PALineDriversValidator` now validates an address's fields based on its country. Previously it assumed `State` was a required field because it was US-specific. While this code is generalized to work with any country, you must update the display key part (`PALineDriversValidator#FieldLabels`) whenever you add a new required visible address field. In the base configuration, this validator validates `AddressLine1`, `State`, `PostalCode`, and `City`.
PC-22609	Removed workarounds to PL-28550 in PCF files.
Jobs	
PC-16312	Fixed an issue where a pending renewal could not be withdrawn after cancellation of the previous term. The renewal is now non-renewed.
PC-18094	The signature of `IPolicyTermPlugin.calculatePeriodEnd(Date, TermType)` has been changed to `calculatePeriodEnd(Date, TermType, PolicyPeriod)`. The prior implementation of `calculatePeriodEnd` did not correct for date-drifting: two half-year terms did not necessarily cover the same number of days as an annual term. To enforce that the two term durations are equivalent, include the prior history of the Policy. In the base configuration, use the `PeriodStart` of the first `PolicyPeriod` in the `Policy` to correct for date drifting. Now the plugin implementation in the base configuration calculates the `PeriodEnd` as follows: A policy incepting on day *X* of a month expires/renews on the same day unless: • The expiring month has fewer than *X* days – The period expires on the last day of the month. • The policy started on the last day of a month and the expiring month has greater number of days than the starting month – The period expires on the last day of the month. This fix adds a new Boolean `shouldPerformDateReconciliation` on Job. This Boolean is used by `PolicyTermPlugin.calculatePeriodEnd` method to determine if the policy period's job should be reconciled. By default this is `false`, however, `SubmissionImpl` and `RenewalImpl` overwrite this to `true`.
PC-22044	Fixed an issue that resulted in validation and underwriting issues being erroneously raised for future periods, even when there were no future periods.
Lines of Business	
PC-16160	Fixed an issue where adding a coverage in a renewal term,then later adding the same coverage in the previous term resulted in a stack dump when the second policy change was applied to the renewal term.
PC-19305	When selecting a quoted version in a multi-version Commercial Package Policy submission, you can now visit the wizard step `ViewQuote`.
Policy File	
PC-17673	Fixed an issue where selecting a policy number from the **QuickJump** text box caused a null pointer exception when you navigated to the **Reinsurance** screen.
PC-21532	In previous releases, `PolicyPeriod.PolicyNumber` used the string literal `Unassigned` to indicate no policy number has been assigned to this period. In this release, the code was modified to use the `null` value.
PC-22567	Fixed an issue where PolicyCenter threw an exception if you tried to set the **Base State** on the **Policy Info** page to **None**. PolicyCenter now displays an error message if the base state is required.
Product Model	
PC-17074	Previously, if two products had the same name, a display such as a drop-down list would have two identical items. If a product has a duplicate name, the product code is appended so that the items are unique.
PC-22629	PolicyCenter 8.0.0 added new typecodes, `Boolean` and `Other`, in the `CovTermModelVal` typelist. This typelist is used by the **Value Type** field in the product model. In this release, these values (`Boolean` and `Other`) have been retired. `Boolean` coverage terms should instead be modeled as generic coverage terms instead of option coverage terms. Generic coverage terms do not have a **Value Type** to be set from the `CovTermModelVal` typelist. In the default configuration, usage of these value types has been eliminated from the product model. These changes, as with all other changes to the product model, should not be merged into a production server. If you attempt to merge these types of changes in a production server, a locking error can occur. If your existing configuration uses **Value Types** of `Boolean` or `Other` in the product model, consider modifying the type of the coverage term from option or package to generic, or adding these values back through an extension of `CovTermModelVal`.

ID	Description
Rating	
PC-17594	Closed as unreproducible an issue in impact testing where PolicyCenter threw an exception when creating baseline policy periods. This occurred if the number of workers for `ImpactTestingTestRunWorkQueue` was `10` in `work-queue.xml`.
PC-17651	Updated `RateBookPanelSet.pcf` to use the correct `sortBy` properties when sorting the rate book list.
PC-18823	In `gw.rating.rtm.query.RatingQueryFacade`, you can use the new method `getAllFactors` method to query a rate table and get back the factors. The return value includes the name of each factor as a `String`, and the value of each factor as an `Object`. The caller needs to cast the return object to get the values.
PC-19250 PC-19214	When a rate table factor was defined with a decimal type and a scale > 3, rating calculations still used a scale of 3. This has been fixed to use the user-specified scale for the factor. To correct a related issue in rating worksheets, the limit on decimal places shown after the decimal point was increased from 6 to 8 for `BigDecimal` values. If the underlying `BigDecimal` has more than 8 decimal places then ellipses are displayed at the end. For example, 1.732050807568877 is displayed as 1.73205080.... When exporting rating worksheets to XML, the number is now a `BigDecimal`, rather than a formatted value such as 1.73205080....
PC-19970	Fixed an issue where the user received a validation error when attempting to promote the rate book or add the rate routine to the rate book. This occurred if the rate table was referenced in a rate routine but not contained in the rate book.
PC-20277	Fixed issues where rating worksheets impacted the performance of the quote significantly and took up a large amount of disk space.
PC-20908	Fixed an issue where rating did not consider currency when calculating state tax.
PC-21341	The `RateBookEnhancement.selectRateBook` method now accepts a `UWCompany` parameter. When `UWCompany` is specified, the method returns the rate books that match the `UWCompany`. When `UWCompany` is not specified, the method returns the rate books that have a null `UWCompany`.
PC-21528	In a rate table included in a rate book, the contents of the rate factor column appear in bold.
PC-21673	Previously, PolicyCenter threw a `DBNullConstraintException` when defining a new rate when the language was not the `DefaultApplicationLocale`. There is now a localization table for label values. Before, there was only a localized table for column label values.
PC-21794	Removed an extra field label for exact match parameters on the **New Rate Table Definition** page.
PC-21911	PolicyCenter now generates warnings when a rate query returns nothing.
PC-22079	When editing a rate routine, the **Error #** column is now always visible. This prevents columns from shifting after validation.
PC-22120	When editing a rate routine, a variable which first appears in an IF block was not always usable in a subsequent IF. This issue is fixed.
PC-22424	Fixed an issue in rate routine if statement that caused a `ClassCastException` when doing a comparison to a typekey and when that typekey was not backed by a generated class.
PC-22625	The `RateBookMatcher`, `QueryFilter`, and `RatingQueryFacade` Gosu classes are now exported and available in Studio. You can use these methods to add or modify filters for finding a rate book.
Reinsurance	
PC-21674	Fixed an issue that caused text to be cut off in the **View Ceded Premiums** on the **Reinsurance** screen.
PC-21707	Fixed an issue where the dates associated with costs ceding to agreements did not agree with the dates of attachment.
PC-21852	Fixed an issue where a long `UWIssue.Issuekey` prevented binding, issuance, and other operations on policies. When trying to perform one of these operations you received this error: Key is not formatted correctly for the field Issuekey: Exceeds the maximum length of 255 characters.

ID	Description
PC-21863	Fixed an issue where the PremiumCeding batch process did not cede premiums if agreements were added after the policy was issued. Previously, validation warning prevented ceding.
PC-22033	The `PolicyPeriodImpl` class accesses the `PolicyPeriodPlugin`. Previously `PolicyPeriodPlugin` accessed the `ReinsurancePlugin`. Now `PolicyPeriodPlugin` no longer calls methods from `ReinsurancePlugin`.
PC-22658	Surplus agreements can now have their ceding reduced to a percentage of their layer. This allows, for example, multiple surplus agreements to share a given layer.
Teams	
PC-15262	On the Team screens in previous releases, views of lists of specific jobs and activities at a group level would also include those matching the criteria in all subgroups of the current group. In this release, these lists now only include the current group, and do not include the jobs or activities in all subgroups. As membership in a specific group does not imply membership in all the subgroups, this detailed view is more consistent with group membership.
Search	
PC-21872	In `schema.xml`, `omitTermFreqAndPositions` is set to `false` in the base configuration. The `schema.xml` is one of the configuration files for the Guidewire Solr extension that provides free-text search.
PC-22483	In this release, unassigned `PolicyNumber` values in Solr change from `Unassigned` to blank. As a result of this change, you cannot find these unassigned submissions by specifying a `PolicyNumber` search pattern. This change does not require a reload of the Solr index after an upgrade. However if not reloaded, previously existing unbound submissions retain their unassigned `PolicyNumber` value and match certain queries differently than the newly unbound submissions. To avoid this discrepancy, it is recommended that you reload the Solr index.
Security	
PC-17871	A product becomes unavailable for a particular producer code when there is a bound submission of that product type associated with a different producer code. This was enforced by filtering the product selection on the **New Submission** screen. However, previously it was possible to select an unavailable code from the **Policy Info** screen once the submission has been created. The user was not notified of producer code unavailability. After issuance, the user would see that the submission was declined without any explanation as to the cause. There is now validation on the **Policy Info** screen which gives the user immediate feedback. Additionally, the drop-down menu on the **Policy Info** screen now only displays available producer codes.
PC-21802	Fixed an issue where a user without proper permission was able to create an account. Improved producer code permission handling with multiple users and multiple producer codes.
PC-22061	The producer code currency is set to the default currency of the system in single currency mode.
Upgrade	
PC-21773	As of this release, `ValueType` is no longer an immutable field on direct and option coverage term types.
PC-21874	Fixed upgrade trigger to update database values moving from big decimal to monetary amounts. After upgrading from PolicyCenter 7.0 to PolicyCenter 8.0, there was an exception during the run command: Argument "10000" must be of the form "AMOUNT CURRENCY".
PC-22031	The upgrade tool now runs without performing validation for the `valueType` parameter against the base configuration product model.
PC-22053	Upgrade now includes Gosu code for post-processing that fills in the `BusinessVehicle` foreign key values on `BACost`.
PC-22844	Due to changes in the product model, the product model must be resynchronized during upgrade. The upgrade trigger forces the system to resynchronize the lookup tables with the XML configuration.

Product Designer improvements and general issues

ID	Description
BCC-1566	Fixed an issue where some tables were not highlighted when missing data caused validation alerts.
BCC-1568	Added validation to check the lookup table when adding an object.
BCC-1580	Fixed an issue where validating an object after another user has deleted it causes an exception.
BCC-1581	Fixed an issue in system tables where conflicts were displayed by column, instead of by row.
BCC-1586	Fixed an issue where changing the language did not change the language of some of the text on the **Changes** panel.
BCC-1599	Added validation when deleting objects that are being used by other objects, for example, categories that are used by coverages.
BCC-1626	Fixed an issue where deleting an object due to a conflict was not removing the object from the navigation panel.
BCC-1628	Added a tooltip to the error icon in the **Changes** panel.
BCC-1636	Improved the quality and stability of conflict detection. Conflicts occur when two or more users in a multiuser scenario edit the same product model object at the same time. When conflicts occur that Product Designer cannot automatically resolve, notifications guide affected users to choose how to resolve the each conflict.
BCC-1640	Fixed an issue where coverage term options are duplicated in the navigation panel.
BCC-1644	Fixed an issue where editing a question's help text in different language/locale displays the help text under the question in language/locale in which the question was last edited.
BCC-1658 BCC-1854 BCC-1855 BCC-2015	Product Designer can now be localized through the use language packs as well as by customer configuration. Localization of Product Designer display strings and online help requires that you unpack the `productdesigner.war` file, add translations, and then create a new `.war` file. The unpack and repack operations are automated by new scripts located in the `productdesigner/language-config` directory. A new `language.properties` file enables customers to configure available and default languages. The process is documented in a `readme.html` file in the same location.
BCC-1679	Fixed an issue where the navigation tree was not updated when a code was added, when that code was identical to an existing code except with different capitalization.
BCC-1693	Fixed an issue where after creating a new change list and making it the active change list, the navigation panel did not indicate objects could be expanded.
BCC-1700	Fixed an issue in Question Sets where reordering multiple lines of help text was not reflected in the preview.
BCC-1706	Fixed an issue where newly added conditions and exclusions were not added to the end of a list.
BCC-1715	Improved the configuration of new system tables. Product Designer now creates and populates an XML file with the required XML header elements when you add a new system table.
BCC-1745	Patterns were disabled in offerings when these steps were followed: 1. Add a pattern to a product. 2. Disable the pattern in an offering. 3. Revert the changes. 4. Re-add the same pattern to the product. This issue has been fixed. Patterns that are removed and then re-added are now correctly enabled in offerings
BCC-1773	Fixed an issue where adding modifiers to policy line and product did not correctly update the navigation panel.
BCC-1812	Fixed an issue where the last row of the system table could not be deleted.
BCC-1814	Improved support for special characters.
BCC-1864	Fixed an issue where the date picker buttons were not localized.
BCC-1865	Fixed an issue where the system table page tool was not localized.

ID	Description
BCC-1866	Fixed an issue where the **Changes** panel was not updated with new descriptions.
BCC-1900	Fixed an issue where validating a question with a missing choice code would throw an exception.
BCC-1905	Fixed an issue where system table conflicts were not displayed properly.
BCC-1927	Fixed an issue where system table changes were not detected.
BCC-1940	Fixed an issue where a pattern's title was not updated after a conflict.
BCC-1945	Fixed an issue where reverting a category change caused an exception.
BCC-1949	Fixed an issue where conflicting changes between the default and edited values for coverage terms were not detected.
BCC-1954	Fixed an issue where adding a new change list did not correctly set the active change list.
BCC-1961	Fixed an issue where conflicts were not detected when deleting system table rows.
BCC-1977	Fixed an issue where editing incorrectly remained enabled when an **Accept Deletion** was pending.
BCC-1985	Fixed an issue where users could not see the list of questions when a conflict was pending.
BCC-1992	Online help content can be changed or translated when configuring Product Designer. Each language specified in the Product Designer language.properties file can have corresponding online help. Guidewire recommends copying the provided en_US files to use as a starting point. To ensure that the help files display properly in the 8.0.1 release, each online help page must conform to the following rules: 1. Must have an `.html` filename suffix. 2. Must not contain Javascript. 3. Must have absolute or relative image references using the tag `<img src="..."` 4. Must not have external link references. 5. Must have relative Internal links using the tag `"<a href="..."'` 6. Must have the same file name as the page it describes. For example, the help file for Product Designer page `"http://<pd url>/#PolicyLines()"` must be named `PolicyLines.html` 7. Must not share image files among different languages. Even though the en_US help contains only images of icons with no text, you must reference a separate copy of these images for each language you add. For more information, see the `readme.html` file located in the `productdesigner/language-config` directory. Also see the previous entry for BCC-1658.
BCC-1993	Fixed an issue where validation of an availability table caused an exception if the backing files had been changed.
BCC-1998	Fixed an issue where using certain source control tools caused errors when launching Product Designer. Source control tools that caused errors are those that put additional folders into the configuration, for example, CVS.
BCC-2004	Fixed an issue where deleting a question that had a conflict removed other edits from the page.
BCC-2017	Fixed an issue where selecting a different change list after selecting a different language did not change the language in some fields.
BCC-2024	In situations where a user's active change list is deleted, Product Designer now prompts the user to select a new active change list.
BCC-2029	Added conflict detection for situations where multiple users delete the same object.
BCC-2074	Fixed an issue where a user adding a change list caused another user session to change its language.
BCC-2075	Fixed an issue where clicking Validate was removing the selected setting on the Question and Incorrect Answer pages.
BCC-2082	Fixed an issue where after resolving a conflict, a question set radio button does not show the selected value.

Platform improvements and general issues

ID	Description

Application Server

| PL-27203 | JBoss 6 deployment now works properly. |

Archiving

| PL-26930 | Modified the `restoreSeveredTransactionOffsetOnsetOnsetLinks` method so it can be called from Gosu. |

Batch Processes

| PL-18406 | In releases prior to 8.0.0, you created custom batch processes by developing Gosu classes that implement the `BatchProcess` interface and subclass from the `BatchProcessBase` class. You configured custom batch processes in the `scheduler-config.xml` file to run on a schedule or on demand from the Administrative user interface. Custom batch processes ran only on the batch processing host in the cluster.
Beginning with release 8.0.0, you also can create custom batch process by developing Gosu classes that subclass from the `WorkQueueBase` class. You configure this newer type of custom batch process in the `work-queue.xml` file, with definitions for its worker processes that run on other servers.
Note: The methods on `WorkQueueBase` are called from different threads at different times on different servers. So, do not attempt to use class properties to configure or maintain state. |

Bundles and Transactions

| PL-22240 | The `touch` method has been reimplemented. Please refer to the *Gosudoc* for detailed usage. |

Cognos Integration

| PL-21153 | Thread usage on the LDAP server is now managed more efficiently. |

Command Line Tools

| PL-27378 | Removed `debug-start` from command line option. |
| PL-28544 | The obsolete command line command `copy-theme` has been removed. |

Configuration Upgrade

PL-27207	Log file of upgrade tool now reflects the difference between platform upgrade triggers and application-specific upgrade triggers.
PL-27208	You can now expand all and collapse all in the upgrade tool.
PL-27209	You can now right-click to export the directory tree within the upgrade tool.
PL-27338	Fixed an issue where running upgrade multiple times created nested rules folders.
PL-27744	Fixed an issue that occurred if you added any `ClaimContactInput` widgets that use `postOnChange`, or if you modified a base `ClaimContactInput` widget by adding `postOnChange`. In these cases, the `postOnChange` would not get upgraded to the new syntax.

Data Distribution

| PL-25702 | The **Data Distribution** page enables **Download comparison Zip file** and **Download Combined Zip file** options only if two executions are selected. In prior versions, the options were erroneously enabled even if there were not two executions selected. |

Database Configuration

| PL-28656 | Fixed a problem that prevented the `DBAuthenticationPlugin` from working. |

Database Support

| PL-24417 | The performance requirement to set the `action="delete"` attribute for `pc_message` in `database-config.xml` has been removed. |
| PL-28323 | Fixed a bug which prevented scheduling of the Database Statistics process. |

Database Support – Oracle

| PL-27438 | Added support for Oracle Date interval partitioning. |

ID	Description
PL-28085	The Oracle fast add column feature is used when adding monetary amount columns to an entity or a sub-type.
PL-28327	Guidewire added an option to switch off the Oracle adaptive optimization feature for Guidewire applications. See "Configuring Oracle Adaptive Optimization for PolicyCenter" in the *PolicyCenter Installation Guide*.

Database Upgrade

PL-27016	Guidewire has added support for using Oracle's parallel DDL execution feature during upgrade. The `createIndexInParallel` attribute of the `<upgrade>` element has been replaced with the new `degree-parallel-ddl` attribute. See "Configuring Parallel DML and DDL Statement Execution" in the *Upgrade Guide*.
PL-27865	Guidewire resolved a rare issue in which the server would be able to start after an incomplete upgrade but subsequent upgrade attempts would fail.
PL-27918	The Database Upgrader now honors the database statistics configuration.
PL-28041	The `updatestatistics` attribute in `database-config.xml` now controls both deletion and collection of database statistics during upgrade.
PL-28122	The `DeferredUpgradeTasks` process is now profiled and appears under Guidewire Profiler. This process is used when deferring creation of archive indexes until after the upgrade. See "Deferring Creation of Archive Indexes" in the *Upgrade Guide*.
PL-28449	Fixed a bug that prevented handling of statistics on the `ID` column properly.

Document Management

PL-27480	Document templates no longer have size constraints.
PL-27577	You can now call the method `DocumentsUtil.createNewDocument` without needing the current user to have the `doccreate` permission. Use this method if you implement the `IDocumentMetadataSource` plugin.

Email

PL-27921	Due to potential cross-site scripting vulnerabilities, HTML is disabled as the content of an email. If you would like to continue to have HTML email and accept the risk of this vulnerability, you may remove the escaping of the subject and body in the document template.

Entities/Metadata

PL-19023	The *Data Dictionary* has been modified to show references for subtypes
PL-24743, PL-27611	*Data Dictionary* descriptions for core locale fields have been updated.
PL-25622	Fixed an issue where `MonetaryAmount` appeared as two separate fields under Actual Amount in the *Data Dictionary* Data Entity View.
PL-25809	Fixed an issue where typelist codes did not include documentation for the typecodes (name and description).
PL-27465	Added upgrade trigger to insert `xmlns` if missing on `.eti` files.
PL-27501	Added the ability to override/add `keyfilters/typefilters` via `extensions.xml`.
PL-27819	Added a new `overlapTable` attribute to the `edgeForeignKey` and `localization` entities, which specifies that the entity implements the `OverlapTable` delegate.

Globalization

PL-26606	Updated `GroupUserSearchDV.pcf` to support Japanese kanji fields in the standard way represented elsewhere in the application.

ID	Description
PL-27394	Values substituted in display keys that are of type `BigDecimal`, `Date`, and `IMoney` and its implementing classes, such as `MonetaryAmount`, are now properly formatted according to the regional formats in effect for users. For example, the substitution value "123456" is formatted as "123,456" for U.S. (English) or "123 456" for France (French).
PL-27609	Fixed an issue where the display value for locale shifts when an admin user is changing his/her own language/locale when viewing the profile of another user.

PL-28068	The configuration parameter for overriding the default maximum width for labels moved from display key `ExtJS.Form.LabelWidth` to XML element `LabelWidth`, with attribute `width` specified in pixels. Use the `LabelWidth` element in the `language.xml` file for the language that you want to configure. For example,
	
	`<GWLanguage`
	`  code="de_DE"`
	`  name="German (Germany)"`
	`  typecode="de_DE">`
	`  <ExtJsSettings>`
	`    <LabelWidth size="220" />`
	`  </ExtJsSettings>`
	`</GWLanguage>`

Gosu	
PL-18217	A Gosu class now preserves annotations inherited from Java classes in the class hierarchy. Note this change effectively restricts access by a subclass to features tagged with the `@InternalAPI` annotation.
PL-25700	Fixed an issue that caused errors on startup under very specific conditions related to the compilation of particular Gosu classes.
PL-27099	Gosu supports annotations on parameters in methods, properties, and constructors.
PL-27428	The Gosu language now provides limited support for the Java annotation `@SuppressWarnings`, which tells the compiler to suppress warnings. Use this annotation on declarations of a type, function, property, constructor, field, or parameter. Note that local variables do not support this annotation. You must pass a `String` value as an argument to indicate what warnings to suppress. Pass the argument `"all"` to suppress all warnings. Pass the argument `"deprecation"` to suppress deprecation warnings. For example, to suppress deprecation warnings in a Gosu class, add the annotation `@SuppressWarnings("deprecation")` on the line before the class declaration.
PL-27651	Gosu now recognizes the annotation `@java.lang.Deprecated` as a form of deprecation, in addition to `@gw.lang.Deprecated` and the `@deprecated` Javadoc tag.
IntelliJ IDE – Compiler	
PL-28346	Fixed a compilation error when compiling an entity or its extension if the entity had a subtype.
IntelliJ IDE – Debugger	
PL-27875	When debugging Gosu code, you can now browse the structure of an Entity object and inspect the property values stored within it. To enable this, in Guidewire Studio, click **File→Settings**, and then in the Guidewire Studio panel set **Enhance Entities Visualization**.
IntelliJ IDE – Display Key Editor	
PL-27971	Improved typing performance in the Display Key editor.
PL-28677	Fixed a bug in Studio where converting a string into a display key caused exceptions. The behavior has changed slightly, so now the locale folder is required to have a `display.properties` file already in it before it appears in the Create Display Key dialog or the Step Name Localizations tab in the Workflow editor.
IntelliJ IDE – Entity Editor	
PL-26336	Column validation has been reenabled in Studio.
PL-26364	Fixed an exception in the Entity editor when attempting to override a read-only attribute.

ID	Description
PL-26375	Fixed an issue in the New Entity dialog where the `viewEntity` was listing suggestions that were not applicable.
PL-26540	Added additional error notes to the Entity editor to highlight the parent elements if a child is invalid.
PL-27220	In the Entity editor in Studio, when editing the `<tag>` subelement of the `<column>` element, there is now a drop-down list showing available values.
PL-27454	Fixed an issue where creating multiple entity extensions with suffixes would produce an exception.
PL-27715	Fixed an error that would occur when the `effDatedBranchType` attribute was not correct in an entity of type `effdated`.
PL-28089	In the Entity editor in Studio, you are now required to specify a value for the `nullok` attribute.
PL-28529	Fixed an error in Studio when creating an entity extension if there is a Java class under `src`.

IntelliJ IDE – Gosu Editor

ID	Description
PL-19418	Fixed a compilation error during bytecode generation on certain annotations from Java source types.
PL-26640	Fixed an exception that would occur when typing "display" into a Gosu class.
PL-26866	The Gosu `using` clause syntax now has an additional feature for adding additional cleanup code. You can optionally add a `finally` clause that runs after the statement body, even if exceptions occur in the body of the `using` clause. See the *Gosu Reference Guide* for details.
PL-27135	Studio now shows additional warnings for improper usages of internal `gw` classes.
PL-27320	Fixed an issue where pressing Ctrl+O threw an exception in Gosu.
PL-27724	Fixed an issue with some deprecated methods not being shown in strikethrough text in Studio.
PL-27873	Fixed an issue when pasting code into Studio where multiline statements were concatenated and merged with comments.
PL-27893	Fixed an exception that was thrown when creating a new Gosu template.
PL-27943	Fixed an issue with some deprecated methods not being shown in strikethrough text in Studio.
PL-27944	The Gosu language has two new compound assignment operators, which are operators that apply an operation to a variable then re-assign the variable to the result. The new operator `&&=` performs the logical AND operation to the previous value. The new operator `||=` performs the logical OR to the previous value. Both operators work with the primitive type `boolean` or the object type `Boolean` on either side of the operator. For example, suppose you have two `boolean` variables called `needsUpdate` and `flagTest`. The statement `needsUpdate ||= flagTest` has the meaning of `needsUpdate = (needsUpdate OR flagTest)`. Do not confuse these new operators with the other operators `&=` and `|=`, which apply bitwise AND and bitwise OR operations.
PL-28019	Fixed a false compile error in the Gosu editor that manifested when a property getter or setter overrode a getter or setter in a superclass implemented in Java.

PL-28027	Gosu does not support numeric expressions in the for statement after the `in` keyword. The code:
	`for (x in 10) {...}`
	is illegal and must be upgraded with an interval such as:
	`for (x in 0..|10) {...}`
	using the provided inspection in Studio.

IntelliJ IDE – Line of Business Editor

ID	Description
PL-26590	The LOB tab has been removed from `.tti` files in Studio to prevent missing loss types.
PL-26595	Added the ability to more easily select multiple typekeys in Studio using the Ctrl or Shift keys.
PL-26620	Retired typecodes are now shown in strikethrough text.
PL-26826	Fixed an issue where Studio would throw an exception when a `categorylist` was added to a `LossType` typecode.

ID	Description
PL-26971	Fixed an issue in Studio where removing a typecode from its parent also incorrectly removed it from all of its other parents.

IntelliJ IDE – Other

ID	Description
PL-27198	Fixed an issue where Run commands in the QuickJump box did not work when the server was started from Studio.
PL-27862	Fixed an issue where Studio would not suggest types defined on XSD files when trying to create an enhancement.

IntelliJ IDE – PCF Editor

ID	Description
PL-26516	Improved the PCF editor to highlight the correct panel when selecting widgets in nested files.
PL-27147	The `PCFMapping` tool has been updated to include fields such as `PanelIterator`.

IntelliJ IDE – IntelliJ IDE – Plugins, OSGi, Plugins

ID	Description
PL-27497	You can now implement plugin interfaces in Java using the OSGi standard. OSGi is a Java module system and service platform that helps isolate code modules and any necessary Java libraries. Guidewire recommends OSGi for all new Java plugin development. To simplify OSGi configuration, PolicyCenter includes IntelliJ IDEA with OSGi Editor, an application separate from Guidewire Studio. For more information, refer to the *Integration Guide*.

IntelliJ IDE – Refresh

ID	Description
PL-24108	Fixed an issue where the server threw an exception and did not handle newly created enhancements.
PL-28174	Fixed an issue where methods added in entity classes were invalid until restarting Studio.
PL-28187	Fixed an error that would occur after renaming an element in an XSD and then navigating to a Gosu type that contained a usage of that element.

IntelliJ IDE – Typelist Editor

ID	Description
PL-24391	Fixed the typelist editor in Studio to filter out options under the drop-down as you enter text.
PL-26445	Fixed an issue where creating the first extension of a typelist caused an exception.
PL-26613	Fixed an issue in Studio where disabled typecodes looked enabled when selected.
PL-26535	Fixed an exception in the text editor of the typelist view.
PL-27174	Fixed an issue where the default setting for filtering metadata did not apply to the Typelist editor.
PL-27570	Fixed an issue where clicking on the name attribute in a typelist extension would not allow you to override it.
PL-27584	Fixed the Add To Category dialog so retired typecodes appear in strikethrough text as options for filtering.
PL-28177	In Studio, the Entity editor and Typelist editor are now case-insensitive when resolving references to other metadata.
PL-28515	Fixed an issue that generated multiple errors in the Studio Typelist editor in the localization panel.

IntelliJ IDE – Web Services Editor

ID	Description
PL-26576	Added a check to the timeout value of web services to insure that it is lower than Studio's maximum integer.

Integration

ID	Description
PL-28196	This release changed how to configure inbound multi-threaded integrations such as the built-in file and JMS integrations. In previous releases, you added configuration parameters in the Plugins registry in Studio. In this release, you set a single parameter `integrationservice` and then do the rest of the configuration in the new file `inbound-integration-config.xml`. Also, the API details for file and JMS integrations changed. There is a new plugin interface called `InboundIntegrationHandlerPlugin`. Also, the file integration now supports processing one file at a time, rather than one line at a time. For details, refer to the *Integration Guide*.

Other – Persistence

PL-25820	Fixed an issue with whitespace not being trimmed by trimming unicode full-width whitespace.

Profiling

PL-28172	Fixed an issue where the `Purge Profiler Data` batch job would throw exceptions when purging web services profiling data.

Queries

PL-18578	This release adds a new class, `gw.api.database.QuerySelectColumns`, with static methods that help you specify the columns that you want selected in a row query. Instead of passing a Gosu block to the select method, you pass a list of `IQuerySelectColumn` objects, which you construct by using static methods on the `QuerySelectColumns` class. Each `IQuerySelectColumn` object represents a column in the result. For example, the following Gosu sample code creates a row query, with Address as the primary entity. The result includes only the City column for Address instances that match the query criteria.
	
	```uses gw.api.database.Query```
	```uses gw.api.database.QuerySelectColumns```
	```uses gw.api.path.Paths```
	```var addressQuery = Query.make(Address)```
	```...```
	```// Join and condition statements go here.```
	```...```
	```var addressResult =```
	``` addressQuery.select({QuerySelectColumns.path(Paths.make(Address#City))})```
	
	The QuerySelectColumns class includes static methods that represent database functions to help you construct aggregate queries. For example, the following Gosu code returns a count of all Address instances.
	
	```uses gw.api.database.DBFunction```
	```uses gw.api.database.Query```
	```uses gw.api.database.QuerySelectColumns```
	```uses gw.api.path.Paths```
	```var addressQuery = Query.make(Address).```
	```var addressQyuery.withDistinct(true) // Always run aggregate queries```
	``` with Distinct set to true.```
	```addressResult = addressQuery.select({QuerySelectColumns.dbFunction(```
	``` DBFunction.Count(Paths.make(Address#City)))```
	```})```

PL-27474	Fixed an issue where the `Count` property on a `gw.api.database.Query` result produced an incorrect SQL statement and count result when `withDistinct(true)` was set on the query.

Search

PL-27359	A new parameter, `absolutePathToSortTmpDir`, has been added to the `batchload-config-`*`databaseBrand`*`.xml` file. This parameter sets the directory into which the sort program used by the free-text batch load command writes its intermediate files. This directory may be the same as or different from the directory specified by the parameter `absolutePathToWorkDir`. This latter parameter still specifies the directory into which the free-text batch load command itself writes its intermediate files.

Web - `ListViews`

ID	Description
PL-10908	Added the `groupedOnEnter` attribute to many cell-based PCF elements. If true, the `ListView` is grouped by this cell upon entering the page. Only one grouped cell is allowed at any given time, and it is applicable only when the column is sortable.
PL-27732	A new `height` attribute has been added to the `ListViewPanel` PCF element. This sets the vertical size, in pixels, of the list view. If the list data is taller than the specified height, then a vertical scroll bar appears within the list view. The height is calculated from the list view header toolbar. Header rows are fixed; the footer scrolls with the data. Note that this is currently an experimental attribute and may not function properly. You should fully test any use of this attribute.
PL-28466	Fixed an issue with columns reordering when switching between filtering views in a list view.
Web – UI/Runtime	
PL-27884	The following new methods were added to the Javascript `gw.api.Util` class: `getValue`, `setValue`, `getValues`, `setValues`. These allow you to get or set values of input elements. For example, you can use these in `TemplatePanel` and pass Gosu variables as arguments into these methods.
PL-28336	Fixed an issue that would occur when there was an action defined on a range cell.
PL-28749	Fixed a critical security vulnerability to persistent cross-site scripting attacks.
Work Queues	
PL-27301	Improved the server performance when selecting available work queue items.
PL-28696	Fixed an issue with `workItem` orphan detection during daylight saving time adjustment.
`XMLElement` **(and XSD types)**	
PL-27131	Gosu XML/XSD types now support use of circular `xs:include` references.

Known Issues and Limitations

This section describes known issues with this release of Guidewire PolicyCenter.

- "PolicyCenter Known Issues" on page 428
- "Product Designer Known Issues" on page 432
- "Studio/Platform Known Issues" on page 433

   Note: For maintenance releases, Guidewire often defers fixing configuration issues that require merging files during the upgrade. Workarounds to many of these issues are listed in the following sections. The goal of this policy is to make upgrades as straightforward as possible.

PolicyCenter Known Issues

Upgrading custom rating worksheet code from PolicyCenter 7.0.7 or 8.0.0 (DOC-9173)

Issue – The rating worksheet object model changes in PolicyCenter 8.0.1 affects custom rating worksheet code. The rating worksheet object is now `WorksheetContainer` entity.

Workaround – If you wrote custom rating worksheet code for PolicyCenter 7.0.7 or 8.0.0, you need to rewrite this code to use the new `WorksheetContainer` entity. For more information, see "Rating Worksheet Data" in the *PolicyCenter New and Changed Guide*.

Additional information about Converting Money to MonetaryAmount (DOC-9164)

Issue – The following statement in the *Upgrade Guide*, "Converting Money to MonetaryAmount" topic is not quite complete: "If you change the definition of your extension properties to use the `MonetaryAmount` datatype rather than the `Money` data type, upgrade will correctly convert your entity extension data."

Workaround – The more complete information is: "If you change the definition of your extension properties to use the `MonetaryAmount` datatype rather than the `Money` data type, upgrade will correctly convert your entity extension

data if you are coming from a single-currency instance (including any local extensions or business practices to manage currency) with the instance currency set properly. If you already manage multiple currencies, either explicitly or implicitly (such as through special notations on the account or the like), you must write an appropriate upgrade trigger as well as examine the supplied single-currency upgrade triggers."

Internal APIs accessible when programming in Java

Issue – When editing Java code for PolicyCenter, you can access internal API methods. Internal methods are defined with the `@InternalAPI` annotation. These internal methods mostly appear in the `gw.api.productmodel` package.
Workaround – When editing in Java, do not use API methods marked with the `@InternalAPI` annotation.

Gosu classes not intended for use (PC-23009)

Issue – Some Gosu classes are visible in Studio, but are not intended for use. You can distinguish these Gosu classes because they have no visibility annotations (neither `@Export` nor `@ReadOnly`) and they are not in a `gw` package.
Workaround – Do not use these methods in configuration. The methods are unsupported and may change or be withdrawn without notice.

Multicurrency list bill not directly supported

Issue – Using list bill features in a multicurrency environment is not supported in the base configuration integration between PolicyCenter and BillingCenter. PolicyCenter has the user interface elements for list bill. However, you will encounter errors if you attempt to use list bill in a multicurrency environment.
Workaround – Remove the user interface elements related to list bill or implement multicurrency list bill according to your specifications.

No file extension on rating worksheet data files (PC-22927)

Issue – The Extract Rating Worksheets batch process extracts the rating worksheet data to files and marks worksheets for purging. The files are in Gzip format but do not have a `.gz` file extension.
Workaround – Modify the `WorksheetExtractPlugin` plugin implementation so that it creates files with the `.gz` extension.

User input of certain currencies may not round as expected (PC-22773)

Issue – Certain currencies are formatted differently than U.S. dollars. For example, Japanese yen do not typically carry fractional parts. In some interactions where the user enters a yen-denominated amount with more than two decimal digits, PolicyCenter displays the error message, "...can have no more than 2 digits to the right of the decimal point."
Workaround – Guidewire is aware of this issue.

Reinsurance warnings not refreshed even after saving a draft (PC-21720)

Issue – When working with reinsurance, PolicyCenter displays validation errors and warnings both at the top and in the worksheet at the bottom of the screen depending on the action that caused validation to run (typically **Save**, or leaving the screen). Depending on the user flow, one or both of these areas may display stale messages, and the messages may not be the same.
Workaround – Refresh the messages in either or both locations before taking the action again.

Error when quoting after adding new location (PC-21656)

Issue – In the Commercial Auto line of business, users sometimes receive an error message after clicking **Quote** on the **Locations** screen.
Workaround – Click **Next** and then **Back** to navigate away from and back to the **Locations** screen before quoting.

Copy-data with an out-of-range date (PC-20279)

Issue – When working with the copy data feature, if you specify a specific date to copy data from and that date is out of range for the policy, PolicyCenter presents a warning that no applicable policy version could be found.

However, the date selector still indicates the requested date, not the date that PolicyCenter will use for the copy if you choose to proceed.

Workaround – As appropriate, modify the user interface to reset the date selector or to hide the ability to select data to copy.

SQL Query performance in free-text search (PC-18794)

Issue – There are open issues related to the performance of SQL query in free-text search. This occurs the first time that you load the PolicyCenter data into the Solr index database.

Workaround – Guidewire is aware of this issue.

Create Baseline button not visible after navigating to the Create Baseline screen (PC-17809)

Issue – When you start the **Create Baseline** step, PolicyCenter kicks of a batch job to create the baseline quotes. When you click **Cancel**, the cancel request is sent to the batch job, but the current quote may take some time to stop. If you click **Cancel**, go back and return to the **Create Baseline** screen. You may see the progress bar from the prior cancellation.

Workaround – Navigate to another screen and wait about a minute before returning to the **Create Baseline** screen.

Limitations of free-text search (PC-17496)

Issue – Free-text search does not support:
- Searching archived policies
- Searching unbound cancellation jobs

For non-U.S. or non-U.S. character set support, refer to the Apache Solr Language Analysis page on the Solr wiki. You can view this page at this address:

`http://wiki.apache.org/solr/LanguageAnalysis`

Free-text search does not remove the following policy periods from the search index database:
- Non-bound policy periods from a side-by-side multi-quote
- Withdrawn, non-renewed, not-taken or declined policy periods

Therefore, free-text search can become out-of-sync with these types of changes in PolicyCenter.

Workaround – Guidewire is aware of this issue.

Integrating PolicyCenter and BillingCenter with multicurrency may require additional configuration (PC-16094)

Issue – The base configurations of PolicyCenter 8.0.0 and BillingCenter 8.0.0 include an integration that permits single-currency interoperation of the two applications. To integrate PolicyCenter 8.0.0 and BillingCenter 8.0.0 in a multicurrency environment, you might need to extend the default implementation.

Workaround – Guidewire is aware of this issue.

Renewal processing for renewal conversion assumes BasedOn is populated (PC-15851)

Issue – The `startNewRenewal` method in `gw.webservice.pc.pc700.job.PolicyRenwalAPI` assumes the existence of a prior term or a `basedOn` version. Unless customized, this method can cause problems in certain convert-on-renewal scenarios. This method has been deprecated.

Workaround – If you are developing convert-on-renewal flows, use the new `gw.webservice.pc.pc700.job.startConversionRenewal` method.

BillingCenter integration: New invoice streams not created when issuing policy with alternate billing account on new account (PC-15309)

Issue – Assume an account has an existing invoice stream. When this account is used as an **Alt Billing Account** for new policies on new accounts, a new invoice stream is not created when the new policy's new invoice stream periodicity matches the invoice stream on the **Alt Billing Account**.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Guidewire is aware of this issue.

BillingCenter integration: Removing alternate billing account when payment instrument selected generates WsdlFault (PC-15308)

Issue – If an automatic payment instrument is selected when using an Alt Billing Account to create a new policy, PolicyCenter throws a `WsdlFault` if the Alt Billing Account gets removed.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Contact Guidewire Support for assistance with this issue.

BillingCenter integration: Automatic payment instrument is used to create invoice stream even if final selection is manual (PC-15307)

Issue – If an automatic payment instrument is setup and selected, the policy will create a new invoice stream with this payment instrument even if the user changes to Manual before issuing the policy.

This is an issue when integrating with BillingCenter 7.0.0 and later.

Workaround – Guidewire is aware of this issue.

PolicyCenter does not send renewal billing instruction to BillingCenter (PC-15082)

Issue – On receiving payment notification from BillingCenter, PolicyCenter tries to bind. If the policy is in a draft status, PolicyCenter does not bind and send the Renewal Billing Instruction to BillingCenter. The policy might be in draft status after an underwriter edited something on the policy. This is an issue with BillingCenter 3.0.1 and later.

Workaround – Guidewire is aware of this issue.

Must enter VIN when moving from quick quote multi-version or side-by-side to full application (PC-13099)

Issue – In a quick quote, you do not have to enter the VIN or other fields that are required for a full application submission. When converting a quick quote with multi-version or side-by-side quoting to full application, to resolve validation issues, you must enter the VIN for each of the versions. In a job with side-by-side quoting, this issue can occur with fields that are not base data.

Workaround – Add code that copies this type of data to the other versions.

When using parallel sessions, approvals made by underwriter are not updated for second user simultaneously viewing same work order (PC-12322)

Issue – Approvals made by an underwriter are not updated for a second user simultaneously viewing the same work order. This occurs in the following sequence of events:

- An agent processes a submission and issues arise
- The agent calls an underwriter, and remains in the policy
- The underwriter approves the issue or issues
- The agent refreshes the **Risk Analysis** screen by going to a different wizard step and returning to the **Risk Analysis** screen

After this sequence of events:

- The new approvals do not appear on the agent's screen
- However, the agent can bind or issue the policy (based on the new approvals by the underwriter)

Workaround – The agent can leave the policy and return to it. The approvals appear as expected. Alternatively, the agent can see the approvals by clicking the **Details** button when issuing the policy.

Answers persist on filtered out questions (PC-10645)

Issue – All answers to questions in question sets are normally persisted, even if the question itself is no longer relevant. When the question is no longer relevant, PolicyCenter removes it from the display. For example, in Personal Auto, the *Please provide the driver name and explain the conviction.* question is only visible if *Any drivers with convictions for moving traffic violations within the past 3 years? If 'Yes' please explain.* is positively responded

to. However, if a user enters an answer to that question and then changes the answer to the question that enabled it, the follow-up question is no longer displayed but the original answer is still saved along with the other answers.

Workaround – The answers to the question that is no longer displayed can be reset by invoking the following Gosu helper method when a wizard step exits. Check the edit mode before calling the method to clear orphaned answers. For example, add the following code to the `onExit` field in the wizard step that displays the question set:

```
if (openForEdit) { clearOrphanedAnswers() }
```

Add the following method in an enhancement of `PolicyLine`.

```
function clearOrphanedAnswers() {
  for (answer in this.Answers) {
    if (answer.Question.isQuestionVisible(this) != true ){
      answer.setValue(null)
    }
  }
}
```

Provide a way to filter out added and removed subentities in a policy differences tree (PC-7929)

Issue – Need a way to filter out the children of an added or removed entity.

Workaround – Guidewire is aware of this issue.

Product Designer Known Issues

PolicyCenter does not start in single currency display mode if the product model is configured to use multiple currencies (DOC-8196)

Issue – Product Designer allows users to configure multicurrency features without regard for the setting of the `MultiCurrencyDisplayMode` in the PolicyCenter `config.sys` file. If you configure multiple currencies in a product model object, and the value of the `MultiCurrencyDisplayMode` property is `SINGLE`, PolicyCenter does not start.

For example, if you add clauses that specify multiple coverage currencies and the PolicyCenter `MultiCurrencyDisplayMode` is set to `SINGLE`, the server fails to start and displays messages similar to the following:

```
ERROR ... PolicyLinePattern : ... [field : AvailableCoverageCurrencies] : - Multiple Coverage
Currencies defined for product not allowed in single Currency mode
ERROR ... [ExclusionPattern] : ... [DirectCovTermPattern] : ... [LimitsSet] : - The pattern specifies
limits for different currencies in SingleCurrencyMode
```

Workaround – If you intend to deploy a product model on PolicyCenter servers that are not configured for multiple currency display mode, do not configure multiple currencies in any part of the product model.

PolicyCenter exception when synchronizing the product model while a submission is in progress (PC-21708)

Issue – PolicyCenter throws a `ClassCastException` and stops responding if you synchronize the product model from Product Designer while a submission policy transaction is in progress. To recover from this state, you must restart the PolicyCenter server.

Workaround – Do not synchronize the product model while a submission policy transaction is in progress. Instead, navigate to a screen outside any policy transaction, such as the **Account Summary** screen, before synchronizing the product model.

Alternatively, edit `config.xml` to enable internal debug tools, and then after synchronizing the product model and prior to performing any operation in PolicyCenter, press `Alt+Shift+L` to reload PCF files.

After switching change lists that are configured to use different languages, some fields still show values in the language of the previous change list (BCC-2017)

Issue – If you do the following steps:

1. View the product model in one change list while using the language specified as the default language in Product Designer's language.properties file.
2. Change to a different language.
3. Switch to a different change list.
4. Continue to view the product model in the new change list.

Some product model values continue to appear in the default language, instead of appearing in the language you selected prior switching change lists.

Workaround – Guidewire is aware of this issue.

Cannot use Product Designer to edit an extended object with no base object (BCC-1759)

Issue – If you attempt to edit a part of the product model that is based on an entity extension file (.etx), and that extension has no base entity file (.eti), Product Designer displays errors and refuses to display the contents of the file.

Although Guidewire best practices recommend that you never create an entity file extension (`.etx` file) when no base entity (`.eti` file) exists, Guidewire Studio did not enforce this recommendation. However, the current version of Product Designer does not allow the use of an entity extension without a corresponding entity base file. If such an entity extension exists, Product Designer displays a **NullPointerException** when you attempt to access the affected objects. You then can use the Back command in the browser to view other pages, but you cannot configure the affected objects.

Workaround – Locate all `.etx` files that do not have corresponding `.eti` files, and rename them with an `.eti` extension.

Cannot edit a Series audit schedule (BCC-1704)

Issue – Two types of checking audits are in common use: *single* checking audits and *series* checking audits. The base configuration of PolicyCenter 8.0.1 does not support either type of checking audit. However, you can configure checking audits through configuration, which is made easier if you use the PolicyCenter Checking Audits Accelerator.

Product Designer provides the user interface that enables you to edit single checking audits when they have been enabled through customer configuration. However, it does not provide a user interface to edit series checking audits.

Workaround – Follow the instructions in the *PolicyCenter Checking Audits - Solution Guide* to manually create and edit an appropriate XML file with the required checking audit series patterns.

When using Google Chrome, Changes and Help panels "bounce" open and closed (BCC-926)

Issue – Clicking the **Changes** button or **Help** button opens a panel on the right side of the Product Designer window. In some cases, when using Google Chrome, the panels open, then immediately close, then open again. This "bouncing" behavior has been observed when certain Chrome extensions are active, and appears to be related to certain mail notification and ad blocking extensions.

Workaround – Disable browser extensions one at a time until the "bouncing" window behavior stops, or use a different browser.

Studio/Platform Known Issues

Problem with regen-java-api command and JAR files (PL-16351)

Issue – If you run the `gwpc regen-java-api` command, PolicyCenter creates a `PolicyCenter`/`java-api/lib` directory and puts JAR files for Java development in this directory. However, PolicyCenter does not always perform this task correctly, resulting in some of the generated JAR files not being copied to this directory.

Workaround – For missing library files, copy or import into your Java development environment the missing library files from the following directory:

```
PolicyCenter/admin/lib
```

Copy them into the following directory:

```
PolicyCenter/java-api/lib
```

Renaming method or property throws ParseResultsException (PL-16633)

Issue – If you rename a property or a method or you change a method signature, and a workflow references that property or method in a Gosu field, PolicyCenter throws `ParseResultsException`. This is the intended behavior.

Workaround – Restart the workflow engine. To do so:

1. Log in to PolicyCenter using an administrative account.
2. Access **Internal Tools→Reload**.
3. Click **Reload Workflow Engine**.

Javadoc command does not generate index file in expected location (PL-27679)

Issue – When you run `gwpc regen-java-api`, an `index.html` file is not created in *PolicyCenter*`/java-api/doc`.

Workaround – The command now generates Javadoc JAR files in *PolicyCenter*`/java-api/doc`. To view the Javadoc, add the JAR files to the Studio project.

Gosu does not automatically downcast if the left side of the typeis or typeof expression uses deprecated members (PL-27724)

Issue – To improve readability of your Gosu code, Gosu automatically downcasts after a `typeis` expression if the type is a subtype of the original type. This is particularly valuable for `if` statements and similar Gosu structures. For example, if a variable has type `Object`, you can use code such as:

```
if( x typeis String ) {
   length // NOTE: length is a property on String, but *not* on Object.
}
```

In this release, Gosu does not automatically downcast if the left side of the `typeis` or `typeof` expression uses deprecated members. This may result in new compilation errors.

Workaround – To fix these compiler errors, explicitly downcast with the `as` keyword before you access properties or methods on the subtype but not on the original type. For example, suppose a property called `Dep` is deprecated:

```
if (x.Dep typeis ExampleType) {
   return (x.Dep as ExampleType).PropertyOnExampleSubtype
}
```

Long text in table cells can add white space to the right of the page (PL-28288)

Issue – In an editable list view, extremely long text entered in a single cell can cause additional white space on the right side of the page. Long text is text that occupies approximately the entre width of the screen. This issue occurs primarily in Chrome.

Workaround – If you expect users to enter large amounts of text into the cells of a column, configure the column to support text wrapping.

Build tool regen-java-api can fail with -Ddeprecated=true (PL-28992)

Issue – If you run the `gwpc regen-java-api` command with the flag `-Ddeprecated=true`, the command can fail.

Workaround – Specify `dev-deploy` as the value of the `depends` attribute, as follows:

1. Open the `build.xml` file in `PolicyCenter/modules/ant` in an editor.

2. Find the entry that starts as follows:

```
<target name="regen-java-api" depends="init"
```

3. Change the value of the `depends` attribute to `dev-deploy`, as follows:

```
<target name="regen-java-api" depends="dev-deploy"
```

4. Save the file and then run the `gwpc regen-java-api` command.

Administrative command-line tools cannot refresh WSDL (PL-29021)

Issue – Some administrative command-line tools rely on web service implementation classes, such as `MaintenanceToolsAPI.gs`. Source files for these classes use the `@Export` annotation, which allows you to edit the file. In this release, the administrative command-line tools cannot refresh the WSDL for these classes. Any change to the web service implementation class that changes the WSDL can prevent the administrative tools from working. Therefore, the only changes you can make to these classes are changes that do not affect the WSDL. For example, you can add `@WsiPermission` annotations to change the permissions without changing the WSDL.

Workaround – Guidewire is aware of this issue.

Security vulnerability – reflected XSS (PL-29052)

Issue – There is a non-persistent cross-site scripting (reflected XSS) vulnerability. Unlike other XSS types, this vulnerability does not permit privilege escalation and does not propagate easily to other users.

Workaround – Guidewire is aware of the issue. Strong email filtering with phishing/malware detection is an effective defense against this type of attack. Contact Customer Support for more information.

Free-text Search indexes cannot be upgraded from 7.0 to 8.0 (PL-29062)

Issue – To upgrade free-text search in PolicyCenter from 7.0 to 8.0, you install the newer version of the Guidewire Solr Extension in 8.0 on top of version from 7.0. Because the PolicyCenter data model also changes, free-text search indexes from 7.0 become obsolete. Guidewire does not support migrating free-text search indexes from 7.0. to 8.0.

Workaround – For Oracle, upgrade the Guidewire Solr Extension by following instructions in the *Upgrade Guide*. Then, run the new free-text batch load command to rebuild the indexes based on the new data model, after you complete your database upgrade. For SQL Server, Guidewire is aware of potential performance problems running the free-text batch load command after you complete your database upgrade.

Chrome browser cannot display product documentation in HTML format (DOC-7251)

Issue – If you use the Google Chrome browser, you can view the HTML Guidewire product documentation only if it is served by an HTTP server using the `http://` protocol. The Chrome browser cannot load HTML product documentation from your local disk by using the `file://` protocol.

Workaround – Use a different browser, such as Microsoft Internet Explorer. If you use the HTML documentation only to access the PDF files, you can open them directly in the `pdf` subdirectory of the `doc` directory.

New mechanism for reloading Gosu classes (DOC-8218)

Issue – In past releases, you could modify your PCF files and Gosu classes in Studio, and then reload the changes into the running server by pressing Alt+Shift+L in the application user interface. This shortcut no longer loads Gosu classes.

Workaround – have the server reload your Gosu classes, in Studio, click **Build→Make Project**. When Studio is finished compiling your project, the changes will be loaded. You can also restart your server to load the Gosu classes.

Guidewire PolicyCenter 8.0.0 Release Notes

Release 8.0.0
Release Notes Update

IMPORTANT These release notes replace the release notes that were included with the official product release. Please disregard the earlier version of the release notes.

Overview

These release notes contain the following:
- "Release Information" on page 437
- "Critical Action Required" on page 438
- "Installing This Release" on page 438
- "Support" on page 438
- "Issues and Major Changes" on page 438
- "Known Issues and Limitations" on page 440

Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

IMPORTANT If you skipped one or more upgrade releases to PolicyCenter, be sure to read the release notes for those releases to learn about changes and fixed issues.

Version Number

This release of Guidewire PolicyCenter is 8.0.0.

- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 7.0.1 or later.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 7.0.0 or later.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 7.0.0.

Critical Action Required

An issue that potentially causes data corruption was discovered in the quote purging batch processes. This issue can occur whether these processes have been intentionally enabled or not. You must explicitly disable the quote purging batch processes. For guidance and instructions, see Knowledge Base article *"Potential for data corruption when using Purge Batch Processes"* on the Guidewire Resource Portal.

This issue will be addressed in a future release.

Installing This Release

Refer to the following:

- The *PolicyCenter Installation Guide* for general installation information
- Prior PolicyCenter release notes for any versions that you have skipped

Support

For assistance, visit the Guidewire Community.

Guidewire Customers

- https://community.guidewire.com

Guidewire Partners

- https://partner.guidewire.com

Issues and Major Changes

This section contains issues and major changes that may affect your installation.

Money properties not upgraded to MonetaryAmount properties

PolicyCenter upgrade cannot automatically convert the `Money` data type to the `MonetaryAmount` data type. If you created entity extensions, the upgrade process will not upgrade your extensions that include properties that use the `Money` data type.

Before you upgrade, you can manually update any extension properties that use the `Money` data type.

If you change the definition of your extension properties to use the `MonetaryAmount` datatype rather than the `Money` data type, upgrade will correctly convert your entity extension data. Define the `MonetaryAmount` property as follows:

- The name of the new `MonetaryAmount` property is the same as the name of the `Money` property
- If the old `Money` property had a `columnName` attribute defined as something other than the `Money` property name, use that old `Money.columnName` as the name of the new `MonetaryAmount.amountColumnName` attribute.
- Set `scaleToCurrency` to `true` unless you have a requirement to do otherwise.
- Set the `soapNullOk` attribute to `true`

The following examples show how you must redefine `Money` properties in your extensions to `MonetaryAmount` properties before you proceed with upgrade:

Old Total

```
<column
    name="Total"
    type="money"/>
```

New Total

```
<monetaryamount
    name="Total"
    amountColumnName="Total"/>
```

Old Total where name and columnName differ

```
<column
    name=" Total"
    columnName="totalColumn"
    type="money"/>
```

New Total

```
<monetaryamount
    name="Total"
    amountColumnName="totalColumn"/>
```

After you complete this configuration step, your database will upgrade successfully.

Base PCF File Changes

The link below requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 7.0.7 to 8.0.0

To view a report of the changes to the base PCF files, *click here*.

Base Resource Changes

The link below requires that the `ReleaseNotes_files` directory be on your local disk in the same directory as this release notes file.

PolicyCenter release 7.0.7 to 8.0.0

To view a report of the changes to the base resources, *click here*.

Changes in This Release Provided in Upgrade Diff Report

Guidewire provides a report detailing certain differences between the current release and your prior release. This report describes changes in display keys, entities, typelists, and the Gosu API. To obtain your custom Upgrade Diff Report, visit the Guidewire Community.

Known Issues and Limitations

This section describes known issues with this release of Guidewire PolicyCenter.

- "PolicyCenter Known Issues" on page 440
- "Product Designer Known Issues" on page 445
- "Studio/Platform Known Issues" on page 446

PolicyCenter Known Issues

Problem with Solr Batch Loader (PC-21908, PC-21895)

Issue – The free-text batch load command does not function correctly in this release.

Workaround – Do not use the free-text batch load command. The administrator can load the Solr index in PolicyCenter by going to **Internal Tools→Free-text Search** and then clicking **Sync policy index**.

This workaround is only intended for development mode in PolicyCenter, and not for production mode with a large number of policies. The Solr batch loader is required for production mode with a large number of policies.

Ceding potentially not recalculated when agreements change after bind/issue (PC-21863)

Issue – If you change the reinsurance agreements associated with a bound and/or issued policy, the Premium Ceding job recalculates the ceding. However, validation warnings added in PolicyCenter 8.0.0 (about agreements not ceding to capacity) prevent successful completion of the recalculation. If you added validation warnings to reinsurance, this problem may also occur with those warnings.

Workaround – Remove the validation warnings to permit the recalculation to complete successfully.

Long UWIssue.Issuekey can prevent binding, issuance, and other operations on policies (PC-21852)

Issue – Observed with certain policies and reinsurance agreements, the user cannot bind or issue the policy because the identifying `Issuekey` calculated is too large for database storage. This can also occur with field-implemented `UWIssue` objects if the `Issuekey` is too long.

In the base configuration, the user sees the following error:

...is not formatted correctly for the field "Issuekey": Exceeds the maximum length of 255 characters

Workaround – Use an `Issuekey` that is shorter than the database limitation but still uniquely identifies the conditions behind the `UWIssue` instance.

In the base configuration, the `FP64` fingerprint method generates a key that is virtually unique from the much longer string generated by the code. You should evaluate the suitability of the FP64 or any other algorithm for your specific purpose.

To resolve the issue in the base configuration, in `RIRiskEnhancement.gsx`, add to the definition of the `UWIssueKey` getter method:

```
uses gw.util.fingerprint.FP64

 property get UWIssueKey() : String {
  var fingerPrint = new FP64()
var includedAttachments = this.Attachments.AttachmentsForRiskCeding.sortBy
        (\ elt -> elt.Agreement.AgreementNumber)
  includedAttachments.each( \ elt -> {
    fingerPrint.extend(elt.Agreement.AgreementNumber)
  })
  return this.Reinsurable.UWIssueKey + fingerPrint.toString()
}
```

Error when quoting after adding new location (PC-16058, PC-21656)

Issue – In the Commercial Auto line of business, users sometimes receive an error message after clicking **Quote** on the **Locations** screen. This has been observed after adding a new location in a policy change transaction.

Workaround – Click **Next** and then **Back** to navigate away from and back to the **Locations** screen before quoting.

Free-text Search: May need to recreate the Solr Index when upgrading to PolicyCenter 8.0 (PC-21550)

Issue – The setting of the Solr parameter, `OmitTermFreqAndPositions`, may require alternate steps to upgrade the Solr index from 3.6 to 4.0.

In PolicyCenter 7.0.6 and 7.0.7, free-text search uses Solr 3.6, in which the default setting of the `OmitTermFreqAndPositions` parameter is `true`. In PolicyCenter 8.0, free-text search uses Solr 4.0, in which the default setting of `OmitTermFreqAndPositions` is `false`. PolicyCenter 8.0 requires that this parameter be set to `false`.

Workaround – If `OmitTermFreqAndPositions` was set to `false` in Solr 3.6 prior to deploying in production, then you can perform the free-text search Solr upgrade as described in the *New and Changed Guide* for PolicyCenter 8.0.0.

If `OmitTermFreqAndPositions` was set to `true` in Solr 3.6, then the Solr index can not be upgraded. Instead, you must recreate the Solr index by re-running the SQL batch loader. For more information, see the Solr batch loader documentation.

DBNullConstraintException while creating new organization (PC-21541)

Issue – If you create a new organization and your current language setting is not the default language, PolicyCenter generates a `DBNullConstraintException`.

Workaround – Set the default language to the language needed by the user creating organizations. Then restart the server.

Taxes improperly calculated for mixed-currency policies in PolicyCenter base lines of business (PC-20908)

Issue – In the lines of business in the base configuration, PolicyCenter calculates taxes by summing the as-rated costs without currency conversion. Therefore, tax calculations are incorrect if the coverage currency of one or more coverages is different than the settlement currency.

Tax calculations for single-currency policies are correct.

Workaround – Configure the tax calculation to include multiple currencies by doing either or both of the following, as appropriate:

- Calculate taxes based on the total premium in the settlement currency.
- Calculate taxes on a coverage-by-coverage basis.

Copy-data with an out-of-range date (PC-20279)

Issue – When working with the copy data feature, if you specify a specific date to copy data from and that date is out of range for the policy, PolicyCenter presents a warning that no applicable policy version could be found. However, the date selector still indicates the requested date, not the date that PolicyCenter will use for the copy if you choose to proceed.

Workaround – As appropriate, modify the user interface to reset the date selector or to hide the ability to select data to copy.

Persistence model for rating worksheets impacts performance (PC-20277)

Issue – In the current implementation, rating worksheets impact the performance of the quote significantly and can take up a large amount of disk space.

Workaround – In this release, use rating worksheets only in development and test.

SQL Query performance in free-text search (PC-18794)

Issue – There are open issues related to the performance of SQL query in free-text search. This occurs the first time that you load the PolicyCenter data into the Solr index database.

Workaround – Guidewire is aware of this issue.

Create Baseline button not visible after navigating to the Create Baseline screen (PC-17809)

Issue – When you start the **Create Baseline** step, PolicyCenter kicks of a batch job to create the baseline quotes. When you click **Cancel**, the cancel request is sent to the batch job, but the current quote may take some time to stop. If you click **Cancel**, go back and return to the **Create Baseline** screen. You may see the progress bar from the prior cancellation.

Workaround – Navigate to another screen and wait about a minute before returning to the **Create Baseline** screen.

Exception in impact testing when creating baselines (PC-17594)

Issue – In impact testing, PolicyCenter throws an exception when creating baselines policy periods. This occurs if the number of workers for `ImpactTestingTestRunWorkQueue` is `10` in `work-queue.xml`.

Impact testing is a feature of Guidewire Rating Management.

This problem occurs if you do the following:

1. In PolicyCenter, go to **Rating→Impact Testing** and advance to the **Create Baseline** screen.
2. Click **Create Baselines**. PolicyCenter creates test policy periods, and begins quoting baseline and test policy periods.

   PolicyCenter throws an exception. You can view the exception in the PolicyCenter command window. The command window displays messages stating that quoting the policy periods failed. However, all policy periods have been quoted.

Workaround – In Studio, open `work-queue.xml` and reduce the number of workers for the `ImpactTestingTestRunWorkQueue` to `1`.

Limitations of free-text search (PC-17496)

Issue – Free-text search does not support:
- Searching archived policies
- Searching unbound cancellation jobs

For non-U.S. or non-U.S. character set support, refer to the Apache Solr Language Analysis page on the Solr wiki. You can view this page at this address:

`http://wiki.apache.org/solr/LanguageAnalysis`

Free-text search does not remove the following policy periods from the search index database:
- Non-bound policy periods from a side-by-side multi-quote
- Withdrawn, non-renewed, not-taken or declined policy periods

Therefore, free-text search can become out-of-sync with these types of changes in PolicyCenter.

Workaround – Guidewire is aware of this issue.

Cancellation does not withdraw future renewal (PC-16312)

Issue – Cancellation does not withdraw a renewal scheduled in the future and does not raise underwriting issue. On the renewal, the **Renewal Notification Date** (`RenewalNotifDate`) has already been set – the user has been offered the Renewal, thus it cannot be withdrawn. This can occur in the following sequence:

1. There is a bound policy with a renewal term in renewing status with an effective date in the future.
2. Cancel the policy effective between the effective date of the bound policy and the effective date of the renewal.
3. Bind the cancellation.

   When binding the cancellation, PolicyCenter displays a message that it will withdraw the renewal work order.

   After binding the cancelation, there is a canceled policy and an activity that the renewal was withdrawn. However, the renewal work order remains in the **Work Orders in Progress** list view with a status of **Renewing**.

Workaround – The workaround for this issue depends on your renewal process. Guidewire recommends that an underwriting issue be created, notifying the underwriter of the situation. At this point, the underwriter can choose the appropriate action to take. Alternately, if your renewal process allows renewals to be withdrawn after the customer has been offered a renewal, `RenewalNotifDate` can be set to null, and the renewal can then be withdrawn.

Integrating PolicyCenter and BillingCenter with multicurrency may require additional configuration (PC-16094)

Issue – The base configurations of PolicyCenter 8.0.0 and BillingCenter 8.0.0 include an integration that permits single-currency inter-operation of the two applications. To integrate PolicyCenter 8.0.0 and BillingCenter 8.0.0 in a multicurrency environment, you may need to extend the default implementation.
Workaround – Guidewire is aware of this issue.

Renewal processing for renewal conversion assumes BasedOn is populated (PC-15851)

Issue – The `startNewRenewal` method in `gw.webservice.pc.pc700.job.PolicyRenwalAPI` assumes the existence of a prior term or a `basedOn` version. Unless customized, this method can cause problems in certain convert-on-renewal scenarios. This method has been deprecated.
Workaround – If you are developing convert-on-renewal flows, use the new `gw.webservice.pc.pc700.job.startConversionRenewal` method.

BillingCenter integration: New invoice streams not created when issuing policy with alternate billing account on new account (PC-15309)

Issue – Assume an account has an existing invoice stream. When this account is used as an **Alt Billing Account** for new policies on new accounts, a new invoice stream is not created when the new policy's new invoice stream periodicity matches the invoice stream on the **Alt Billing Account**.
This is an issue when integrating with BillingCenter 7.0.0 and later.
Workaround – Guidewire is aware of this issue.

BillingCenter integration: Removing alternate billing account when payment instrument selected generates WsdlFault (PC-15308)

Issue – If an automatic payment instrument is selected when using an Alt Billing Account to create a new policy, PolicyCenter throws a `WsdlFault` if the Alt Billing Account gets removed.
This is an issue when integrating with BillingCenter 7.0.0 and later.
Workaround – Contact Guidewire Support for assistance with this issue.

BillingCenter integration: Automatic payment instrument is used to create invoice stream even if final selection is manual (PC-15307)

Issue – If an automatic payment instrument is setup and selected, the policy will create a new invoice stream with this payment instrument even if the user changes to Manual before issuing the policy.
This is an issue when integrating with BillingCenter 7.0.0 and later.
Workaround – Guidewire is aware of this issue.

PolicyCenter does not send renewal billing instruction to BillingCenter (PC-15082)

Issue – On receiving payment notification from BillingCenter, PolicyCenter tries to bind. If the policy is in a draft status, PolicyCenter does not bind and send the Renewal Billing Instruction to BillingCenter. The policy might be in draft status after an underwriter edited something on the policy. This is an issue with BillingCenter 3.0.1 and later.
Workaround – Guidewire is aware of this issue.

Must enter VIN when moving from quick quote multi-version or side-by-side to full application (PC-13099)

Issue – In a quick quote, you do not have to enter the VIN or other fields that are required for a full application submission. When converting a quick quote with multi-version or side-by-side quoting to full application, to resolve

validation issues, you must enter the VIN for each of the versions. In a job with side-by-side quoting, this issue can occur with fields that are not base data.

Workaround – Add code that copies this type of data to the other versions.

When using parallel sessions, approvals made by underwriter are not updated for second user simultaneously viewing same work order (PC-12322)

Issue – Approvals made by an underwriter are not updated for a second user simultaneously viewing the same work order. This occurs in the following sequence of events:

- An agent processes a submission and issues arise
- The agent calls an underwriter, and remains in the policy
- The underwriter approves the issue or issues
- The agent refreshes the **Risk Analysis** screen by going to a different wizard step and returning to the **Risk Analysis** screen

After this sequence of events:

- The new approvals do not appear on the agent's screen
- However, the agent can bind or issue the policy (based on the new approvals by the underwriter)

Workaround – The agent can leave the policy and return to it. The approvals appear as expected. Alternatively, the agent can see the approvals by clicking the **Details** button when issuing the policy.

Answers persist on filtered out questions (PC-10645)

Issue – All answers to questions in question sets are normally persisted, even if the question itself is no longer relevant. When the question is no longer relevant, PolicyCenter removes it from the display. For example, in Personal Auto, the *Please provide the driver name and explain the conviction.* question is only visible if *Any drivers with convictions for moving traffic violations within the past 3 years? If 'Yes' please explain.* is positively responded to. However, if a user enters an answer to that question and then changes the answer to the question that enabled it, the follow-up question is no longer displayed but the original answer is still saved along with the other answers.

Workaround – The answers to the question that is no longer displayed can be reset by invoking the following Gosu helper method when a wizard step exits. Check the edit mode before calling the method to clear orphaned answers. For example, add the following code to the `onExit` field in the wizard step that displays the question set:

```
if (openForEdit) { clearOrphanedAnswers() }
```

Add the following method in an enhancement of `PolicyLine`.

```
function clearOrphanedAnswers() {
  for (answer in this.Answers) {
    if (answer.Question.isQuestionVisible(this) != true ){
      answer.setValue(null)
    }
  }
}
```

Provide a way to filter out added and removed subentities in a policy differences tree (PC-7929)

Issue – Need a way to filter out the children of an added or removed entity.

Status – Guidewire is aware of this issue.

Product Designer Known Issues

PolicyCenter does not start in single currency display mode if the product model is configured to use multiple currencies (DOC-8196)

Issue – Product Designer allows users to configure multicurrency features without regard for the setting of the `MultiCurrencyDisplayMode` in the PolicyCenter `config.sys` file. If you configure multiple currencies in a product model object, and the value of the `MultiCurrencyDisplayMode` property is `SINGLE`, PolicyCenter does not start.

For example, if you add clauses that specify multiple coverage currencies and the PolicyCenter `MultiCurrencyDisplayMode` is set to `SINGLE`, the server fails to start and displays messages similar to the following:

```
ERROR ... PolicyLinePattern : ... [field : AvailableCoverageCurrencies] : - Multiple Coverage
Currencies defined for product not allowed in single Currency mode
ERROR ... [ExclusionPattern] : ... [DirectCovTermPattern] : ... [LimitsSet] : - The pattern specifies
limits for different currencies in SingleCurrencyMode
```

Requirement – If you intend to deploy a product model on PolicyCenter servers that are not configured for multiple currency display mode, do not configure multiple currencies in any part of the product model.

PolicyCenter exception when synchronizing the product model while a submission is in progress (PC-21708)

Issue – PolicyCenter throws a `ClassCastException` and stops responding if you synchronize the product model from Product Designer while a submission policy transaction is in progress. To recover from this state, you must restart the PolicyCenter server.

Workaround – Do not synchronize the product model while a submission policy transaction is in progress. Instead, navigate to a screen outside any policy transaction, such as the **Account Summary** screen, before synchronizing the product model.

Alternatively, edit `config.xml` to enable internal debug tools, and then after synchronizing the product model and prior to performing any operation in PolicyCenter, press `Alt+Shift+L` to reload PCF files.

Cannot use Product Designer to edit an extended object with no base object (BCC-1759)

Issue – If you attempt to edit a part of the product model that is based on an entity extension file (.etx), and that extension has no base entity file (.eti), Product Designer displays errors and refuses to display the contents of the file.

Although Guidewire best practices recommend that you never create an entity file extension (`.etx` file) when no base entity (`.eti` file) exists, Guidewire Studio did not enforce this recommendation. However, the current version of Product Designer does not allow the use of an entity extension without a corresponding entity base file. If such an entity extension exists, Product Designer displays a **NullPointerException** when you attempt to access the affected objects. You then can use the Back command in the browser to view other pages, but you cannot configure the affected objects.

Workaround – Locate all `.etx` files that do not have corresponding `.eti` files, and rename them with an `.eti` extension.

When using Google Chrome, Changes and Help panels "bounce" open and closed (BCC-926)

Issue – Clicking the **Changes** button or **Help** button opens a panel on the right side of the Product Designer window. In some cases, when using Google Chrome, the panels open, then immediately close, then open again. This "bouncing" behavior has been observed when certain Chrome extensions are active, and appears to be related to certain mail notification and ad blocking extensions.

Workaround – Disable browser extensions one at a time until the "bouncing" window behavior stops, or use a different browser.

Studio/Platform Known Issues

Chrome browser cannot display product documentation in HTML format (DOC-7251)

Issue – If you use the Google Chrome browser, you can view the HTML Guidewire product documentation only if it is served by an HTTP server using the `http://` protocol. The Chrome browser cannot load HTML product documentation from your local disk by using the `file://` protocol.

Workaround – Use a different browser, such as Microsoft Internet Explorer. If you use the HTML documentation only to access the PDF files, you can open them directly in the `pdf` subdirectory of the `doc` directory.

GX model generated XSD cannot be parsed by JAXB (PL-13598)

Issue – XSD generated by the GX model cannot be parsed by JAXB.

Workaround – Add JAXB annotation elements to the XSD to specify the necessary metadata, such as class names, to enable JAXB to generate the Java class files. Contact Guidewire Support for a sample XSD file that is annotated this way.

Problem with regen-java-api command and JAR files (PL-16351)

Issue – If you run the `gwXX regen-java-api` command, PolicyCenter creates a `PolicyCenter/java-api/lib` directory and puts JAR files for Java development in this directory. However, PolicyCenter does not always perform this task correctly, resulting in some of the generated JAR files not being copied to this directory.

Workaround – For missing library files, copy or import into your Java development environment the missing library files from the following directory:

```
PolicyCenter/admin/lib
```

Copy them into the following directory:

```
PolicyCenter/java-api/lib
```

Renaming method or property throws ParseResultsException (PL-16633)

Issue – If you rename a property or a method or you change a method signature, and a workflow references that property or method in a Gosu field, PolicyCenter throws `ParseResultsException`. This is the intended behavior.

Workaround – Restart the workflow engine. To do so:

1. Log in to PolicyCenter using an administrative account.
2. Access **Internal Tools→Reload**.
3. Click **Reload Workflow Engine**.

Gosu class can override @InternalAPI methods when a PublishInGosu java class is subclassed (PL-18217)

Issue – When a Gosu class extends a Java class, it is possible for the Gosu class to override methods in the Java class that are marked as `@InternalAPI`. These overrides could lead to unpredictable behavior.

Workaround – Do not override methods marked as `@InternalAPI` when creating a Gosu class that subclasses a Java class.

Client-side document production scripts cannot be customized in this release (PL-21502)

Issue – In previous releases, client-side document production scripts downloaded by the ActiveX Document Assistant could be customized by modifying JScript files in the web application and removing the cached copies from a `temp` directory on all user computers. In this release, the ActiveX control was replaced by a signed Java Web Start (JWS) application. Because client-side scripts are encapsulated in the signed JWS application, you cannot change the scripts in this release.

Workaround – Guidewire is aware of this issue.

New inbound integration system requires additional configuration information (PL-25227)

Issue – This release includes a new inbound integration system, which is documented in the PolicyCenter Integration Guide. Additional configuration information is necessary to use the new API in this release.

Workaround – Contact Guidewire Customer Support for details.

JBoss 6 application server unable to start (PL-27203)

Issue – JBoss 6 generates an exception when it is deployed with PolicyCenter 8.0.0.

Workaround – Remove or comment out the tag `<resource-ref>` in the file `web.xml`.

Multiple rule folders are created during a configuration upgrade (PL-27338)

Issue – Multiple rule folders are created if you repeatedly run the configuration upgrade tool followed by the `clean` command.

Workaround – Restore the innermost rules folder to its proper location.

1. Copy the innermost folder to a temporary location.
2. Remove all the nested folders from the original location.
3. Copy back the innermost folder from the temporary location to its proper location.
4. Make sure the folder is named correctly (`PolicyCenter8_0_0Rules`).

List view columns that are initially not visible and then set to visible always appear on the right side (PL-27556)

Issue – In PolicyCenter 8.0, you can reorder the columns of list views or change their width. These settings are then saved by PolicyCenter in the web browser for each list view, so the same ordering and width can be used when the page is revisited.

However, a layout of a list view can change due to differences in data or because the server configuration has changed. New columns added to the list view since the last time the user visited show up on the far right side of the list view, and not in the order specified in the PCF file. This can be confusing, especially when you must scroll the page to the right to see the new columns.

This behavior can also occur when there are two modes of a PCF page containing list views of a similar structure, but with a different ordering of columns. The order and width settings can be applied to the wrong list view in this case, and columns can appear in a different order than intended.

Workaround – If list view columns seem to be missing, first scroll to the right to see if they are there. To correct the order of list view columns, you can reset your layout preferences to restore the default list ordering and widths. To do this, select **Options** →**Clear Layout Preference** in Guidewire PolicyCenter.

Javadoc command does not generate index file in expected location (PL-27679)

Issue – When you run `gwXX regen-java-api`, an `index.html` file is not created in *PolicyCenter*/java-api/doc.

Workaround – The command now generates Javadoc JAR files in *PolicyCenter*/java-api/doc. To view the Javadoc, add the JAR files to the Studio project.

Command to generate data dictionary fails if maxSPVInclusions option is specified (PL-27693)

Issue – The data dictionary is not generated when you run the `gwXX regen-dictionary` command with the `maxSPVInclusions` option.

Workaround – Do not use the `maxSPVInclusions` option with this command.

Gosu does not automatically downcast if the left side of the typeis or typeof expression uses deprecated members (PL-27724)

Issue – To improve readability of your Gosu code, Gosu automatically downcasts after a `typeis` expression if the type is a subtype of the original type. This is particularly valuable for if statements and similar Gosu structures. For example, if a variable has type `Object`, you can use code such as:

```
if( x typeis String ) {
  length // NOTE: length is a property on String, but *not* on Object.
}
```

In this release, Gosu does not automatically downcast if the left side of the typeis or typeof expression uses deprecated members. This may result in new compilation errors.

Workaround – To fix these compiler errors, explicitly downcast with the "as" keyword before you access properties or methods on the subtype but not on the original type. For example, suppose a property called Dep is deprecated:

```
if (x.Dep typeis ExampleType) {
  return (x.Dep as ExampleType).PropertyOnExampleSubtype
}
```

Upgrade trigger for postOnChange on PCF widgets is not working in some cases (PL-27755)

Issue – In PCF files for some widgets, the PolicyCenter 8.0.0 upgrade tool does not upgrade the postOnChange property to the new syntax for this property. This problem can occur with any widgets that you have added that use postOnChange and with any widgets in the base configuration for which you have set postOnChange.

Workaround – After running the PolicyCenter 8.0.0 upgrade tool:

1. Find all instances of widgets that did not have their postOnChange properties converted. For example, search the *PolicyCenter*/modules/configuration/config/web/pcf folder and subfolders for occurrences of postOnChange=. The instances that you will find are either widgets that need correction or widgets that are commented out (disabled.) There is no need to make the correction on disabled widgets, although there is also no harm in doing so.

2. For each widget that needs correction, open its PCF file in an XML editor and change the following old syntax to the new syntax:
   • Old syntax example:

3. To verify that you have corrected all instances, open Guidewire Studio and navigate in the **Project** window to **configuration**→**config**→**Page Configuration**. Then compile all files in the pcf folder. If there are no errors relating to postOnChange, your corrections are complete.

New mechanism for reloading Gosu classes (DOC-8218)

Issue – In past releases, you could modify your PCF files and Gosu classes in Studio, and then reload the changes into the running server by pressing Alt+Shift+L in the application user interface. This shortcut no longer loads Gosu classes.

Workaround – To have the server reload your Gosu classes, in Studio, click **Build**→**Make Project**. When Studio is finished compiling your project, the changes will be loaded. You can also restart your server to load the Gosu classes.