

# Guidewire PolicyCenter®

## PolicyCenter New and Changed Guide

RELEASE 8.0.2

Copyright © 2001-2014 Guidewire Software, Inc. All rights reserved.

Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Reinsurance Management, Guidewire ContactManager, Guidewire Vendor Data Management, Guidewire Client Data Management, Guidewire Rating Management, Guidewire InsuranceSuite, Guidewire ContactCenter, Guidewire Studio, Guidewire Product Designer, Guidewire Live, Guidewire DataHub, Guidewire InfoCenter, Guidewire Standard Reporting, Guidewire ExampleCenter, Guidewire Account Manager Portal, Guidewire Claim Portal, Guidewire Policyholder Portal, ClaimCenter, BillingCenter, PolicyCenter, InsuranceSuite, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks, service marks, or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries.

All other trademarks are the property of their respective owners.

**This material is confidential and proprietary to Guidewire and subject to the confidentiality terms in the applicable license agreement and/or separate nondisclosure agreement.**

Guidewire products are protected by one or more United States patents.

Product Name: Guidewire PolicyCenter

Product Release: 8.0.2

Document Name: PolicyCenter New and Changed Guide

Document Revision: 20-May-2014

# Contents

<b>About PolicyCenter Documentation .....</b>	<b>15</b>
Conventions in This Document .....	16
Support .....	16

## Part I

### What's New and Changed in 8.0 Maintenance Releases

<b>1 New and Changed in PolicyCenter 8.0.2.....</b>	<b>19</b>
New in PolicyCenter 8.0.2 .....	19
Multicurrency Integration between BillingCenter and PolicyCenter .....	19
New Plugin Interface InboundIntegrationMessageReply .....	19
Duplicating a Product Offering in Product Designer .....	20
Changed in PolicyCenter 8.0.2 .....	20
Removed Configuration Parameters .....	20
Changes to Rating Management .....	20
Complex Schedules.....	21
Changes to the Product Model .....	21
Integration Changes .....	21
<b>2 New and Changed in PolicyCenter 8.0.1.....</b>	<b>23</b>
New in PolicyCenter 8.0.1 .....	23
New in Rating Management .....	23
High Volume Requests for Quotes .....	24
Affinity Groups .....	24
New Support for OSGi Plugins.....	24
Gosu Using Clause Now Supports Finally Clause .....	25
Changed in PolicyCenter 8.0.1 .....	25
Changes to Rating Management .....	25
Changes to Web Service Transaction IDs.....	26
Change to Inbound Integration Configuration .....	26
Gosu Suppress Warnings Annotation .....	27
Gosu Compound Assignment Operators for Logical AND and Logical OR .....	27

## Part II

### What's New and Changed in 8.0.0

<b>3 New and Changed in PolicyCenter 8.0 .....</b>	<b>31</b>
New in PolicyCenter 8.0.....	31
Account Holder Info Screen .....	31
Policy Data Spreadsheet Import/Export .....	31
Multicurrency Policies .....	32
Service Tier Field on Account .....	32
Data Change Menu Link .....	33

Changed in PolicyCenter 8.0 .....	33
Policy Transactions .....	33
Changes to the Team Tab .....	33
Administration Menu Items Reorganized.....	34
Rate Routine Label Changed .....	34
Service Tier Field on Account .....	34
My Submissions Screen .....	34
My Other Policy Transactions Screen.....	35
Region and Region Search Screens Combined.....	35
Reinsurance Attachment Inclusions .....	35
<b>4 New and Changed in Configuration in 8.0 .....</b>	<b>37</b>
Important Changes to the Configuration Module .....	37
Changes to the Data Model.....	38
Changes to the Generate Dictionary Command .....	38
Changes to Guidewire Studio .....	38
Studio Now Built on IntelliJ IDEA Community Edition .....	38
Gosu Tester .....	38
Display Keys Editor .....	38
Studio Features That Are No Longer Available .....	38
Editing the Product Model .....	39
Changes to Configuration Parameters .....	39
Removed Configuration Parameters.....	39
Added Configuration Parameters .....	39
Changed Configuration Parameters .....	40
Changes to Web Services .....	40
Changes Related to PCF Files .....	40
Template Page PCF Files Removed .....	40
Miscellaneous PCF Changes.....	40
Changes to Search Criteria .....	43
Gosu Classes that Extend EntitySearchCriteria .....	43
Gosu Classes that Extend SearchCriteria .....	44
Changes to Configuring PolicyCenter .....	44
Changes to Team Tab Configuration .....	44
Changes to Configuring Underwriting Authority.....	44
Changes to the PolicyCenter Product Model .....	44
All Coverage Terms Must Have Value Types .....	45
Changes to Validation.....	45
New Documentation .....	45
Changes to Configuring Lines of Business.....	45
Lines of Business Modularization.....	45
Registering Lines of Business.....	46
Product Type Property .....	46
Policy Line Configuration Class .....	46
Logical Matcher Replaces checkForDuplicates Method .....	47
Blank Availability Rows are Not Valid .....	47
Handling Visibility of Coverages .....	47
Changes Related to Integrating with ContactManager.....	47
<b>5 New and Changed in Product Designer in 8.0 .....</b>	<b>49</b>
Editing the Product Model .....	49
Changes to Validation.....	49
New Documentation .....	50

<b>6 New and Changed in Gosu in 8.0 .....</b>	<b>51</b>
New in Gosu in PolicyCenter 8.0 .....	51
New Assert Statement .....	51
In For Loop Declarations, Local Variable is Now Optional .....	51
Final Local Variable and Class Variable Initialization Separate from Declaration .....	52
The new Operator is Now Optionally a Statement .....	53
Support Annotations on Function Parameters .....	53
Named Arguments in Annotations .....	53
New Support for Code Coverage Tools for Gosu Code .....	53
The Entity Touch API is Changed and No Longer Deprecated .....	53
Changed in Gosu in PolicyCenter 8.0 .....	54
Gosu is Now Case Sensitive In Most Cases .....	54
Bundle Changes from Gosu and Java .....	54
Gosu Map Enhancement Method Changes .....	54
Entity Literal Syntax is Deprecated .....	55
Entity Methods loadByKey and loadByPublicID Deprecated, New Method loadBean .....	55
For Various Types, Packages Changed to Reflect Public and Internal Status .....	55
Gosu Concurrency API Changes .....	56
The <> Inequality Operator Is Now Invalid .....	57
The Gosu Command Line Tool Built-in Editor Removed .....	57
Change in Gosu Named Parameters Usage .....	57
<b>7 New and Changed in Integration in 8.0 .....</b>	<b>59</b>
New in Integration in PolicyCenter 8.0 .....	59
New Phone Number Normalizer Plugin .....	59
For Free-Text Search, New Support for High Availability Using Apache Zookeeper .....	59
New MTOM Support for Results of Published WS-I Web Service .....	60
Multi-threaded Inbound Integration .....	60
New PolicyEarnedPremiumAPI Web Service .....	60
New Plugin Interface IFXRatePlugin .....	60
New Messaging Destination Option for Messages Without Primary Object .....	61
Regenerating WSDL and XSD in Web Service Collections (regen-from-wsc) .....	61
Changed in Integration in PolicyCenter 8.0 .....	61
RPCE Web Services Deprecated in 8.0 .....	61
Changes to Existing Web Services .....	62
Important Changes for Java Code .....	66
Plugin Registry Changes in Studio .....	72
Changes to Importing the PolicyCenter Product Model into ClaimCenter .....	73
Changes to Document Management Integration .....	73
Changes to Plugins and Classes for Line-of-business Modularization .....	74
ClaimCenter Typelist Generator Tool .....	75
Contact Messaging Flow Changed .....	75
PolicyCenter Plugin Changes For Multicurrency .....	75
Free-text Search Upgrades .....	76
New PCPolicyPublicID Property in PolicyCenter-BillingCenter Integration .....	76
<b>8 New and Changed in System Administration in 8.0 .....</b>	<b>77</b>
New in System Administration in 8.0 .....	77
Phone Number Normalizer Work Queue .....	77
Changed in System Administration in 7.0 .....	77
Changes to Database Configuration .....	77
Changes to Work Queue Configuration .....	79
Changes to Database Statistics Updating .....	80
User Batch Process Now Named Team Screens .....	80

<b>9</b>	<b>New and Changed in Rules in 8.0 .....</b>	<b>81</b>
	Changes in Rules in PolicyCenter 8.0 .....	81
	Preupdate Rules Removed .....	81
	Evaluation Rule Set Functionality Replaced .....	81
<b>10</b>	<b>New and Changed in Globalization in 8.0.....</b>	<b>83</b>
	New to Globalization in PolicyCenter 8.0 .....	83
	Separation of Language and Regional Format .....	84
	Base Configuration Language Support.....	84
	Base Configuration Region Support.....	84
	Base Configuration Currency Support .....	85
	Base Configuration Geographical Support.....	85
	Phone Number Data and Phone Validator Gosu Class .....	86
	Automatic Address Completion and Fill-in Plugin Functionality .....	87
	Ability to Provide a Sort Order for the Typecodes in a Typelist .....	87
	Language Module Installation Utility.....	88
	Globalization-related Configuration Parameters.....	88
	Globalization-related Typelists.....	88
	Ability to Set First Day of Week by Region.....	88
	New in the PolicyCenter 8.0 Data Model.....	89
	Changes to Globalization in PolicyCenter 8.0 .....	90
	Changes to Address Configuration 8.0.....	91
	Changes to Admin Data Localization.....	91
	Changes to 8.0 Typelists .....	92
	Changes to Files Used with Globalization .....	92

## Part III

### What's New and Changed in 7.0.0

<b>11</b>	<b>New and Changed in PolicyCenter 7.0 .....</b>	<b>97</b>
	PolicyCenter Lines of Business .....	97
	New Commercial Property Blankets.....	97
	Copying Coverages.....	98
	Changes to Personal Auto.....	98
	Business Auto Becomes Commercial Auto .....	98
	PolicyCenter Jobs .....	99
	Side-by-side Quoting .....	99
	Rescheduling a Cancellation.....	99
	Changing the Effective Date of a Policy Change .....	99
	Rewrite New Account Job .....	100
	PolicyCenter Components .....	100
	Changes to Activities .....	100
	Changes to Account History Screen and Events.....	101
	Account Relationships .....	101
	Moving or Rewriting Policies Between Accounts .....	102
	Merging Accounts.....	102
	Copying Data Between Policies .....	103
	Splitting and Spinning Policies .....	103
	Contact Tab.....	103
	Linking Addresses on Contacts.....	104
	Updating Contacts in Effective Time .....	104
	Reinsurance.....	104
	Change to End Date Field in Underwriting Entities .....	105

Administering PolicyCenter .....	105
Administering Policy Holds .....	105
Changes to Administering Policy Forms .....	105
Reassigning Job Roles .....	106
Product Integrations .....	106
Changes to the Billing System Integration .....	106
<b>12 New and Changed in Configuration in 7.0 .....</b>	<b>109</b>
Changes to the Data Model .....	109
Deprecation of MotorVehicleRecord .....	109
Modifications to Search Functionality .....	110
Modifications to Associative Arrays .....	110
Changes to Guidewire Studio .....	111
Modifications to Type System Refresh .....	111
Modifications to the Resources Pane .....	111
Modifications to Tools Options .....	111
Modifications to Dialogs .....	111
Modifications to Typelists .....	112
Changes to Configuration Parameters .....	113
Changes to Web Services .....	113
Changes Related to PCF Files .....	114
New iterator Attribute on Buttons .....	114
New disablePostOnEnter Attribute on Cell .....	114
New onEnter Attribute on WizardStep .....	115
New isDraftOnEnter Attribute .....	115
New Link widgets Menu Command .....	115
New Exposes Tab on ListViewPanel Widget .....	115
Deprecated Unintended ListView ContentCell Configuration .....	116
New Modal Cell and CellRef Widgets .....	116
New CheckedValuesMenuItem Iterator .....	116
Changes to the PolicyCenter Product Model .....	116
Changes to Question Sets .....	116
New Reloading Availability Data on a Running Server .....	117
Quote Rounding Level Within a Product .....	117
Changes to Configuring Lines of Business .....	117
Changes to Studio Icons for a Line of Business .....	117
Configuring Copy Data for a Line of Business .....	118
Improvements in Localization .....	118
Address Localization .....	118
Typelist Localization .....	118
New Jurisdiction Typelist .....	119
Display Key Localization .....	119
Command Line Export Translation File Import Tool .....	119
<b>13 New and Changed in Gosu in 7.0 .....</b>	<b>121</b>
New in Gosu in 7.0 .....	121
Gosu Support for Intervals .....	121
Gosu Support for Numeric Literals .....	121
New Null-safe Operators .....	122
New Named Function Arguments and Argument Defaults .....	122

Changed in Gosu in 7.0 . . . . .	122
Gosu Compiles to Java Virtual Machine Bytecode . . . . .	123
Gosu Syntax Changes (Compile Time Issues) . . . . .	123
Gosu Behavior Changes (Run Time Issues) . . . . .	128
New XML APIs and Improved XSD Support Using ‘XmlElement’ . . . . .	132
Guidewire XML Modeler (GX Modeler) Upgraded to Use New XML API . . . . .	133
Interceptors Removed . . . . .	133
<b>14 New and Changed in Integration in 7.0 . . . . .</b>	<b>135</b>
New in Integration in 7.0 . . . . .	135
Motor Vehicle Record (MVR) Plugin . . . . .	135
Reinsurance Architecture and Plugin . . . . .	135
New ClaimCenter PolicyCenter Integration . . . . .	135
Policy Hold Evaluation Plugin . . . . .	136
New Populator Plugin . . . . .	136
New Methods in ProductModelAPI . . . . .	136
Changes in Integration in 7.0 . . . . .	136
WS-I Web Services . . . . .	137
Suite Configuration File Overrides URLs to Guidewire Applications . . . . .	141
Setting Locale in WS-I Requests for Guidewire Servers Only . . . . .	142
Changes to Locally-Accessed RPCE Web Services . . . . .	142
IAccountPlugin Web Service Changes for Transferring Policies . . . . .	143
SubmissionAPI Changes . . . . .	143
PolicyAPI Changes . . . . .	144
PolicyRenewalAPI Changes . . . . .	144
Confirming Renewals Integration Flow . . . . .	145
Guidewire InsuranceSuite Plugin Implementations are Versioned . . . . .	145
SOAP Implementation Classes and WSDL Packages Include Version . . . . .	146
Geocoding Only Some Addresses . . . . .	147
Changes to PolicyCenter Contact-related Plugins . . . . .	147
ContactAPI Web Service Changes . . . . .	148
Improvements to Matching Classes . . . . .	148
Escalation and Exception Plugins . . . . .	150
Billing Plugin Changes . . . . .	151
Billing Summary Plugin Changes . . . . .	151
For Messaging, a Contact is Now a Safe-ordered Object . . . . .	151
Integrating PolicyCenter 7.0.0 with ClaimCenter 6.0.0 . . . . .	152
AccountSyncable Changes . . . . .	152
Changes to Messaging System Safe Ordering . . . . .	153
<b>15 New and Changed in System Administration in 7.0 . . . . .</b>	<b>155</b>
New in System Administration in 7.0 . . . . .	155
DCE VM Strongly Recommended for Development Environments . . . . .	155
Support for Importing and Exporting Policy Forms and Policy Holds . . . . .	157
Changed in System Administration in 7.0 . . . . .	157
Updated System Requirements . . . . .	157
Database Configuration Changes . . . . .	157
Maximum Contact Search Results Parameter Name . . . . .	157
<b>16 New and Changed in Rules in 7.0 . . . . .</b>	<b>159</b>
Changes in Rules in PolicyCenter 7.0 . . . . .	159

## Part IV

### Release Notes Archive

<b>17 Guidewire PolicyCenter 4.0.0 Release Notes .....</b>	<b>163</b>
Overview .....	163
Release Information .....	163
Version Number .....	164
Installation Requirements .....	164
Support .....	164
Installing This Release .....	164
InetSoft Evaluation .....	164
Upgrade Issues .....	164
Changes in This Release .....	164
Base PCF File Changes .....	165
Configuration Changes .....	165
Known Issues and Limitations .....	165
<b>18 Guidewire PolicyCenter 4.0.1 Release Notes .....</b>	<b>171</b>
Overview .....	171
Release Information .....	171
Version Number .....	172
Installation Requirements .....	172
Support .....	172
Installing This Release .....	172
Software Updates .....	172
InetSoft Evaluation .....	172
Upgrade Issues .....	172
Changes in This Release .....	173
Base PCF File Changes .....	173
Configuration Changes .....	173
Improvements and General Issues .....	173
Known Issues and Limitations .....	186
<b>19 Guidewire PolicyCenter 4.0.2 Release Notes .....</b>	<b>191</b>
Overview .....	191
Release Information .....	191
Version Number .....	192
Installation Requirements .....	192
Support .....	192
Installing This Release .....	192
InetSoft Evaluation .....	192
Changes in This Release .....	192
Base PCF File Changes .....	192
Base Resources Changes .....	193
Improvements and General Issues .....	193
Known Issues and Limitations .....	203
<b>20 Guidewire PolicyCenter 4.0.3 Release Notes .....</b>	<b>209</b>
Overview .....	209
Release Information .....	209
Version Number .....	210
Installation Requirements .....	210
Support .....	210

Installing This Release .....	210
InetSoft Evaluation .....	210
Issues and Major Changes.....	210
Changes in This Release.....	211
Base PCF File Changes.....	211
Base Resources Changes.....	211
Improvements and General Issues .....	211
Known Issues and Limitations .....	218
<b>21 Guidewire PolicyCenter 4.0.4 Release Notes.....</b>	<b>225</b>
Overview.....	225
Release Information .....	225
Version Number .....	226
Support .....	226
Installing This Release .....	226
InetSoft Evaluation .....	226
Changes in This Release.....	226
Base PCF File Changes.....	226
Improvements to Upgrade Diff Report.....	227
Improvements and General Issues .....	227
Known Issues and Limitations .....	231
<b>22 Guidewire PolicyCenter 4.0.5 Release Notes.....</b>	<b>239</b>
Overview.....	239
Release Information .....	239
Version Number .....	240
Support .....	240
Installing This Release .....	240
InetSoft Evaluation .....	240
Changes in This Release.....	240
Base PCF File Changes.....	240
Custom Upgrade Diff Report .....	241
Change in Web Service Behavior for Committing Data to the Database (PL-14984).....	241
Improvements and General Issues .....	242
Known Issues and Limitations .....	245
PolicyCenter Known Issues .....	245
Platform Known Issues .....	246
<b>23 Guidewire PolicyCenter 4.0.6 Release Notes.....</b>	<b>251</b>
Overview.....	251
Release Information .....	252
Version Number .....	252
Installing This Release .....	252
InetSoft Evaluation .....	252
Support .....	252
Changes in This Release.....	252
Base PCF File Changes.....	253
Custom Upgrade Diff Report .....	253
Improvements and General Issues .....	253
Known Issues and Limitations .....	256
PolicyCenter Known Issues .....	256
Platform Known Issues .....	258
<b>24 Guidewire PolicyCenter 7.0.0 Release Notes.....</b>	<b>263</b>
Overview.....	263

Release Information .....	264
Version Number .....	264
Changes to Application Server Requirements .....	264
Support .....	264
Installing This Release .....	264
Changes in This Release .....	264
Improvements to Upgrade Diff Report .....	264
Known Issues and Limitations .....	265
PolicyCenter Known Issues .....	265
Platform Known Issues .....	270
<b>25 Guidewire PolicyCenter 7.0.1 Release Notes .....</b>	<b>275</b>
Overview .....	275
Release Information .....	276
Version Number .....	276
Changes to Application Server Requirements .....	276
Installing This Release .....	276
Upgrade Issues .....	276
Support .....	277
Changes in This Release .....	277
Rate Table Maintenance .....	277
Base PCF File Changes .....	277
Base Resources Changes .....	277
Upgrade Diff Report .....	278
Improvements and General Issues .....	278
PolicyCenter Improvements and General Issues .....	278
Platform Improvements and General Issues .....	291
Known Issues and Limitations .....	295
PolicyCenter Known Issues .....	295
Studio/Platform Known Issues .....	298
<b>26 Guidewire PolicyCenter 7.0.2 Release Notes .....</b>	<b>305</b>
Overview .....	305
Release Information .....	306
Version Number .....	306
Installing This Release .....	306
Support .....	306
Issues and Major Changes .....	306
Upgrade Tool does not handle rules upgrade properly (PL-18021) .....	307
Archiving Not Supported .....	307
ContactManager Integration .....	307
Studio Product Model Improvements .....	307
Geocoding Using Bing and MapPoint (PL-16708) .....	309
Rate Table Maintenance Improvements .....	309
Base PCF File Changes .....	310
Base Resources Changes .....	310
Changes in This Release Provided in Upgrade Diff Report .....	310
Improvements and General Issues .....	311
Known Issues and Limitations .....	316
PolicyCenter Known Issues .....	317
Studio/Platform Issues .....	320
<b>27 Guidewire PolicyCenter 7.0.3 Release Notes .....</b>	<b>323</b>
Overview .....	323

Release Information .....	324
Version Number .....	324
Installing This Release .....	324
Support .....	324
Issues and Major Changes .....	324
Integrating with BillingCenter .....	325
Archiving in PolicyCenter .....	325
Changes to Search Requirements .....	325
Policy Location Search API .....	326
Studio Product Model Improvements .....	326
Guidewire Standard Reporting .....	326
Base PCF File Changes .....	328
Base Resources Changes .....	328
Changes in This Release Provided in Upgrade Diff Report .....	328
Improvements and General Issues .....	329
Known Issues and Limitations .....	337
PolicyCenter Known Issues .....	337
Studio/Platform Known Issues .....	340
<b>28 Guidewire PolicyCenter 7.0.4 Release Notes .....</b>	<b>345</b>
Overview .....	346
Release Information .....	346
Version Number .....	346
Installing This Release .....	346
Support .....	346
Issues and Major Changes .....	346
Changes and Additions to Rating Management .....	347
Base PCF File Changes .....	347
Base Resources Changes .....	347
Changes in This Release Provided in Upgrade Diff Report .....	347
Improvements and General Issues .....	348
Known Issues and Limitations .....	357
PolicyCenter Known Issues .....	357
Studio/Platform Known Issues .....	362
<b>29 Guidewire PolicyCenter 7.0.5 Release Notes .....</b>	<b>365</b>
Overview .....	365
Release Information .....	366
Version Number .....	366
Installing This Release .....	366
Support .....	366
Issues and Major Changes .....	366
Changes and Additions to Rating Management .....	367
Generic Schedules .....	367
Localization Improvements .....	367
Logging API Changes .....	367
Base PCF File Changes .....	367
Base Resources Changes .....	367
Changes in This Release Provided in Upgrade Diff Report .....	367
Improvements and General Issues .....	368
Known Issues and Limitations .....	372
PolicyCenter Known Issues .....	372
Studio/Platform Known Issues .....	377

<b>30 Guidewire PolicyCenter 7.0.6 Release Notes .....</b>	<b>381</b>
Overview .....	381
Release Information .....	382
Version Number .....	382
Installing This Release .....	382
Support .....	382
Issues and Major Changes .....	382
Quote Purging .....	382
Free-text Search .....	385
Rating Management .....	385
Base PCF File Changes .....	385
Base Resources Changes .....	385
Changes in This Release Provided in Upgrade Diff Report .....	386
Improvements and General Issues .....	386
PolicyCenter Improvements and General Issues .....	386
Platform Improvements and General Issues .....	390
Known Issues and Limitations .....	393
PolicyCenter Known Issues .....	394
Studio/Platform Known Issues .....	398
<b>31 Guidewire PolicyCenter 7.0.7 Release Notes .....</b>	<b>401</b>
Overview .....	401
Release Information .....	402
Version Number .....	402
Installing This Release .....	402
Support .....	402
Issues and Major Changes .....	402
Base PCF File Changes .....	402
Base Resources Changes .....	403
Changes in This Release Provided in Upgrade Diff Report .....	403
Improvements and General Issues .....	403
PolicyCenter Improvements and General Issues .....	403
Platform Improvements and General Issues .....	407
Known Issues and Limitations .....	409
PolicyCenter Known Issues .....	409
Studio/Platform Known Issues .....	414
<b>32 Guidewire PolicyCenter 8.0.0 Release Notes .....</b>	<b>417</b>
Overview .....	417
Release Information .....	418
Version Number .....	418
Installing This Release .....	418
Support .....	418
Issues and Major Changes .....	418
Upgrade does not convert Money properties to MonetaryAmount properties in entity extensions .....	418
Base PCF File Changes .....	419
Base Resource Changes .....	419
Changes in This Release Provided in Upgrade Diff Report .....	419
Known Issues and Limitations .....	420
PolicyCenter Known Issues .....	420
Product Designer Known Issues .....	425
Studio/Platform Known Issues .....	426
<b>33 Guidewire PolicyCenter 8.0.1 Release Notes .....</b>	<b>431</b>
Overview .....	431

---

Release Information .....	432
Version Number .....	432
Installing This Release .....	432
Support .....	432
Issues and Major Changes .....	432
Product Designer Localization .....	433
Product Designer Requires Committing All Change Lists Before Upgrading PolicyCenter ..	433
Base PCF File Changes.....	433
Base Resource Changes .....	433
Changes in This Release Provided in Upgrade Diff Report.....	433
Improvements and General Issues .....	433
PolicyCenter Improvements and General Issues.....	434
Product Designer Improvements and General Issues .....	439
Platform Improvements and General Issues .....	441
Known Issues and Limitations .....	447
PolicyCenter Known Issues .....	447
Product Designer Known Issues .....	452
Studio/Platform Known Issues .....	453

# About PolicyCenter Documentation

The following table lists the documents in PolicyCenter documentation.

Document	Purpose
<i>InsuranceSuite Guide</i>	If you are new to Guidewire InsuranceSuite applications, read the <i>InsuranceSuite Guide</i> for information on the architecture of Guidewire InsuranceSuite and application integrations. The intended readers are everyone who works with Guidewire applications.
<i>Application Guide</i>	If you are new to PolicyCenter or want to understand a feature, read the <i>Application Guide</i> . This guide describes features from a business perspective and provides links to other books as needed. The intended readers are everyone who works with PolicyCenter.
<i>Upgrade Guide</i>	Describes how to upgrade PolicyCenter 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.
<i>New and Changed Guide</i>	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.
<i>Installation Guide</i>	Describes how to install PolicyCenter. The intended readers are everyone who installs the application for development or for production.
<i>System Administration Guide</i>	Describes how to manage a PolicyCenter system. The intended readers are system administrators responsible for managing security, backups, logging, importing user data, or application monitoring.
<i>Configuration Guide</i>	The primary reference for configuring initial implementation, data model extensions, and user interface (PCF) files. The intended readers are all IT staff and configuration engineers.
<i>Globalization Guide</i>	Describes how to configure PolicyCenter for a global environment. Covers globalization topics such as global locales, languages, date and number formats, names, currencies, addresses, and phone numbers. The intended readers are configuration engineers who work with locales and languages.
<i>Rules Guide</i>	Describes business rule methodology and the rule sets in PolicyCenter Studio. The intended readers are business analysts who define business processes, as well as programmers who write business rules in Gosu.
<i>Contact Management Guide</i>	Describes how to configure Guidewire InsuranceSuite applications to integrate with ContactManager and how to manage client and vendor contacts in a single system of record. The intended readers are PolicyCenter implementation engineers and ContactManager administrators.
<i>Best Practices Guide</i>	A reference of recommended design patterns for data model extensions, user interface, business rules, and Gosu programming. The intended readers are configuration engineers.
<i>Integration Guide</i>	Describes the integration architecture, concepts, and procedures for integrating PolicyCenter with external systems and extending application behavior with custom programming code. The intended readers are system architects and the integration programmers who write web services code or plugin code in Gosu or Java.
<i>Gosu Reference Guide</i>	Describes the Gosu programming language. The intended readers are anyone who uses the Gosu language, including for rules and PCF configuration.
<i>Glossary</i>	Defines industry terminology and technical terms in Guidewire documentation. The intended readers are everyone who works with Guidewire applications.

Document	Purpose
<i>Product Model Guide</i>	Describes the PolicyCenter product model. The intended readers are business analysts and implementation engineers who use PolicyCenter or Product Designer. To customize the product model, see the <i>Product Designer Guide</i> .
<i>Product Designer Guide</i>	Describes how to use Product Designer to configure lines of business. The intended readers are business analysts and implementation engineers who customize the product model and design new lines of business.

## Conventions in This Document

Text style	Meaning	Examples
<i>italic</i>	Emphasis, special terminology, or a book title.	A <i>destination</i> sends messages to an external system.
<b>bold</b>	Strong emphasis within standard text or table text.	You <b>must</b> define this property.
<b>narrow bold</b>	The name of a user interface element, such as a button name, a menu item name, or a tab name.	Next, click <b>Submit</b> .
<code>monospaced</code>	Literal text that you can type into code, computer output, class names, URLs, code examples, parameter names, string literals, and other objects that might appear in programming code. In code blocks, bold formatting highlights relevant sections to notice or to configure.	Get the field from the <code>Address</code> object.
<code>monospaced italic</code>	Parameter names or other variable placeholder text within URLs or other code snippets.	Use <code>getName(first, last)</code> . <code>http://SERVNAME/a.html</code> .

## Support

For assistance with this software release, contact Guidewire Customer Support:

- At the Guidewire Resource Portal – <http://guidewire.custhelp.com>
- By email – [support@guidewire.com](mailto:support@guidewire.com)
- By phone – +1-650-356-4955

# What's New and Changed in 8.0 Maintenance Releases

This part includes information about new and changed features in PolicyCenter 8.0 maintenance releases. For information about new and changed features in 8.0.0, see “What's New and Changed in 8.0.0” on page 29.



# 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.

This topic includes:

- “New in PolicyCenter 8.0.2” on page 19
- “Changed in PolicyCenter 8.0.2” on page 20

## New in PolicyCenter 8.0.2

This topic includes:

- “Multicurrency Integration between BillingCenter and PolicyCenter” on page 19
- “New Plugin Interface InboundIntegrationMessageReply” on page 19
- “Duplicating a Product Offering in Product Designer” on page 20

### Multicurrency Integration between BillingCenter and PolicyCenter

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

**See also**

- “Multicurrency Integration Between BillingCenter and PolicyCenter” on page 719 in the *Application Guide*

### New Plugin Interface InboundIntegrationMessageReply

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. For related upgrade information, see “InboundIntegrationPlugin Plugin Interface Changed Name” on page 21,

## Duplicating a Product Offering in Product Designer

You can now 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.

## Changed in PolicyCenter 8.0.2

This topic includes:

- “Removed Configuration Parameters” on page 20
- “Changes to Rating Management” on page 20
- “Complex Schedules” on page 21
- “Changes to the Product Model” on page 21
- “Integration Changes” on page 21

### Removed Configuration Parameters

The following configuration parameters have been removed from PolicyCenter 8.0.2:

- `BucketSizeForHistogramsOnAllIndexedColumns`
- `CollectHistogramsOnAllIndexedColumns`

See “Configuring Database Statistics Generation” on page 41 in the *System Administration Guide*.

### Changes to Rating Management

The following changes were made to Guidewire Rating Management.

#### Interpolation With and Without Relaxing

You can 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

- “Rate Table with Interpolated Rate Factor” on page 540 in the *Application Guide*
- “Matching Rules” on page 561 in the *Application Guide*

#### Match Operators in Rate Table Definitions

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

- “Matching Rules” on page 561 in the *Application Guide*

## Complex Schedules

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

The following changes were made to the product model.

### Rate is Relative for Rate Modifiers

When specifying a rate modifier in Product Designer, there is a new **Rate is Relative** field where you can specify a rate modifier relative to 0 or 1.

#### See also

- “Rate Modifiers” on page 52 in the *Product Model Guide*

### Added Peril to the list of coverage term model types

You can now 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

- “Coverage Term Model Types” on page 35 in the *Product Model Guide*

## Integration Changes

### InboundIntegrationPlugin Plugin Interface Changed Name

Two related changes happened in this release 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](#)**

The `IBillingSystemPlugin` plugin interface has two new methods for syncing producers and producer codes: `syncProducer` and `syncProducerCode`.

# 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.

This topic includes:

- “New in PolicyCenter 8.0.1” on page 23
- “Changed in PolicyCenter 8.0.1” on page 25

## New in PolicyCenter 8.0.1

This topic includes:

- “New in Rating Management” on page 23
- “High Volume Requests for Quotes” on page 24
- “Affinity Groups” on page 24
- “New Support for OSGi Plugins” on page 24
- “Gosu Using Clause Now Supports Finally Clause” on page 25

### New in Rating Management

This topic describes what’s new in Guidewire Rating Management.

#### Extracting and Purging Rating Worksheets

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**

- “Extracting and Purging Rating Worksheets” on page 551 in the *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. 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**

- “Rate Table Normalization” on page 537 in the *Application Guide*

## Editing Long Rate Routines

When 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**

- “Editing Long Rate Routines” on page 580 in the *Application Guide*

## High Volume Requests for Quotes

Some insurance carriers need to handle high volumes of quote requests generated by external applications, such as the websites of comparative raters. To handle these requests, these carriers must generate large volumes of quotes. An external application waits for a quote, therefore PolicyCenter must generate the quote quickly.

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 carriers, PolicyCenter provides an API and plugin interface for generating quotes.

**See also**

- “Handling High Volume Quote Requests” on page 563 in the *Configuration Guide*

## Affinity Groups

Some carriers write policies based on the insured belonging to a certain group. For these carriers, the lines of business in the base configuration enable you to associate an affinity group with a policy. Affinity groups are defined in the **Administration** tab by users who have the required **Affinity Group Administration** permission. 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**

- “Affinity Groups” on page 172 in the *Application Guide*

## New Support for OSGi Plugins

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**

- “Overview of PolicyCenter Plugins” on page 182 in the *Integration Guide*
- “Overview of Java and OSGi Support” on page 657 in the *Integration Guide*
- “OSGi Plugin Deployment with IntelliJ IDEA with OSGi Editor” on page 675 in the *Integration Guide*

## Gosu Using Clause Now Supports Finally Clause

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**

- “Optional Use of a finally Clause with a using Clause” on page 109 in the *Gosu Reference Guide*

## Changed in PolicyCenter 8.0.1

This topic includes:

- “Changes to Rating Management” on page 25
- “Changes to Web Service Transaction IDs” on page 26
- “Change to Inbound Integration Configuration” on page 26
- “Gosu Suppress Warnings Annotation” on page 27
- “Gosu Compound Assignment Operators for Logical AND and Logical OR” on page 27

### Changes to Rating Management

This topic describes changes to Guidewire Rating Management.

#### Configuring Rating Worksheets

In this release, there are fewer steps for configuring rating worksheets for other lines of business.

**See also**

- “Configuring Rating Worksheets” on page 540 in the *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`.

**See also**

- “Memory Threshold Parameter for Rate Tables” on page 520 in the *Configuration Guide*

#### 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**

- “Extracting and Purging Rating Worksheets” on page 551 in the *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**

- “Configuring a New Match Operation” on page 526 in the *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**

- “Include Property” on page 548 in the *Configuration Guide*
- “Filter Irrelevant Items” on page 549 in the *Configuration Guide*

## Changes to Web Service Transaction IDs

This release changed 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 `@WsCheckDuplicateExternalTransaction`. 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**

- “Checking for Duplicate External Transaction IDs” on page 65 in the *Integration Guide*
- “Setting Guidewire Transaction IDs” on page 89 in the *Integration Guide*

## Change to Inbound Integration Configuration

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`. Register an implementation of this plugin to add your custom logic to your file and JMS integration.

**See also**

- “Multi-threaded Inbound Integration Overview” on page 321 in the *Integration Guide*

## Gosu Suppress Warnings Annotation

The Gosu language now provides limited support for the Java annotation `@SuppressWarnings`, which tells the compiler to suppress warnings.

### See also

- “Built-in Annotations” on page 220 in the *Gosu Reference Guide*

## Gosu Compound Assignment Operators for Logical AND and Logical OR

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

- “Variable Assignment” on page 89 in the *Gosu Reference Guide*



# What's New and Changed in 8.0.0

This part includes information about new and changed features in PolicyCenter 8.0.0. For information about new and changed features in 8.0 maintenance releases, see “What’s New and Changed in 8.0 Maintenance Releases” on page 17.



# New and Changed in PolicyCenter 8.0

This document describes the new and changed features in PolicyCenter 8.0.

This topic includes:

- “New in PolicyCenter 8.0” on page 31
- “Changed in PolicyCenter 8.0” on page 33

## New in PolicyCenter 8.0

This topic describes what’s new in this version of PolicyCenter.

### 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

- “Account Holder Information” on page 369 in the *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

- “Policy Data Spreadsheet Import/Export” on page 461 in the *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, a carrier 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

- “Multicurrency Features” on page 515 in the *Application Guide*
- “Configuring Multicurrency” on page 567 in the *Configuration Guide*
- “Configuring Currencies” on page 87 in the *Globalization Guide*

## Service Tier Field on Account

Service tiers enable the carrier 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 **Data Change** 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

- “Data Change API” on page 47 in the *System Administration Guide*

## Changed in PolicyCenter 8.0

This topic describes changes in PolicyCenter 8.0.

### 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 44
- “Team Management” on page 663 in the *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
<b>Primary Insured</b>	This field is unchanged.
<b>Effective Date</b>	The <b>PolicyPeriod.EditEffectiveDate</b> from the policy period used to determine the <b>Status</b> . By default, this field is used to sort the list.
<b>Quote Needed</b>	For submission policy transactions, this field displays <b>Submission.DateQuoteNeeded</b> . For issuance policy transactions, this field is blank.
<b>Transaction #</b>	This field was previously labeled <b>Job #</b> .
<b>Type</b>	This field is unchanged.
<b>Status</b>	This field is unchanged.
<b>Issued</b>	This field displays <b>Policy.Issued</b> .
<b>Product</b>	This field was previously labeled <b>Policy Type</b> .
<b>Producer</b>	This field is visible if user does not have the View Producer Desktop Details permission. This is a new permission.
<b>Underwriter</b>	This field displays the user with the Underwriter role in the <b>Job.RoleAssignments</b> 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 <code>PolicyPeriod.EditEffectiveDate</code> from the query used to determine the Status. By default, this field is used to sort the list.
Transaction #	This field was previously labeled <b>Work Order #</b> .
Status	This field is unchanged.
Product	This field was previously labeled <b>Policy Type</b> .
Producer	This field is visible if user does not have the <b>View Producer Desktop Details</b> permission. This is a new permission.
Underwriter	This field displays the user with the <b>Underwriter</b> role in the <code>Job.RoleAssignments</code> 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 `ListPageSizeDefault`.

The `SearchOn` property is defined in `RegionsUIHelper.gs` in the `gw.admin` package. In the base configuration, this property is set to **AUTO**.

The `ListPageSizeDefault` 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.



# New and Changed in Configuration in 8.0

This topic includes:

- “Important Changes to the Configuration Module” on page 37
- “Changes to the Data Model” on page 38
- “Changes to the Generate Dictionary Command” on page 38
- “Changes to Guidewire Studio” on page 38
- “Changes to Configuration Parameters” on page 39
- “Changes to Web Services” on page 40
- “Changes Related to PCF Files” on page 40
- “Changes to Search Criteria” on page 43
- “Changes to Configuring PolicyCenter” on page 44
- “Changes to the PolicyCenter Product Model” on page 44
- “Changes to Configuring Lines of Business” on page 45
- “Changes Related to Integrating with ContactManager” on page 47

## Important Changes to the Configuration Module

In version 8.0, the structure and meaning of modules has changed.

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` 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

- “Regenerating the Data Dictionary and Security Dictionary” on page 32 in the *Configuration Guide*

## Changes to Guidewire Studio

### 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 “Using the Display Keys Editor” on page 137 in 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 *PolicyCenter Product Designer Guide*. For field-level information while editing the product model, click the **Help** button while viewing any page in Product Designer.

## Changes to Configuration Parameters

### 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

The following configuration parameters have been changed in PolicyCenter 8.0.0:

- `AllowActiveX` – Renamed to `AllowDocumentAssistant`, and default changed to `false`.
- `ClusterProtocolStack` – Default value changed. See “`ClusterProtocolStack`” on page 46 in the *Configuration Guide*.
- `PolicyChangesStatisticsWindowSize` – renamed to `OtherWorkOrdersStatisticsWindowSize`.
- `UISkin` – Default value changed to `Titanium`.
- `UseGuidewireActiveXControlToDisplayDocuments` – Renamed to `UseDocumentAssistantToDisplayDocuments`.

## Changes to Web Services

For changes related to web services, see “Changes to Existing Web Services” on page 62.

## 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 “`Servlets`” on page 629 in the *Integration Guide*. Your servlet implementation class can use Gosu templates and render a Gosu template. See “`Why Gosu Templates are Useful for Data Extraction`” on page 637 in the *Integration Guide*. Also see “`Template Overview`” on page 347 in 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 “`Web Services Introduction`” on page 39 in 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.
- 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 dependant 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 more distinction between a stacked vs non-stacked area)
  - XYStep → XYLine
  - XYStepArea → XYArea

## Changes to Search Criteria

In PolicyCenter 7.0, search criteria was represented using entities, Java, and Gosu classes. In PolicyCenter 8.0, most of the search criteria entities were converted to Gosu classes, which you can extend. These classes typically extend the `EntitySearchCriteria` or `SearchCriteria` class.

The `PolicySearchCriteria` entity has not been converted.

### See also

- “Working with Database Search Criteria in Gosu” on page 339 in the *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
<code>AbstractClassCodeSearchCriteria</code>	<code>gw.lob</code>	
<code>BOPClassCodeSearchCriteria</code>	<code>gw.lob.bop</code>	
<code>CPClassCodeSearchCriteria</code>	<code>gw.lob.cp</code>	
<code>GLClassCodeSearchCriteria</code>	<code>gw.lob.g1</code>	
<code>WCClassCodeSearchCriteria</code>	<code>gw.lob.wc</code>	
<code>AccountSearchCriteria</code>	<code>gw.account</code>	
<code>AccountSearchCriteria700</code>	<code>gw.webservice.pc.pc700.a</code>	This is for backwards compatibility. <code>ccount</code>
<code>ActivitySearchCriteria</code>	<code>gw.activity</code>	
<code>AgreementSearchCriteria</code>	<code>gw.reinsurance.search</code>	
<code>FacultativeSearchCriteria</code>	<code>gw.reinsurance.search</code>	
<code>FormPatternSearchCriteria</code>	<code>gw.forms</code>	
<code>HistorySearchCriteria</code>	<code>gw.history</code>	
<code>IndustryCodeSearchCriteria</code>	<code>gw.product</code>	
<code>MessageSearchCriteria</code>	<code>gw.system.messaging</code>	
<code>PolicyLocationBoundingBoxSearchCriteria</code>	<code>gw.policylocation</code>	
<code>PolicyLocationSearchCriteria</code>	<code>gw.policylocation</code>	
<code>ProducerCodeSearchCriteria</code>	<code>gw.product</code>	
<code>ProgramSearchCriteria</code>	<code>gw.reinsurance.search</code>	
<code>RateBookSearchCriteria</code>	<code>gw.rating.rtm.query</code>	
<code>RateRoutineSearchCriteria</code>	<code>gw.rating.rtm.query</code>	
<code>RateTableDefinitionSearchCriteria</code>	<code>gw.rating.rtm.query</code>	
<code>RILocationRiskProximitySearchCriteria</code>	<code>gw.reinsurance.search</code>	
<code>RiskClassSearchCriteria</code>	<code>gw.lob.common</code>	
<code>SharedContactAccountSearchCriteria</code>	<code>gw.account</code>	
<code>TaxLocationSearchCriteria</code>	<code>gw.lob.common</code>	
<code>TerritoryLookupCriteria</code>	<code>gw.lob.common</code>	

## Gosu Classes that Extend SearchCriteria

The `SearchCriteria` abstract class provides a standard way of searching for objects other than entities. The Gosu classes that extend the `SearchCriteria` class (in the `gw.search` package) are:

Gosu class	Package	Notes
<code>BillingAccountSearchCriteria</code>	<code>gw.plugin.billing</code>	
<code>ClausePatternSearchCriteria</code>	<code>gw.productmodel</code>	This class replaces <code>CoveragePatternSearchCriteria</code> .
<code>PCZoneSearchCriteria</code>	<code>gw.api.admin</code>	
<code>UWIssueTypeSearchCriteria</code>	<code>gw.job.uw</code>	

## 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 33
- “Configuring the Team Tab” on page 511 in the *Configuration Guide*
- “`OtherWorkOrdersStatisticsWindowSize`” on page 48 in the *Configuration Guide*
- “Configuring the Team Screens Batch Process for Team Statistics” on page 511 in the *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

- “Configuring Underwriting Authority” on page 469 in the *Configuration Guide*
- “Overview of Configuring Underwriting Authority” on page 469 in the *Configuration Guide*
- “Checking Sets and Evaluators” on page 481 in the *Configuration Guide*

## Changes to the PolicyCenter Product Model

This topic describes changes to the PolicyCenter Product Model.

## 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.

## Changes to Validation

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 validate 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

- “[Preventing Illegal Product Model Changes](#)” on page 111 in the *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:

- *PolicyCenter 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 the **Help** button 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.

#### See also

- “Guidelines for Modularizing Line-of-business Code” on page 443 in the *Configuration Guide*
- “Changes to Plugins and Classes for Line-of-business Modularization” on page 74

## 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.

## 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 type-code.

#### See also

- “Step 2: Register the New Line of Business” on page 128 in the *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. The new properties are defined in the `gw.productmodel.ProductEnhancement` enhancement.

## 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**

- “Defining Availability in Lookup Tables” on page 85 in the *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**

- “Rendering Common Coverages” on page 29 in the *Product Model Guide*

## Changes Related to Integrating with ContactManager

For changes in PolicyCenter related to integrating with ContactManager and changes in ContactManager itself, see “New And Changed Features in ContactManager 8.0.0” on page 15 in the *Contact Management Guide*.

In particular, see the following table:

- “PolicyCenter File Name and Package Changes for ContactManager Integration” on page 20 in the *Contact Management Guide*



# New and Changed in Product Designer in 8.0

This topic includes:

- “Editing the Product Model” on page 49

## 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 *PolicyCenter Product Designer Guide*. For field-level information while editing the product model, click the **Help** button while viewing any page in Product Designer.

## Changes to Validation

Because product model editing in Product Designer is decoupled from data model editing in Studio, validation and verification of the product model has changed. 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 validate 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. Product Designer does not allow the commit to 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**

- “Preventing Illegal Product Model Changes” on page 111 in the *Product Model Guide*

## New Documentation

To better describe product model configuration to the intended audiences, the PolicyCenter documentation set includes the following new documentation:

- *PolicyCenter 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 the Help button in Product Designer to display help for the current page.

# New and Changed in Gosu in 8.0

This topic includes:

- “New in Gosu in PolicyCenter 8.0” on page 51
- “Changed in Gosu in PolicyCenter 8.0” on page 54

## New in Gosu in PolicyCenter 8.0

### 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 “Assert Statements” on page 110 in the *Gosu Reference Guide*.

### In For Loop Declarations, Local Variable is Now Optional

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 Local Variable and Class 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 “New Is Optionally a Statement” on page 88 in the *Gosu Reference Guide*.

## Support 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 “Annotating a Class, Method, Type, Class Variable, or Argument” on page 219 in the *Gosu Reference Guide*.

There are currently no compile time APIs for changing IDE behavior based on the annotations.

You can access function parameter annotations using Java reflection APIs. See “Gosu Class Function Parameter Argument Annotations at Run Time” on page 223 in the *Gosu Reference Guide*.

## Named Arguments in Annotations

Gosu annotations now support the named arguments calling convention:

```
@KnownBreak(:targetUser = "user", :targetBranch = "branch", :jira = "ABC-xxxxx")
```

For related information about named arguments, see “Named Arguments and Argument Defaults” on page 98 in 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 “Code Coverage Support” on page 37 in 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 “Entity Instance Versioning and the Entity Touch API” on page 344 in 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 “Entity Instance Versioning and the Entity Touch API” on page 344 in the *Gosu Reference Guide*.

## Changed in Gosu in PolicyCenter 8.0

### 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 “Bundles and Database Transactions” on page 331 in 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`. Typical customers 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 “Important Changes for Java Code” on page 66.

### 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
<code>retainWhereKeys(keyFilter)</code>	Destructively removes all entries whose keys do not satisfy the <code>keyFilter</code> expression. Returns <code>true</code> 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 <code>true</code> or <code>false</code> .
<code>retainWhereValues(valueFilter)</code>	Destructively removes all entries whose values do not satisfy the <code>valueFilter</code> expression. Return <code>true</code> if this map changed as a result of the call. The value filter block must take one argument (a value) and return <code>true</code> or <code>false</code> .

Map method name	Description
<code>filterByKeys(keyFilter)</code>	Returns a new map that is a clone of the original map but without entries whose keys do not satisfy the <code>keyFilter</code> expression. The key filter block must take one argument (a key) and return true or false.
<code>filterByValues(valueFilter)</code>	Returns a new map that is a clone of the original map but without entries whose values do not satisfy the <code>valueFilter</code> 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 “Query Builder APIs” on page 125 in the *Gosu Reference Guide*.

## Entity Methods `loadByKey` and `loadByPublicID` Deprecated, New Method `loadBean`

The `Bundle` class methods `loadByKey` and `loadByPublicID` 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
```

### See Also

- “Query Builder APIs” on page 125 in the *Gosu Reference Guide*
- “Getting an Entity from a Public ID or a Key (Internal ID)” on page 336 in the *Gosu Reference Guide*

## For Various Types, Packages 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
<code>com.guidewire</code>	<code>gw.pl.exception</code>
<code>com.guidewire.commons.file</code>	<code>gw.pl.file</code>
<code>com.guidewire.commons.io</code>	<code>gw.pl.io</code>
<code>com.guidewire.commons.util</code>	<code>gw.pl.util</code>
<code>com.guidewire.logging</code>	<code>gw.pl.logging</code>
<code>com.guidewire.main</code>	<code>gw.pl.simpletool</code>
<code>com.guidewire.modules</code>	<code>gw.pl.modules</code>
<code>com.guidewire.modules.pom</code>	<code>gw.pl.modules.pom</code>
<code>com.guidewire.testharness</code>	<code>gw.pl.testharness</code>
<code>com.guidewire.util</code>	<code>gw.pl.util</code>
<code>com.guidewire.util.csv</code>	<code>gw.pl.util.csv</code>
<code>com.guidewire.util.diff.lcs</code>	<code>gw.pl.util.lcs</code>
<code>com.guidewire.util.webservices</code>	<code>gw.pl.util.webservices</code>
<code>com.guidewire.util.webservices.axis</code>	<code>gw.pl.util.webservices.axis</code>
<code>com.guidewire.util.webservices.login</code>	<code>gw.pl.util.webservices.login</code>
<code>com.guidewire.util.xml</code>	<code>gw.pl.util.xml</code>
<code>com.guidewire.util.xml.helpers</code>	<code>gw.pl.util.xml.helpers</code>
<code>com.guidewire.pl.plugin</code>	<code>gw.pl.plugin</code>
<code>com.guidewire.pl.plugin.addressbook</code>	<code>gw.pl.plugin.addressbook</code>
<code>com.guidewire.pl.plugin.approval</code>	<code>gw.pl.plugin.approval</code>
<code>com.guidewire.pl.plugin.credentials</code>	<code>gw.pl.plugin.credentials</code>
<code>com.guidewire.pl.plugin.custompicker</code>	<code>gw.pl.plugin.custompicker</code>
<code>com.guidewire.pl.plugin.dbauth</code>	<code>gw.pl.plugin.dbauth</code>
<code>com.guidewire.pl.plugin.document</code>	<code>gw.pl.plugin.document</code>
<code>com.guidewire.pl.plugin.management</code>	<code>gw.pl.plugin.management</code>
<code>com.guidewire.pl.plugin.messaging</code>	<code>gw.pl.plugin.messaging</code>
<code>com.guidewire.pl.plugin.search</code>	<code>gw.pl.plugin.search</code>
<code>com.guidewire.pl.plugin.security</code>	<code>gw.pl.plugin.security</code>
<code>com.guidewire.pl.plugin.util</code>	<code>gw.pl.plugin.util</code>
<code>com.guidewire.pl.plugin.validation</code>	<code>gw.pl.plugin.validation</code>
<code>com.guidewire.main</code>	<code>gw.pl.main</code>
<code>com.guidewire.external</code>	<code>gw.pl.external</code>

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 “Concurrency” on page 369 in 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 “Concurrency” on page 369 in the *Gosu Reference Guide*.

### **The `<>` 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 “Gosu Command Line Shell” on page 377 in 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 “Named Arguments and Argument Defaults” on page 98 in the *Gosu Reference Guide*.



# New and Changed in Integration in 8.0

This topic includes:

- “New in Integration in PolicyCenter 8.0” on page 59
- “Changed in Integration in PolicyCenter 8.0” on page 61

## New in Integration in PolicyCenter 8.0

### 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.

For more information, see “Phone Number Normalizer Plugin” on page 308 in the *Integration Guide*.

### For Free-Text Search, New Support for High Availability Using 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, you must make one minor change to the `solrserver-config.xml` file. In the `<solrserver>` element, change the `type` attribute to the new optional value `cloud`. Next, set the `host` and `port` attributes in child elements of type `<param>` to the Zookeeper instance host and port number respectively. As in previous releases, the `name` attribute of the `<solrserver>` element must match the `name` attribute of a `<document>` element in that same XML file.

For example:

```
<solrserver name="my_zookeeper_instance" type="cloud">
  <param name="host" value="zookeeper_host_name.mycompany.com"/>
  <param name="port" value="2181"/>
```

```
</solrserver>
...
<document name="policy" servername="my_zookeeper_instance"/>
```

The existing type values of http and embedded have their same behavior in PolicyCenter version 8.0 as in version 7.0.

## 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.

<b>Role of PolicyCenter</b>	<b>Sending or receiving</b>	<b>Version 7.0 support</b>	<b>Version 8.0 support</b>
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

For information about MTOM where Gosu is the SOAP client. See “MTOM Attachments with Gosu as Web Service Client” on page 94 in the *Integration Guide*.

For information about MTOM where Gosu is the SOAP server. See “WS-I Web Service Invocation Context” on page 53 in the *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 “Multi-threaded Inbound Integration” on page 321 in the *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 “Policy Earned Premium Web Services” on page 171 in the *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 “Implementing an Exchange Rate Service” on page 577 in the *Configuration 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

For more details, see “Message Ordering and Multi-Threaded Sending” on page 375 in the *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 “Loading WS-I WSDL Locally Using Studio Web Service Collections” on page 80 in the *Integration Guide*.

## Changed in Integration in PolicyCenter 8.0

### RPCE Web Services Deprecated in 8.0

PolicyCenter has two types of web services:

- **WS-I web services** – WS-I web services use SOAP protocol and are compatible with the WS-I standard.
- **RPC Encoded (RPCE) web services** – RPCE web services are an older style of web service publishing APIs. Versions of PolicyCenter before version 7.0 support only RPCE web service publishing.

In version 7.0, Guidewire recommended that you publish and consume web services using the WS-I APIs rather than the RPCE APIs. Although support for RPCE web services was for backward compatibility only, RPCE web services were not officially deprecated.

In version 8.0, RPCE web services are deprecated. RPCE web services continue to work in this release. However, it is strongly recommended to immediately start to convert RPCE code to the WS-I equivalent code.

### Built-in RPCE Web Services Converted to WS-I Web Services

In conjunction with deprecation of RPCE web services, some built-in RPCE web services now have WS-I equivalents. The WS-I versions are similar in structure as their RPCE equivalents. However, in some cases the parameter types changed to accommodate differences between RPCE and WS-I. For example, WS-I does not support

entity types directly as arguments or return types. In some cases, entity type arguments or return types are replaced with Gosu classes or XML objects that represent the same information.

The service name changed slightly for some web services, for example to remove the initial I from the name. The package names are different for the WS-I web services, and include the version number of PolicyCenter. As with all WS-I web services, use CNTL-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 p1/800, that version was introduced in PolicyCenter 8.0.0.

For a summary of changes to specific web services, including the name of any RPCE web services it replaces, see “Changes to Existing Web Services” on page 62.

## Changes to Existing Web Services

For details of converted WS-I web services, refer to the documentation topic for each web service as specified in the following table. Note that all RPCE web services are deprecated in version 8.0. Begin to convert any RPCE web service code to WS-I.

For any web services that converted from RPCE to WS-I, the WS-I versions use different types so there will be code changes. For example, any arguments or return values that took entity types now are WS-I variants based on Gosu classes with those fields or XSD types.

WS-I web service	Any RPCE web services that it replaces	Description and notable changes in version 8.0
AccountAPI	n/a	The <code>findAccounts</code> method takes a different type of object. The 7.0 version took an XSD type. The new version takes an account search criteria from a Gosu class definition of <code>AccountSearchInfo</code> . See the implementation class for details.
<code>ClaimToPolicySystemNotificationAPI</code>	n/a	Handles a notification from a claims system. For example, a claim amount exceeding a threshold level set for that type of policy. In 8.0, the <code>claimExceedsThreshold</code> method signature changed. The <code>grossTotalIncurred</code> parameter was a <code>String</code> , and now is a <code>MonetaryAmount</code> . See “Policy System Notifications” on page 557 in the <i>Integration Guide</i> .

WS-I web service	Any RPCE web services that it replaces	Description and notable changes in version 8.0
ContactAPI	n/a	<p>Manipulates contact data. For customers who use ContactManager, ContactManager calls this web service.</p> <ul style="list-style-type: none"> <li>In version 7.0, some methods took a transaction ID, which represents a unique identifier for this request. If the request is duplicated later, the application knows that it is a duplicate and does not repeat the request. In 8.0, the methods no longer have the transaction ID as an explicit method argument. Instead, the methods have the annotation <code>@WsICheckDuplicateExternalTransaction</code>, which automatically enforces unique transaction IDs based on SOAP headers. This change affects the methods <code>updateContact</code>, <code>removeContact</code>, and <code>mergeContacts</code>. See “Checking for Duplicate External Transaction IDs” on page 65 in the <i>Integration Guide</i>. If the web service client is another Guidewire application, you can set the transaction ID with the API <code>ContactAPIUtil.setTransactionId()</code>.</li> <li>New methods required by the <code>ABCClientAPI</code>, which the web service implements: <code>pendingUpdateRejected</code>, <code>pendingCreateRejected</code>, <code>pendingCreateApproved</code>, <code>pendingUpdateApproved</code>. These methods manage pending updates but the implementation is different for each Guidewire application.</li> </ul> <p>In PolicyCenter, PolicyCenter does not generate pending updates or creation for ContactManager, so this callback is not called in the default configuration of both applications. PolicyCenter throws an exception.</p>
CCPolicySearchIntegration	n/a	<p>Handles policy search requests from a claims system. In addition to conversion to WS-I, the method signature of <code>retrievePolicy</code> changed. The method now returns an object of type <code>Envelope</code> instead of <code>CCPolicy</code>. Within PolicyCenter, <code>Envelope</code> is an XSD type that encapsulates types to send to and from web services to claim systems. See “Policy Search Web Service (For Claim System Integration)” on page 557 in the <i>Integration Guide</i>.</p>
ImportToolsAPI	IImportToolsAPI IUserAPI IGroupAPI	<p>In addition to converting <code>IImportToolsAPI</code> from RPCE to WS-I, the <code>IImportToolsAPI</code> web service has new methods:</p> <ul style="list-style-type: none"> <li><code>importXml</code> – For administrative data only, this method imports data from XML conforming to either an archiving import xsd, or from <code>pc_import.xsd</code>.</li> <li><code>importArchiveXmlData</code> – new method to import administrative data from XML data in UTF-8 encoding of archiving data format XML.</li> <li><code>importArchiveXmlDataAsByteArray</code> – new method to import administrative data from XML data in a byte array of archiving data format XML</li> </ul> <p>From WS-I web services if you want to add a new user or group, use the WS-I web service <code>IImportToolsAPI</code>. Do not use the old web services specific to users and groups: <code>IUserAPI</code> and <code>IGroupAPI</code>. You can add a user or group with methods that import XML or CSV formatted text.</p> <p>For other methods in the RPCE web services <code>IUserAPI</code> and <code>IGroupAPI</code>, there is no equivalent WS-I web service. If you used <code>IUserAPI</code> or <code>IGroupAPI</code>, Guidewire recommends that you write your own custom WS-I web service that transfers only the necessary data for each integration point.</p>

WS-I web service	Any RPCE web services that it replaces	Description and notable changes in version 8.0
LoginAPI	ILoginAPI (however, see note)	<p>Although the names are similar, the RPCE web service ILoginAPI and the WS-I web service LoginAPI are actually notably different.</p> <p>The RPCE web service ILoginAPI had a critical role in authenticating with the server for later API calls with other web services. If you use the generated Java libraries and the included APILocator utility class, APILocator indirectly uses ILoginAPI.</p> <p>The WS-I web service LoginAPI does not have the analogous role in WS-I authentication. In effect, WS-I authentication happens with each API call. However, if your web service client code wants to explicitly test specific authentication credentials, PolicyCenter publishes the built-in LoginAPI web service. Call this web service's login method, which takes a user name as a String and a password as a String. If authentication fails, the API throws an exception. You can also use LoginAPI to purposely leave a user session open for logging purposes.</p> <p>See "Login WS-I Authentication Confirmation" on page 73 in the <i>Integration Guide</i>.</p>
MaintenanceToolsAPI	IMaintenanceToolsAPI	<p>Tools related to managing the application. There are various changes in 8.0 in addition to conversion to WS-I:</p> <ul style="list-style-type: none"> <li>• terminateBatchProcessByName changed name to terminateBatchProcessByName</li> <li>• terminateBatchProcessByID changed name to requestTerminationOfBatchProcessByID</li> <li>• stopWorkQueueWorkers – new method to stop query workers</li> <li>• getWQueueStatus – new method that replaces getWorkQueueStatus.</li> <li>• getNumActiveWorkItems – new method to get number of active work items.</li> <li>• new methods for archiving: startValidateArchiveLinksBatchProcessAll, startValidateArchiveLinksBatchProcessByDateRange, startValidateArchiveLinksBatchProcessByIds</li> </ul> <p>See "Maintenance Web Services" on page 154 in the <i>Integration Guide</i>.</p>
MessagingToolsAPI	IMessagingToolsAPI	<p>Tools related to messaging, such as suspending a destination. There are various changes in 8.0 in addition to conversion to WS-I:</p> <ul style="list-style-type: none"> <li>• configureDestination – new method to configure some performance and error handling aspects of a messaging destination on a running server. Also see the getConfiguration method.</li> <li>• getConfiguration – get performance and error handling parameters from a messaging destination on a running server. Also see the configureDestination method.</li> </ul>
PolicyRenewalAPI	n/a	<p>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.</p> <p>See "Messaging Tools Web Service" on page 397 in the <i>Integration Guide</i>.</p> <p>APIs related to policy renewal. In the notifyPaymentReceivedForRenewalOffer method, the BigDecimal argument changed to MonetaryAmount. See "Policy Renewal Web Services" on page 172 in the <i>Integration Guide</i>.</p>

WS-I web service	Any RPCE web services that it replaces	Description and notable changes in version 8.0
ProducerAPI	ProducerAPI (same name but different package for RPCE version)	<p>Producer related actions. There are various changes in 8.0 in addition to conversion to WS-I:</p> <ul style="list-style-type: none"> <li>• uses of the Organization entity are now OrganizationDTO, which is a Gosu class in package <code>gw.webservice.pc.pcVERSION.community.datamodel</code>.</li> <li>• <code>getOrganizationByPublicId</code> – method changed names to <code>getAgencyOrganization</code></li> <li>• <code>createOrganization</code> – new method to create an organization</li> <li>• The synchronize methods <code>synchronizeAgency</code>, <code>synchronizeBranch</code>, and <code>synchronizeProducerCode</code> – removed, replaced by new method <code>updateOrganization</code></li> <li>• <code>getGroupByPublicId</code> – new method to get a group by public ID</li> <li>• <code>addNewGroupToOrganization</code> – new method to add a new group to an organization</li> <li>• <code>updateGroup</code> – new method to update a group</li> <li>• <code>getProducerCodeByPublicId</code> – new method to get producer code by public ID</li> <li>• <code>addNewProducerCodeToOrganization</code> – new method to add a new producer code to an organization</li> <li>• <code>updateProducerCode</code> – new method to update and commit a producer code based on producer code XML.</li> <li>• <code>getBranch</code> – old method removed</li> <li>• <code>moveProducerCodeToExistingAgency</code> – old method removed</li> <li>• <code>moveProducerCodeToSyncedAgency</code> – old method removed</li> <li>• <code>moveProducerCodeToExistingBranch</code> – old method removed</li> <li>• <code>moveProducerCodeToSyncedBranch</code> – old method removed</li> <li>• <code>removeProducerCodeFromExistingAgency</code> – old method removed</li> </ul> <p>See “Producer Web Services” on page 166 in the <i>Integration Guide</i>.</p>
ProfilerAPI	IProfilerAPI	<p>There are various changes in 8.0 in addition to conversion to WS-I:</p> <ul style="list-style-type: none"> <li>• <code>setEnableProfilerForBatchProcess</code> – the method argument of type <code>BatchProcessType</code> now is <code>String</code></li> <li>• <code>setEnableProfilerForWorkQueue</code> – the method argument of type <code>BatchProcessType</code> now is <code>String</code>.</li> </ul> <p>See “Profiling Web Services” on page 157 in the <i>Integration Guide</i>.</p>

<b>WS-I web service</b>	<b>Any RPCE web services that it replaces</b>	<b>Description and notable changes in version 8.0</b>
SystemToolsAPI	ISystemToolsAPI	<p>There are various changes in 8.0 in addition to conversion to WS-I:</p> <ul style="list-style-type: none"> <li>getDBCState – new method that returns the state of the consistency checks process.</li> <li>submitDBCCBatchJob – new method that submits the consistency checks batch job on the underlying physical database</li> <li>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.</li> <li>cancelUpdateStats – new method that cancels the update stats process if running. Afterwards, use the getUpdateStatsState method to check on the process status.</li> <li>getUpdateStatsState – new method that checks whether the update stats process is currently running.</li> <li>submitDmvReportBatchJobWithParams - the previous version used to take three parameters, and now takes only the includeDatabaseStatistics argument.</li> <li>getRecentAWRSnapshotInfo - method now returns DatabaseSnapshotInfo[] instead of String</li> <li>updateLoggingLevel -- logging level was passed as String, now is LoggingLevel</li> <li>submitAwrReportBatchJob and submitAwrReportBatchJobAll10pts – new method to submit Oracle AWR Performance Report batch jobs.</li> <li>getPerfReportsInfo – Retrieve the information about the latest Oracle AWR downloads.</li> </ul> <p>See “System Tools Web Services” on page 155 in the <i>Integration Guide</i></p>
TemplateToolsAPI	ITemplateToolsAPI	<p>Validates document templates. In addition to the conversion to WS-I, there are additional differences:</p> <ul style="list-style-type: none"> <li>validateTemplateInLocale – a new method to validate one template using a specific locale code</li> <li>validateAllTemplatesInLocale – a new method to validate all templates using a specific locale code</li> </ul> <p>See “Template Web Service APIs” on page 280 in the <i>Integration Guide</i>.</p>
TypeListToolsAPI	ITypeListToolsAPI	<p>“Mapping Typecodes to External System Codes” on page 151 in the <i>Integration Guide</i></p>
WorkflowAPI	IWorkflowAPI	<p>There are various changes in 8.0 in addition to conversion to WS-I:</p> <ul style="list-style-type: none"> <li>invokeTrigger - method argument that used to be type WorkflowTriggerKey now is String.</li> <li>isTriggerAvailable – method argument that used to be type WorkflowTriggerKey now is String</li> </ul> <p>See “Workflow Web Services” on page 156 in the <i>Integration Guide</i>.</p>

## Important Changes for Java Code

PolicyCenter 8.0 made significant changes to how 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 69.

- 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 “Deploying Non-OSGi Java Classes and JARs” on page 674 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 “Adding a New Plugin Interface Implementation” on page 108 in 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 71.
  - For more about plugin directories, see “Adding a New Plugin Interface Implementation” on page 108 in 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 “Inspections to Flag Unsupported Internal Java APIs” on page 659 in 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 69.

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 70.
- 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 70.
- 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 70.
- 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 70.
- 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, it 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 69
- “Java Library Generation in Version 8.0” on page 69
- “Getting a Reference to a Bundle in Java” on page 70
- “Creating Entities in the PolicyCenter 8.0 Java API” on page 70
- “Calling Gosu Classes from Java” on page 70
- “Gosu Enhancements on Java Classes or Gosu Classes” on page 70

## 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 “Inspections to Flag Unsupported Internal Java APIs” on page 659 in 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 “Accessing Entity and Typecode Data in Java” on page 661 in 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:

- “Accessing Entity and Typecode Data in Java” on page 661 in the *Integration Guide*
- “Entity Packages and Customer Extensions from Java” on page 662 in the *Integration Guide*
- “Typecode Classes from Java” on page 666 in the *Integration Guide*.

**WARNING** The Java entity interfaces are supported only in Java code. From Gosu, use the syntax `entity.ENTITYNAME`, or where unambiguous you can 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 “Regenerating Java API Libraries” on page 661 in 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 71. For more about plugin directories, see “Adding a New Plugin Interface

Implementation” on page 108 in 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 “Bundles and Database Transactions” on page 331 in 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 “Getting a Reference to an Existing Bundle in Java” on page 670 in 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 “Creating New Entity Instances from Java” on page 670 in 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 “Accessing Gosu Classes from Java Using Reflection” on page 672 in 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 “Gosu Enhancement Properties and Methods in Java” on page 673 in 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 “Typecode Classes from Java” on page 666 in 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 “Query Builder APIs” on page 125 in the *Gosu Reference Guide*.

For minor differences in the Gosu and Java APIs for querying, see “Querying for Entity Data in Java” on page 672 in 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 “Adding a New Plugin Interface Implementation” on page 108 in 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. See “Using the Plugins Registry Editor” on page 107 in the *Configuration Guide*.

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. For a complete reference, see “Where to Deploy Your Java Classes and Libraries” on page 71. For the 8.0 Java API, also see “Deploying Non-OSGi Java Classes and JARs” on page 674 in the *Integration Guide*.

## Changes to Importing the PolicyCenter Product Model into ClaimCenter

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.

To learn how to run the ClaimCenter Typelist Generator in PolicyCenter 8.0, see “Running the ClaimCenter Typelist Generator” on page 564 in the *Integration Guide*.

## Changes to Document Management Integration

### 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, see the *Guidewire Platform Support Matrix*, available from the Guidewire Resource Portal at <https://guidewire.custhelp.com/app/resources/products/platform>.

It is no longer necessary to make PCF edits to the file `NewTemplateDocumentDV.pcf` when enabling or disabling the Document Assistant.

#### See also

- “Configuring Guidewire Document Assistant” on page 121 in the *System Administration Guide*
- “Document Production” on page 261 in the *Integration Guide*

### 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.

#### Changes to Plugins and Classes for Line-of-business Modularization

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*.

- “Lines of Business Modularization” on page 45
- “Guidelines for Modularizing Line-of-business Code” on page 443 in the *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

- “Policy Period Plugin” on page 212 in the *Integration Guide*
- “Policy Term Plugin” on page 211 in the *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 `LOBPolicyLineMethods`. See the method `createRatingEngine`.

## Changes to Policy Period Diff Plugin

The `PolicyPeriodDiffPlugin` plugin used to rely upon line-of-business abbreviations to return `lobDiffTree.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

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

- “PolicyCenter Product Model Import into ClaimCenter” on page 560 in the *Integration Guide*

## Contact Messaging Flow Changed

The messaging flow changed for contact-related messages to external systems. There are different methods on the `ContactSystemPlugin` interface. For more information about the new messaging flow for contacts, see “Integrating with a Contact Management System” on page 573 in the *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.

- `GWClaimSearchPlugin`
- `ProrationPlugin`
- `PCReinsuranceCedingPlugin`
- `PCReinsurancePlugin`
- `ReinsuranceConfigPlugin`
- `LossHistoryPlugin`

For customers who upgrade, fix any compilation errors in plugin implementations to use `MonetaryAmount` as required by the new versions of the plugin interfaces.

## Free-text Search Upgrades

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 7.0 is based on Apache Solr 3.

### To upgrade the Guidewire Solr Extension from 7.0 to 8.0

1. Back up the existing index from your current installation of the Guidewire Solr Extension.
2. Install the Guidewire Solr Extension included with PolicyCenter 8.0, from the distribution file `PolicyCenter/solr/pc-gwsolr.zip`.  
For complete instructions, see “Free-text Search Setup” on page 85 in the *Installation Guide*.
3. Restore the existing index from your backup location to your Guidewire Solr Extension hierarchy
4. Start the Guidewire Solr Extension.
5. In a browser, open the administrative user interface for the Guidewire Solr Extension, by entering the following URL:  
`http://hostName:8983/pc-gwsolr`
6. Issue the Solr command to upgrade the index files.

## New PCPolicyPublicID Property in PolicyCenter-BillingCenter Integration

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.

# New and Changed in System Administration in 8.0

## New in System Administration in 8.0

This topic describes what is new for system administration of PolicyCenter 8.0. This includes:

- “Phone Number Normalizer Work Queue” on page 77

### 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 “Upgrading Phone Numbers” on page 172 in the *Upgrade Guide*.

## Changed in System Administration in 7.0

This topic describes what has changed for system administration of PolicyCenter 8.0. This includes:

### 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 `<oracle-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 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 `database-config.xml`. Specify `false` to gather full database statistics.

## Cancelling Database Statistics Updating

You can now cancel the database statistics updating process while it is running. This capability comes with changes to how database statistics updating is launched.

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 distributed 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 Process Now Named Team Screens

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 33
- “Team Screens” on page 113 in the *System Administration Guide*

# New and Changed in Rules in 8.0

This topic covers what is new and changed with the business rules in PolicyCenter 8.0.

This topic includes:

- “Changes in Rules in PolicyCenter 8.0” on page 81

## Changes in Rules in PolicyCenter 8.0

This topic describes what has changed with rules in PolicyCenter 8.0.

### Preupdate Rules Removed

The rules in the Preupdate rule set were deprecated in PolicyCenter 2.0.4. In PolicyCenter 8.0, the Preupdate rule set has been removed.

Guidewire expects you to implement preupdate functionality using the Guidewire-provided preupdate classes and your own custom classes. However, if you prefer to use this rule set, you can configure PolicyCenter to use it. In `config.xml`, set the `UseOldStylePreUpdate` parameter to `true`.

#### See also

- “Preupdate Handler Plugin” on page 310 in the *Integration Guide*

### Evaluation Rule Set Functionality Replaced

In PolicyCenter 8.0, the functionality in the Evaluation rule set has been moved to Gosu classes. The Evaluation rule set contained the Policy Evaluation (New) and Segmentation rules. The rule set remains, but the rules in the rule set have been removed from the product.

#### Policy Evaluation (New) Rules Replaced by Evaluator Classes

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`)

Guidewire expects you to implement any necessary evaluation functionality using the Guidewire-provided evaluator classes and your own custom classes. However, if you prefer to use this rule set, you can configure PolicyCenter to use it.

**See also**

- “Configuring Underwriting Issues” on page 479 in the *Configuration Guide*
- “Checking Sets and Evaluators” on page 481 in the *Configuration Guide*

### [Segmentation Rules Replaced by Segment Evaluator Classes](#)

In PolicyCenter 7.0, the Segmentation rules determined the segmentation, and, therefore, the available underwriting companies for a policy. In PolicyCenter 8.0, these rules have been replaced by the following Gosu classes:

- `SegmentEvaluator` – The interface
- `AbstractSegmentEvaluator` – Implements the `SegmentEvaluator` interface
- Subclasses – `DefaultSegmentEvaluator` and line specific ones like `WC_SegmentEvaluator` that is created from `PolicyLineMethods#createSegmentEvaluator`.

Guidewire expects you to implement any necessary segmentation functionality using the Guidewire-provided classes (described above) and your own custom classes. However, if you prefer to use this rule set, you can configure PolicyCenter to use it.

**See also**

- “Selecting the Underwriting Company through Segmentation” on page 589 in the *Configuration Guide*

# New and Changed in Globalization in 8.0

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.

This topic includes:

- “New to Globalization in PolicyCenter 8.0” on page 83
- “Changes to Globalization in PolicyCenter 8.0” on page 90

## New to Globalization in PolicyCenter 8.0

The following globalization-related features are new in PolicyCenter 8.0:

- “Separation of Language and Regional Format” on page 84
- “Base Configuration Language Support” on page 84
- “Base Configuration Region Support” on page 84
- “Base Configuration Currency Support” on page 85
- “Base Configuration Geographical Support” on page 85
- “Phone Number Data and Phone Validator Gosu Class” on page 86
- “Automatic Address Completion and Fill-in Plugin Functionality” on page 87
- “Ability to Provide a Sort Order for the Typecodes in a Typelist” on page 87
- “Language Module Installation Utility” on page 88
- “Globalization-related Configuration Parameters” on page 88
- “Globalization-related Typelists” on page 88
- “Ability to Set First Day of Week by Region” on page 88

- “New in the PolicyCenter 8.0 Data Model” on page 89

## 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, which is English. If your installation has more than one language, then you can select among the multiple languages using the PolicyCenter **Options** → **Language** submenu. 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 language only. Each base configuration, each `localization` file resides in the following location in Studio:

`configuration` → `Localizations` → `xx_yy`

`xx_yy`, in this case, is a standard ISO locale code.

In a language module, `localization.xml` resides in `config/locale.xx_yy`, with `xx_yy` again being the ISO locale code.

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, then 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

- “Working with Languages” on page 21 in the *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**

- “Working with Regional Formats” on page 71 in the *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 → xxx`

For example, the `currency.xml` file for the Australian dollar resides in the following location in Studio:

`configuration → config → currencies → aud`

The effect of this change is to reduce merge conflicts during upgrade.

The following table describes the use of file `currency.xml` in single or multiple currency rendering mode.

Currency rendering mode	Use of currency files
Single currency rendering mode	If operating in single currency rendering mode, then you do not use <code>currency.xml</code> files. Instead, use file <code>localization.xml</code> to provide information for formatting the default application currency.
Multiple currency rendering mode	If operating in multiple currency rendering mode, then PolicyCenter ignores file <code>localization.xml</code> . Instead, the PolicyCenter installation contains multiple <code>currency</code> files, each of which contains formatting information for a specific currency alone.

**See also**

- “Configuring Currencies” on page 87 in the *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 – United 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` file into multiple `address-config` files..

PolicyCenter stores the `address-config` files in individual country folders (`xx`) in Studio:

`configuration → config → geodata → xx`

For example, the `address-config.xml` file for the address-related information in Australia resides in the following location in Studio:

`configuration → config → geodata → AU`

#### **See also**

- “Configuring Address Information” on page 103 in the *Globalization Guide*

### [New Multiple country.xml Files](#)

New in 8.0 are `country.xml` files that sets address formatting as it pertains to 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` in its own country folder (`xx`) in Studio:

`configuration → config → geodata → xx`

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**

- “Configuring Address Information” on page 103 in the *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 splits the single `zone-config` file into multiple `zone-config` files, each file for use in defining such items as the address autofill fields to use.

PolicyCenter stores the `zone-config` files in individual country folders (`xx`) in Studio:

`configuration → config → geodata → xx`

For example, the `zone-config.xml` file for the zone information in Australia resides in the following location in Studio:

`configuration → config → geodata → AU`

#### **See also**

- “Configuring Zone Information” on page 97 in the *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`

To enable the new PhoneValidator validation functionality, you need to configure `fieldvalidators.xml` with the fully qualified name of the Gosu class, for example:

```
<ValidatorDef description="Validator.Phone"
    name="LocalizedPhoneValidator"
    validation-type="gosu"
    value="gw.api.validation.PhoneValidator"/>
```

Also, you must set the validation type to `gosu`:

```
validation-type="gosu"
```

The validator does not trigger validation unless the associated phone country is set. This trigger functionality provides backwards compatibility with old data.

## 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 autocomplete 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 autfills 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 autfill 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 Auto-Complete 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 auto-completion 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 8.0” on page 91.

### See also

- “Automatic Address Completion and Fill-in” on page 112 in the *Globalization Guide*
- “Automatic Address Completion and Fill-in Plugin” on page 307 in the *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

- “Installing Display Languages” on page 23 in the *Globalization Guide*
- “Key PolicyCenter `gwpc` Commands” on page 59 in the *Installation 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

- “Globalization Parameters” on page 56 in the *Configuration Guide*

## Globalization-related Typelists

The following typelists related to globalization are new or have a new function in PolicyCenter 8.0.

Typelist	Description
<code>LocaleType</code>	Use to define supported regions. The value of configuration parameter <code>DefaultApplicationLocale</code> must be a typecode from the <code>LocaleType</code> typelist.
<code>LanguageType</code>	Use to define supported languages. The value of configuration parameter <code>DefaultApplicationLanguage</code> must be a typecode from the <code>LanguageType</code> typelist.  Typelist <code>LanguageType</code> 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 <code>LanguageType</code> 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	<ul style="list-style-type: none"><li>• en_AU</li><li>• en_CA</li><li>• en_US</li><li>• fr_CA</li><li>• ja_JP</li></ul>
Monday	<ul style="list-style-type: none"><li>• de_DE</li><li>• en_GB</li><li>• fr_FR</li></ul>

#### See also

- For more information how PolicyCenter uses the ICU library, see “Configuring Locale Codes for Default Application Locale and the ICU Library” on page 72 in 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 89
- “New Globalization-related Fields on Contact and Address Entities” on page 89
- “Monetary Amount Data Model Element in PolicyCenter 8.0” on page 89
- “Gosu MonetaryAmount Entity in PolicyCenter 8.0” on page 90
- “Monetary Amount Widgets in PolicyCenter 8.0” on page 90
- “Attributes on <localization> Column Element” on page 90

### 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	<ul style="list-style-type: none"><li>• FaxPhoneCountry</li><li>• HomePhoneCountry</li></ul>
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
<code>MonetaryAmountCell</code>	Use to display <code>MonetaryAmount</code> values in PolicyCenter
<code>MonetaryAmountInput</code>	Use to enter <code>MonetaryAmount</code> 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
<code>unique</code>	<code>Boolean</code>	If you set this attribute to <code>true</code> , PolicyCenter prohibits duplicate values in the set of value for each language.
<code>nullok</code>	<code>Boolean</code>	<p>If you set this attribute to <code>true</code>, PolicyCenter flags missing entries that it finds during a database consistency check.</p> <p>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.</p> <p>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.</p>

## Changes to Globalization in PolicyCenter 8.0

The following globalization-related features are changed in PolicyCenter 8.0:

- “Changes to Address Configuration 8.0” on page 91

- “Changes to Admin Data Localization” on page 91
- “Changes to 8.0 Typelists” on page 92
- “Changes to Files Used with Globalization” on page 92

## Changes to Address Configuration 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	Used with...
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

- See “Configuring Address Information” on page 103 in the *Globalization Guide* for more information.

## 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 8.0 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 Contact Manager. 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.
Jurisdiction	In previous releases, Guidewire overloaded the use of the State typelist to represent a jurisdiction as well as a geographical area. In PolicyCenter 8.0, Guidewire de-couples the concepts of state and jurisdiction. State and jurisdiction are separate concepts, and PolicyCenter represents them independently with the State and Jurisdiction typelists.  Guidewire now represents entity fields that were previously of type State as type Jurisdiction.
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.
State	Guidewire has modified the base configuration State typelist to include state definitions for the following countries: <ul style="list-style-type: none"><li>• AU</li><li>• CA</li><li>• DE</li><li>• JP</li><li>• US</li></ul>
StateAbbreviation	Guidewire has added the StateAbbreviation typelist in 8.0. This typelist includes abbreviations for states in the following countries: <ul style="list-style-type: none"><li>• Australia</li><li>• Germany</li><li>• United States</li></ul> 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

- “Configuring Geographic Data” on page 95 in the *Globalization Guide*
- “Working with Regional Formats” on page 71 in the *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 changed 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: <b>configuration → config → geodata → xx</b> Substitute xx with a standard ISO country code. In the base PolicyCenter 8.0 release, Guidewire provides support for the following countries: <ul style="list-style-type: none"><li>• AU</li><li>• CA</li><li>• DE</li><li>• FR</li><li>• GB</li><li>• JP</li><li>• US</li></ul>
config.xml	Globalization parameters new in PolicyCenter 8.0: <ul style="list-style-type: none"><li>• DefaultApplicationLanguage</li><li>• DefaultPhoneCountryCode</li><li>• DefaultNANPCCountryCode</li><li>• AlwaysShowPhoneWidgetRegionCode</li></ul> Globalization parameters with modified functionality in PolicyCenter 8.0: <ul style="list-style-type: none"><li>• DefaultApplicationLocale</li><li>• MulticurrencyDisplayStyle</li></ul>
	<b>See also</b> <ul style="list-style-type: none"><li>• “Globalization Parameters” on page 56 in the <i>Configuration Guide</i></li></ul>
language.xml	This is an optional file. 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.
localization.xml	In 8.0, Guidewire splits file localization.xml into multiple files, each of which contains formatting and parsing rules for a single language only. Each base configuration localization.xml file resides in the following location in Studio: <b>configuration → Localizations → xx_yy</b> Substitute xx_yy with a standard ISO locale code. 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.
	<b>See also</b> <ul style="list-style-type: none"><li>• “Configuring Locale Codes for Default Application Locale and the ICU Library” on page 72 in the <i>Globalization Guide</i></li></ul>
nanpa.properties	Area codes defined by the North American Numbering Plan Administration (NANPA). You find this file in Studio in the following location: <b>configuration → config → phone</b>
	<b>See also</b> <ul style="list-style-type: none"><li>• “Working with Phone Configuration Files” on page 113 in the <i>Globalization Guide</i></li><li>• “Setting Phone Configuration Parameters” on page 114 in the <i>Globalization Guide</i></li></ul>



# What's New and Changed in 7.0.0

This part includes information about new and changed features in PolicyCenter 7.0. For information about new and changed features in 7.0 maintenance releases, see “Release Notes Archive” on page 161.



# New and Changed in PolicyCenter 7.0

This document describes the new and changed features in PolicyCenter 7.0.

This topic includes:

- “PolicyCenter Lines of Business” on page 97
- “PolicyCenter Jobs” on page 99
- “PolicyCenter Components” on page 100
- “Administering PolicyCenter” on page 105
- “Product Integrations” on page 106

## PolicyCenter Lines of Business

PolicyCenter 7.0 includes the following new and changed features in PolicyCenter lines of business:

- “New Commercial Property Blankets” on page 97
- “Copying Coverages” on page 98
- “Changes to Personal Auto” on page 98
- “Business Auto Becomes Commercial Auto” on page 98

### New Commercial Property Blankets

Blanket insurance is available in the commercial property line of business. Blanket insurance provides coverage for a combination of items with a single limit for all the covered items that are included in the blanket. The **Blankets** screen displays blankets applied to the policy. Blankets are grouped into two types: direct loss and time element.

#### See also

- “Blankets Screen for Commercial Property” on page 217 in the *Application Guide*

## Copying Coverages

In the personal auto and commercial property lines of business, you can now copy coverages from a particular vehicle or building to one or more vehicles or buildings, respectively. In personal auto, the **Copy Coverages** button is on the **PA Coverages** screen. In commercial property, the **Copy Coverages** button is on the **Building and Locations** screen.

Through configuration, you can add copy coverages to other lines of business.

### See also

- “Copying Coverages to Other Vehicles” on page 258 in the *Application Guide*
- “Configuring Copy Coverages for Personal Auto” on page 273 in the *Application Guide*
- “Copying Coverages to Other Buildings” on page 214 in the *Application Guide*
- “Configuring Copying of Coverages for Commercial Property” on page 227 in the *Application Guide*

## Changes to Personal Auto

In this version of PolicyCenter, a number of changes were made to the personal auto line of business.

### Motor Vehicle Records in Personal Auto

In PolicyCenter 7.0, motor vehicle records have been added for personal auto. A motor vehicle record (MVR) documents a driver’s driving history. The MVR report contains information such as identifying data, license status, convictions, traffic violations, accidents, license suspensions, and revocations on file with the driver’s home state. In the U.S., the information in this report usually comes from the Department of Motor Vehicles (DMV) for each state. The information in the report can vary by state. In the U.S., most service providers provide MVR data for all states, so that you only need to integrate with a single service provider.

The personal auto line of business provides MVR reports in the default configuration.

### See also

- “Motor Vehicle Records in Personal Auto” on page 256 in the *Application Guide*

### Changes to Quick Quote in Personal Auto

In personal auto, **Quick Quote** is now even quicker with only two screens:

- A data entry screen that includes:
  - A few key policy information fields
  - Space for two drivers and two vehicles
- A quote screen that includes offering selection, coverage selection and quote results. When you enter this screen, coverages default to values specified in the default offering selection.

### Excluded Drivers in Personal Auto

You can now exclude a driver on a personal auto policy. For example, you might want to exclude drivers with bad driving records. When you exclude a driver, PolicyCenter adds a form to the policy based on the exclusion, and you cannot assign the excluded driver to a vehicle. PolicyCenter creates an underwriting issue if an excluded driver exists on a policy.

## Business Auto Becomes Commercial Auto

The business auto line of business has been renamed commercial auto. Entity and field names have not changed. For example, the entity for the policy line is still **BusinessAutoLine** with field **BACosts**. The **BusinessAutoLine**

entity has a new `PolicyType` field. In Studio, business auto has been renamed commercial auto in the product model and in display keys.

In addition, the following changes were made:

- Commercial auto now has line-level coverages and an associated wizard screen. The liability, hired auto, and non-owned auto coverages are now line level coverages instead of state level coverages.
- The PIP coverages have been reorganized, including moving them to the jurisdiction level instead of vehicle level.
- The `BAJurisdiction` entity now delegates to `Modifiable`. Some modifiers have been moved to `BAJurisdiction`.
- The `BusinessVehicle` entity now delegates to `Modifiable`. Some data fields are now modifiers in the product model.

## PolicyCenter Jobs

PolicyCenter 7.0 includes the following new and changed features in PolicyCenter jobs:

- “Side-by-side Quoting” on page 99
- “Rescheduling a Cancellation” on page 99
- “Changing the Effective Date of a Policy Change” on page 99
- “Rewrite New Account Job” on page 100

### Side-by-side Quoting

In PolicyCenter 7.0, side-by-side quotes has been added. With side-by-side quoting, you can view multiple versions of a policy job on one screen. You can modify the coverages and terms of each version in the side-by-side screen, and see the side-by-side comparison of the costs and benefits of each version. You can use side-by-side quoting with quick quote. In the default configuration, side-by-side quoting is supported in the personal auto line of business. You can configure side-by-side quoting for other lines of business.

#### See also

- “Side-by-side Quoting” on page 157 in the *Application Guide*

### Rescheduling a Cancellation

PolicyCenter now enables you to reschedule a cancellation by changing the cancellation effective date of an open cancellation.

#### See also

- “Changing a Cancellation” on page 105 in the *Application Guide*
- “Changing the Cancellation Effective Date on an Open Cancellation” on page 109 in the *Application Guide*

### Changing the Effective Date of a Policy Change

You can now edit the effective date of an unbound policy change. However, after the policy is bound and issued, you can no longer edit the effective date.

Editing the effective date of a policy change is often useful in personal lines of business. A policyholder calls to make a policy change. The agent starts a policy change, enters new policy information and generates a quote. Later, the customer calls requesting a change to the effective date of the policy change.

**See also**

- “Editing the Effective Date of a Policy Change” on page 114 in the *Application Guide*
- “Editing the Policy Change Effective Date” on page 119 in the *Application Guide*

## Rewrite New Account Job

In PolicyCenter 7.0, there is a new rewrite new account job. When you rewrite policies to a new account, PolicyCenter creates rewrite new account jobs for each policy. This job takes data from an existing policy and creates a new policy with a new policy number in the new account. Unlike a rewrite job, a rewrite new policy job can have pre-qualification questions. You can only rewrite canceled or expired policies to a new account.

**See also**

- “Moving or Rewriting Policies Between Accounts” on page 102
- “Rewrite New Account Policy Transaction” on page 131 in the *Application Guide*

# PolicyCenter Components

PolicyCenter 7.0 includes the following new and changed features in PolicyCenter components:

- “Changes to Activities” on page 100
- “Changes to Account History Screen and Events” on page 101
- “Account Relationships” on page 101
- “Moving or Rewriting Policies Between Accounts” on page 102
- “Merging Accounts” on page 102
- “Copying Data Between Policies” on page 103
- “Splitting and Spinning Policies” on page 103
- “Contact Tab” on page 103
- “Linking Addresses on Contacts” on page 104
- “Updating Contacts in Effective Time” on page 104
- “Reinsurance” on page 104
- “Change to End Date Field in Underwriting Entities” on page 105

## Changes to Activities

In PolicyCenter 7.0, there are a number of changes to activities.

### Attach Activities to Policies

Activities can now be directly attached to policies. Previously, activities were only associated with jobs and accounts in the default configuration.

### View Notes for an Activity

In the **Activity Detail** tab, there is a new **View Notes** button. Click this button to view all notes attached to the current activity. This button is visible if you have the **View notes** permission. The code for this permission is `noteview`.

**See also**

- “Activities” on page 359 in the *Application Guide*

## Changes to Account History Screen and Events

In the account, there is now a **History** screen that displays events associated with the account and policies in the account.

The new history events associated with accounts are:

- **Account created** – A history event is created when new accounts are created through the user interface or programmatically through the `AccountAPI`.
- **Account changed** – PolicyCenter creates this event when:
  - The account status changes from **Pending** to **Active** at the start of a submission.
  - The account holder changes to a different contact. The event contains the old and new values.

You can add additional history events by configuring PolicyCenter.

### See also

- “Account File History Screen” on page 327 in the *Application Guide*
- “History Events in the Default Configuration” on page 587 in the *Configuration Guide*

## New Methods for History

There are two new enhancement methods for creating history entries on `Job`, `PolicyTerm`, and `Account` entities.

- The first method creates a `History` with the given `CustomHistoryType` and `description`. The `description` argument is wrapped in a block so that it can be localized to the primary language of the policy or account.

```
function createCustomHistoryEvent(type : CustomHistoryType, description : block() : String) : History
```
- The second method is similar, but allows you to provide the `originalValue` and `newValue` fields of the `History`.

```
function createCustomHistoryEvent(type : CustomHistoryType, description : block() : String, originalValue : String, newValue : String) : History
```

### Deprecated Methods

The following methods have been deprecated in favor of the methods above.

In the `Job` class, the two methods below are deprecated. The first method takes a simple `String` argument for the `description` instead of a `block`. The second method has an inconsistent method name and no return value:

```
function createCustomHistoryEvent(type : CustomHistoryType, description : String) : History  
function createHistoryEvent(type : CustomHistoryType, description : block() : String)
```

Similarly, in the `PolicyTerm` class, the method which takes a `String` `description` is also deprecated.

```
function createCustomHistoryEvent(type : CustomHistoryType, description : String) : History
```

### See also

- “Configuring Job History Events” on page 586 in the *Configuration Guide*

## History Search Criteria Class

The `HistorySearchCriteria` entity has been converted to an exported Gosu class `gw.history.HistorySearchCriteria`.

## Account Relationships

In PolicyCenter 7.0, you can now associate accounts with one another. In the default configuration, the account relationships are:

- **Parent and child** – Use this account relationship for hierarchical accounts, such as a corporate parent and subsidiaries.

- **Common owner** – Use this account relationship to link commercial accounts that have a common owner. The common owner might be a person or a corporate entity. For example, you can associate all accounts for companies owned by one holding company, even though there is no account for the parent company.

You can modify the existing account relationships or create your own account relationship types.

With account relationships, you can also search for accounts with a shared contact. In the default configuration, this search finds accounts that have an account holder or named insured in common. The contact in common does not have to be a named insured or account holder on both accounts. For example, if a contact is a account holder on one account and a named insured in another, these accounts are related. You can modify or add to the search criteria.

#### See also

- “Related Accounts Overview” on page 322 in the *Application Guide*
- “Configuring Account Relationships” on page 335 in the *Application Guide*

## Account Merge Changes

Changes were made to the account merge functionality to handle account relationships. When merging account B into account A, all the relationships of B reference A with the following exceptions:

- The merge removes any relationship between A and B. This prevents self-referential relationships. For example, if A is the parent of B and B is merged with A, the merge removes the parent to child relationship.
- The merge removes any relationship of B which would create a duplicate of an existing relationship on A after the merge. For example, if A and B are both children of C, then merging A and B will remove the relationship between B and C. Only the child relationship between A and C remains.

It is possible to end up with cycles in relationships after a merge. For example, assume A is the parent of C and C is the parent of B. Neither of the situations above applies, so merging A and B together results in A being the parent of C and C being the parent of A.

To modify the default behavior, you can make changes to the `mergeAccounts` method in the `AccountPlugin` or `AccountAPI`.

## Moving or Rewriting Policies Between Accounts

In PolicyCenter 7.0, you can now move or rewrite policies from a source account to a target account.

Moving a policy between accounts moves all policy terms, jobs, and everything else associated with the policy. PolicyCenter removes the policy from the source account.

Rewriting a policy to a new account moves the policy going forward to a target account, but the previous policy terms stay with the source account.

#### See also

- “Rewrite New Account Job” on page 100
- “Overview of Moving and Rewriting Policies Between Accounts” on page 322 in the *Application Guide*
- “Configuring Moving Policies Between Accounts” on page 335 in the *Application Guide*

## Merging Accounts

In PolicyCenter, you can now merge an account, the source, into another account, the target. The policies, work orders, notes, activities, and other data from the source account are moved to the target account. When you merge two accounts, only the target account remains. In the database, PolicyCenter marks the source account as frozen.

An underwriter may need to merge two accounts into one when two accounts represent the same person or company. This may occur due to an error such as a misspelled name, or when bringing in accounts from a legacy system.

**See also**

- “Overview of Merging Accounts” on page 323 in the *Application Guide*
- “Merging Accounts” on page 331 in the *Application Guide*

## Copying Data Between Policies

In PolicyCenter 7.0, you can now copy data from an existing policy or work order to an open work order. In the default configuration, this functionality is available in the personal auto line of business. By configuring PolicyCenter, you can add this functionality to other lines of business.

In personal auto, you may want to copy data between policies for one of the following reasons:

- An agent added a car to the wrong personal auto policy. The agent can copy the car over to the correct policy without needing to reenter the information.
- A family has a personal auto policy that covers several vehicles. A daughter moves out of the house, and her parents give her one of the vehicles. The daughter gets her own account and personal auto policy. The agent copies the vehicle from the parents’ policy to the daughter’s policy. The agent does not need to reenter the vehicle information in the daughter’s policy.
- The daughter buy a new vehicle and returns the old vehicle to her parents. The agent reinstates the old vehicle by copying the vehicle from an earlier version of the parents’ or daughter’s policy onto a policy change for the parents’ policy.

**See also**

- “Overview of Copying Data Between Policies” on page 301 in the *Application Guide*

## Splitting and Spinning Policies

In PolicyCenter 7.0, you can now *split* off two policies from an existing policy. In addition, you can *spin* off a single policy from an existing policy.

In the default configuration, this functionality is available in the personal auto line of business. By configuring PolicyCenter, you can add this functionality to other lines of business.

In personal auto, you might want to split or spin a policy for one of the following reasons:

- **Split** – A couple gets divorced. Both spouses wish to remain with the carrier. The carrier creates two new accounts, and splits the coverables on the existing policy into coverables on policies in the new accounts. The split creates two submission jobs.
- **Spin** – A son moves out of the house, and takes a car that is on his parents’ policy. The carrier creates a new account for the son, and moves the car from the parents’ policy onto a new policy on the son’s account. Spin creates a single submission.

**See also**

- “Overview of Split and Spin-off Policies” on page 303 in the *Application Guide*

## Contact Tab

The new **Contact** tab in PolicyCenter gives users a view of contacts across the Guidewire suite, including policy, billing and claims information.

On the **Contact** tab, you can search for existing contacts, add new contacts, and create a new account from a contact.

**See also**

- “Contact Tab” on page 46 in the *Application Guide*

## Linking Addresses on Contacts

In PolicyCenter 7.0, you can link a group of addresses on contacts. Two or more contact often have the same address. For example, a personal auto policy has four drivers: the husband and wife and their two teenage children. Each driver on the policy has separate contact information with an address. Because all drivers live at the same address, the address information is the same. In PolicyCenter, to avoid retyping the address, you can link a group of addresses that are the same.

**See also**

- “Linking an Address Between Multiple Contacts” on page 372 in the *Application Guide*

## Updating Contacts in Effective Time

Contact data is synchronized between accounts and policies. PolicyCenter treats this information as follows:

- **Account** – Contact data is not revised. Because accounts are not revised, all data, including the contact data, is the current data.
- **Policy** – All contact information is effective dated. Revised contact fields are accurate as of a particular point in time.

In PolicyCenter 7.0, there were changes made to how PolicyCenter handles revised contact data in a back-dated or future-dated policy change.

Updates to contact information can be back-dated or future-dated depending on the effective date of a work order. In this PolicyCenter release, the contact’s last update time is tracked so that the policy correctly represents the contact data.

For example:

1. Jane Smith got married on 10/30/2010. Her name and marriage status changed on that date, and a policy change effective that date reflects that change. PolicyCenter rates the policy again because of the marriage status change.
2. Then Jane requests another policy change effective 7/1/2010. This is an out-of-sequence policy change. You do not want Jane’s name and marriage status on the policy as of 7/1/2010 updated to the values as of 10/30/2009.

In previous versions of PolicyCenter, because the contact information was synchronized from the account when the out-of-sequence policy change started, Jane’s name and marriage status were changed as of 7/1/2010. In this version of PolicyCenter, we now know Jane’s contact information was updated effective 10/30/2010, so PolicyCenter does not apply those contact changes on a work order dated 7/1/2010. This includes other similar types of changes to address back-dated and future-dated changes to contact data.

**See also**

- “Revisioning Contact Information in Policies” on page 369 in the *Application Guide*

## Reinsurance

In PolicyCenter 7.0, you can define reinsurance agreements. Reinsurance is insurance risk transferred to another insurance company for all or part of an assumed liability. When a company reinsures its liability with another it cedes business. In the default configuration, PolicyCenter provides reinsurance for all lines of business.

PolicyCenter provides the following types of reinsurance agreements:

- **Treaties** – Agreements between the carrier and the reinsurer that provide reinsurance without the carrier's having to submit each and every risk to the reinsurer.
- **Facultative agreements** – Agreements placed on an individual case basis.
- **Programs** – Sets of treaties put together by the carrier.

**See also**

- “Reinsurance Management” on page 589 in the *Application Guide*

## Change to End Date Field in Underwriting Entities

In previous versions of PolicyCenter, the approval for an `UWIssue` had a *valid through* date. This date differed from other date ranges and `EffDated` entities. The previous field, `ApprovalExpirationDate`, on both `UWIssue` and `UWIssueHistory`, has been replaced by the field `ApprovalInvalidFrom`. PCF files were modified to reflect this change.

You might need to make the following changes:

- Update changed or added code that uses `ApprovalExpirationDate` to use `ApprovalInvalidFrom`. For example, see the definition of the property set `Duration` in `UWIssueApprovalEnhancement.gsx`.
- Merge the label changes in the PCF files, and make similar changes to custom PCF files.
- Update integration, data collection, reporting, and similar code that uses this field.

## Administering PolicyCenter

PolicyCenter 7.0 includes the following new and changed features in PolicyCenter administration:

- “Administering Policy Holds” on page 105
- “Changes to Administering Policy Forms” on page 105
- “Reassigning Job Roles” on page 106

### Administering Policy Holds

In PolicyCenter 7.0, you can administer policy holds. Policy holds allow a carrier to prevent users from creating new policies or changing existing policies in a specific area for a period of time. Some examples for using policy holds are:

- **Natural disaster** – The carrier can put a hold on issuing or changing policies in an area affected by a natural disaster.
- **Business changes** – The carrier can put a hold on issuing or changing policies for a period of time when the insurance rates are not finalized or when coverage forms are changing.

**See also**

- “Policy Holds Administration” on page 669 in the *Application Guide*

### Changes to Administering Policy Forms

In PolicyCenter 7.0, configuration of policy forms has moved from Guidewire Studio to the **Policy Forms** screen on the **Administration** tab of PolicyCenter. The advantages of this change are:

- Business users can more easily configure forms.
- Permissions and security control which users can modify forms configuration.
- Form changes take effect immediately, and do not require rebuilding or restarting the PolicyCenter server.

The following changes were made to simplify configuring forms:

- **Line and product-based forms** – Forms now use a single entity type to handle both line and product-based forms. Form patterns can be associated with particular products, policy lines, or both. Previously, the product model had separate interfaces for configuring line-based and product-based forms in Studio.  
In PolicyCenter 7.0, you define product-based forms on the **Products** tab. For more information, see “Products Tab for Form Patterns” on page 691 in the *Application Guide*.
- **Processing type** – Previously, the product model form patterns had a **ProcessingType** field, which the form inference engine used to determine how to handle a form. This configuration has been simplified, and is now configured by using Boolean fields and an array of applicable jobs. For more information see, “Policy Change Tab for Form Patterns” on page 694 in the *Application Guide*.
- **Forms inference** – In the previous version, you specified form pattern inference classes in a plain text field in Studio. In this field, you entered the fully-qualified name of a class that extended `gw.forms.FormData`. This version of PolicyCenter utilizes generic inference classes that are not specific to any particular line, product, or form. The generic inference classes handle common inference scenarios such as when to include a form based upon the presence of one of the following on the policy:
  - Coverage
  - Coverage term value
  - Typelist selection

- **New form pattern entity** – In this version, there is a new **FormPattern** entity that is functionally equivalent to the previous implementation in the `gw.api.productmodel.FormPattern` Gosu class. There is a direct correspondence between the Gosu class and fields in the entity.

The **FormPattern** entity replaces the **InferenceClass** field with a **GenericInferenceClass** field, which only handles generic inference. In the **Inference** tab for the form pattern, you can select a generic inference class from a list of options. For backwards compatibility and to accommodate use cases not covered by the generic inference classes, you can specify your own inference classes. You specify these inference classes in the new `custom_form_inference` system table. For form information, see “Configuring Custom Form Inference” on page 697 in the *Application Guide*.

#### See also

- “Policy Form Pattern Administration” on page 687 in the *Application Guide*

## Reassigning Job Roles

Users, primarily supervisors, can select activities across policies and accounts and reassign them all to another user on the **Team** tab. In this version of PolicyCenter, you can now do a similar reassignment of job roles on submissions, renewals, and work orders. To reassign job roles, navigate to the **Team** tab. Click **Submissions**, **Renewals**, or **Other Work Orders**. Select jobs and click **Assign**.

## Product Integrations

PolicyCenter 7.0 includes the following changes to product integrations:

- “Changes to the Billing System Integration” on page 106

### Changes to the Billing System Integration

In PolicyCenter 7.0, the integration with a billing system now provides subaccounts for billing, invoice streams, and alternate billing accounts.

#### See also

- “Billing System Integration” on page 705 in the *Application Guide*

## Subaccounts

An account can have multiple subaccounts for billing policies on the account. For example, a company has one subaccount for paying workers' compensation policies, and another subaccount for paying business auto premiums. An accountant at the company can easily see the billing information for each type of policy.

### See also

- “Subaccounts for Billing” on page 707 in the *Application Guide*

## Invoice Streams

Each account or subaccount can have multiple invoice streams. Invoice streams are retrieved from the billing system for the currently selected account. You can create a new invoice stream for an account.

### See also

- “Invoice Streams for an Account” on page 707 in the *Application Guide*

## Alternate Billing Account

In PolicyCenter, you can specify an alternate billing account for a policy. This account can be another account or a subaccount retrieved from the billing system.

### See also

- “Alternate Billing Accounts” on page 708 in the *Application Guide*



# New and Changed in Configuration in 7.0

This topic includes:

- “Changes to the Data Model” on page 109
- “Changes to Guidewire Studio” on page 111
- “Changes to Configuration Parameters” on page 113
- “Changes to Web Services” on page 113
- “Changes Related to PCF Files” on page 114
- “Changes to the PolicyCenter Product Model” on page 116
- “Changes to Configuring Lines of Business” on page 117
- “Improvements in Localization” on page 118

## Changes to the Data Model

In PolicyCenter 7.0, Guidewire has made the following changes to the data model:

- Deprecation of MotorVehicleRecord
- Modifications to Search Functionality
- Modifications to Associative Arrays

### Deprecation of MotorVehicleRecord

As of the PolicyCenter 7.0 release, Guidewire has deprecated the `MotorVehicleRecord` entity.

In this version of PolicyCenter, motor vehicle records have been implemented in the personal auto line of business. This implementation does not use the `MotorVehicleRecord` entity. The implementation uses new entities for representing motor vehicle records.

If you add motor vehicle records to another line of business, you can use these new motor vehicle record entities.

#### See also

- “Motor Vehicle Record Object Model” on page 270 in the *Application Guide*

## Modifications to Search Functionality

Guidewire has made changes to search functionality in this release of PolicyCenter 7.0:

- Added Conditional-use forceEqMatch Attribute to Criterion Element
- Implemented Configurable Linguistic Searching on SQL Server

#### Added Conditional-use forceEqMatch Attribute to Criterion Element

In PolicyCenter 7.0, Guidewire added a new `forceEqMatchType` attribute to the `<Criterion>` element in `search-config.xml`. The `forceEqMatchType` attribute specifies the name of a Boolean property on the criteria entity:

- If this attribute evaluates to `true`, the criterion uses an `eq` (equality) match.
- If this attribute evaluates to `false`, the criterion uses the `matchType` that the `Criterion` specifies to perform the match.

For example:

```
<Criterion property="StringProperty"
           forceEqMatchType="FlagProperty"
           matchType="startsWith"/>
```

This code uses a `startsWith` match for `StringProperty` unless the `FlagProperty` on the criteria entity is `true`, in which case, the match uses an `eq` match type.

For more information, see “The `<Criterion>` Subelement” on page 338 in the *Configuration Guide*.

#### Implemented Configurable Linguistic Searching on SQL Server

Guidewire now supports configurable linguistic search on SQL Server, enabling you to configure the case-sensitivity and accent-sensitivity of search results. You configure this by setting the strength of the `LinguisticSearchCollation` element on the `GWLocale` element of the default application locale, as specified in `localization.xml`.

---

**IMPORTANT** It is possible for this change to have an upgrade impact on installations with a SQL Server database. See “Setting Linguistic Search Collation” on page 248 in the *Upgrade Guide* for more information.

---

## Modifications to Associative Arrays

It is now possible to set a `hasSetter` and `hasGetter` attribute on each `<link-association>` element of an associative array. For example:

```
<link-association hasGetter="true" hasSetter="true">
  <typelist-map field="TAccountType"/>
</link-association>
```

For these attributes:

- If `hasGetter` is `true`, then you can read the property.
- If `hasSetter` is `true`, then you can update the property.

If you do not specify either of these attributes, then PolicyCenter defaults to `hasGetter="true"`.

## Changes to Guidewire Studio

In PolicyCenter 7.0, Guidewire has made the following changes to Guidewire Studio:

- Modifications to Type System Refresh
- Modifications to the Resources Pane
- Modifications to Tools Options
- Modifications to Dialogs
- Modifications to Typelists

### Modifications to Type System Refresh

If you create or modify an entity or entity extension, then Studio no longer requires a restart to recognize your changes.

- Studio immediately recognizes any entities that you add or any entity extensions (to existing entity fields) that you create.
- Studio immediately generates error messages on any existing references to removed types or entity fields.

### Modifications to the Resources Pane

Guidewire has modified the Studio Resources pane in PolicyCenter 7.0:

- Guidewire has added a **Localizations** node to the Studio Resources pane.
- Guidewire has removed the **Localizations** tab from the Typelist editor. Guidewire has moved this functionality to the **Typelist Localization** editor located under the **Localizations** node.
- Guidewire has moved the **Display Key** editor under the **Localizations** node. Use this editor to manage all display keys and their localization strings.
- Guidewire has renamed the previously existing **Web Services** node to **RPC-Encoded Web Services**. The name change indicates that you use this editor only to work with web services that use Remote Procedure Call (RPC) encoding. See “Changes to Web Services” on page 113 for details.
- Guidewire has added a way to manage WS-I compliant web services through Guidewire Studio. To access, select the **Classes** node in the Studio Resources tree, select a package, then right-click and select **New → WebService Collection**. See “Changes to Web Services” on page 113 for details.

#### See also

- “Typelist Localization” on page 118
- “Display Key Localization” on page 119

### Modifications to Tools Options

In PolicyCenter 7.0, Guidewire has added the ability to disable Studio from rescanning the local file system on Studio focus. This provides the ability to manually update and refresh the Studio Resources view at will.

To support the manual refresh functionality, Studio contains the following:

- A Studio toolbar icon to activate the file refresh manually.
- A **Tools → Options → Version Control → General Settings → Refresh files on focus** option that enables and disables automatic rescanning of the file system on Studio focus.

### Modifications to Dialogs

Using the keyboard Escape key (Esc) while a confirmation dialog box is open now closes the dialog box. This is the same behavior as clicking **Cancel**.

## Modifications to Typelists

In PolicyCenter 7.0, Guidewire has added a new `Jurisdiction` typelist as an addition to—but not a replacement of—the `State` typelist. Previously, PolicyCenter used the `State` typelist to indicate a regulatory jurisdiction. In the United States, this happens to be the same as the list of states, as each state regulates its own insurance market.

However, in other countries, the states or provinces are not regulatory jurisdictions. Instead, there is often only one regulatory jurisdiction, the country itself. In a multi-national deployment, it is possible that there are several regulatory jurisdictions that do not overlap at all with the list of States used to create addresses. Finally, it is possible that there is a small local regulatory jurisdiction at a level below the state.

To address this issue, Guidewire has created a new `Jurisdiction` typelist. In the base configuration, this typelist contains exactly the same values as the base configuration `State` typelist. However, in metadata and extensions, Guidewire now points most of the base configuration state fields to jurisdiction, but did not rename the fields to mitigate upgrade issues.

Guidewire has replaced a reference to the `State` typelist with a reference to the `Jurisdiction` typelist in the following base configuration entities:

- `BAJurisdiction.State`
- `BOPTaxCost.TaxState`
- `Coverable.CoverableState` (for example, `BusinessVehicle.CoverableState`, `BOPLocation.CoverableState`)
- `CPStateTaxCost.TaxState`
- `DBTerritory.State`
- `FormPatternLookup.Jurisdiction`
- `GLStateCost.GLState`
- `InclusionPerson.State`
- `LicensedState.State`
- `*Line.CoverableState` (for example, `BOPLine.CoverableState`)
- `*Line.ModifiableState` (for example, `BOPLine.ModifiableState`)
- `Modifier.State`
- `NotificationConfig.Jurisdiction`
- `OfficialIDValidationInfo.State`
- `PolicyOwnerOfficer.State`
- `PolicyPeriod.BaseState`
- `PolicyProductRoot.State`
- `PolicySearchCriteria.State`
- `ProducerSelection.State`
- `RefDateLookupCriteria.State`
- `StateDrivenLookup.State`
- `TaxLocation.State`
- `TaxLocationSearchCriteria.State`
- `TerritoryLookupCriteria.State`
- `UWCompany.State`
- `WCACraftSeat.State`
- `WCExcludedWorkplace.State`
- `WCJurisdiction.State`
- `WCStateMultiplier.State`
- `WCWaiverOfSubro.State`

The following types (listed previously) are delegates and thus impact multiple tables:

- `StateDrivneLookup`
- `Modifier`

The following extension field points to `Jurisdiction` rather than `State`:

- `UWCompany.State`

## Changes to Configuration Parameters

Guidewire has removed the following configuration parameters in PolicyCenter 7.0.0:

- `IgnoreSoapWebServiceReferenceNamespaceCollisions`

Guidewire has added the following configuration parameters in PolicyCenter 7.0.0:

- `ExternalProductModelDirectory`
- `GlobalCacheSizeMegabytes`
- `ProximityRadiusSearchDefaultMaxResultCount`
- `UseSafeBundleForWebServiceOperations`

Guidewire has added the following configuration parameters in PolicyCenter 7.0.0 that relate to the domain graph:

- `DomainGraphKnownLinksWithIssues`
- `DomainGraphKnownUnreachableTables`

### See also

- “Changes to Web Services” on page 113 for more details on `UseSafeBundleForWebServiceOperations`
- “Changes in Rules in PolicyCenter 7.0” on page 159 for more details on `ProximityRadiusSearchDefaultMaxResultCount`
- “Application Configuration Parameters” on page 35 in the *Configuration Guide* for a description of these configuration parameters.

## Changes to Web Services

In PolicyCenter 7.0, Guidewire provides support for the following types of web services:

- RPCE web services or RPC Encoded web services
- WS-I web services, for both server (publishing) and client (consuming) web service APIs

As a consequence, you now work with each web service type differently in Guidewire Studio:

Web service type	Studio
RPC-encoded	The Web Services node in the Studio Resource pane is now called <b>RPC-Encoded Web Services</b> .
WS-I	You now create WS-I web services using <b>Classes → package → New → WebServices Collection</b> .

**Note:** Although not currently deprecated, Guidewire discourages the use of RPC-Encoded web services.

### New Configuration Parameter to Support Backwards Compatibility

Guidewire has added a configuration parameter—`UseSafeBundleForWebServiceOperations`—that changes the behavior of bundle commits in web services published on this server. The default value is `true`.

- If set to `false`, the application ignores entity version conflicts as it commits a bundle.
- If set to `true`, the application detects (and does not ignore) entity version conflicts.

This bundle behavior affects RPC-Encoded web services only.

See the description of this configuration parameter in the PolicyCenter Configuration Guide for more details.

**See also**

- For more information on changes to web services in PolicyCenter 7.0.0, see “WS-I Web Services” on page 137.
- For information on how to convert your existing web services to WS-I web services, see “Converting Your Own Published Web Services from RPCE to WS-I” on page 140.
- For an introduction to web services, see “Web Services Introduction” on page 39 in the *Integration Guide*.
- For the list of all built-in web services and whether each one is RPCE or WS-I, see “Reference of All Built-in Web Services” on page 41 in the *Integration Guide*.
- For background about the WS-I standard and its use of the Document Literal encoding, see “Calling WS-I Web Services from Gosu” on page 79 in the *Integration Guide*.
- For an example of calling a RPCE web service from Gosu, see “Calling RPCE Web Service from Gosu: ICD-9 Example” on page 102 in the *Integration Guide*.

## Changes Related to PCF Files

In PolicyCenter 7.0, Guidewire has made the following change to the PCF editor:

- “New iterator Attribute on Buttons” on page 114
- “New disablePostOnEnter Attribute on Cell” on page 114
- “New onEnter Attribute on WizardStep” on page 115
- “New isDraftOnEnter Attribute” on page 115
- “New Link widgets Menu Command” on page 115
- “New Exposes Tab on ListViewPanel Widget” on page 115
- “Deprecated Unintended ListView ContentCell Configuration” on page 116
- “New Modal Cell and CellRef Widgets” on page 116
- “New CheckedValuesMenuItem Iterator” on page 116

### New iterator Attribute on Buttons

In PolicyCenter 7.0, Guidewire has added an `iterator` attribute as a required attribute on the following buttons:

- Add and Remove buttons
- Iterator buttons
- Tree buttons

### New disablePostOnEnter Attribute on Cell

In PolicyCenter 7.0, Guidewire has added a new `disablePostOnEnter` attribute to PCF row cells. You use this attribute to dictate whether a change to an input cell triggers a POST operation to the server. This is a Boolean value (either `true` or `false`.)

- If set to `false`, then PolicyCenter computes the Gosu expression defined in the `onChange` attribute as a user enters the PCF page.
- If set to `true`, then PolicyCenter does not compute the `onChange` condition and does not perform a POST operation.

PolicyCenter only computes the `Onchange` Gosu expression as a user enters the PCF page. If you need to change this behavior, then you need to explicitly refresh or reload the page.

## New onEnter Attribute on WizardStep

Guidewire has added the ability to specify an `onEnter` attribute on a `WizardStep` element within a Wizard PCF. PolicyCenter executes the Gosu in such an attribute anytime that a user enters that step through the PolicyCenter interface.

## New isDraftOnEnter Attribute

Guidewire has added a new PCF configuration option—`isDraftOnEnter`—to determine, upon entering a wizard, whether PolicyCenter highlights unsaved work prior to changes being made.

## New Link widgets Menu Command

In PolicyCenter 7.0, Guidewire has added a `Link widgets` menu item to the `Page Config` menu in the PCF editor. You use this functionality to connect a source widget (for example, a button) and a target widget (for example, a row iterator) in the PCF editor.

### See also

- “Linking Widgets” on page 299 in the *Configuration Guide*

## New Exposes Tab on ListViewPanel Widget

It is possible to locate a widget that operates on an iterator in a separate PCF file. This requires that you be able to reference that iterator from the other PCF file. To facilitate explicit iterator configuration in PolicyCenter 7.0, Guidewire has added an `Exposes` tab to a number of top-level widgets. The following widgets all have the ability to expose iterators, either iterators defined in its file, or an iterator in an included panel:

- `ListViewPanel`
- `ListDetailPanel`
- `PanelSet`
- `PanelRow`
- `RowSet`

You use the `Exposes` tab to add or to set values for the following:

- `ExposeIterator`
- `ExposeRowTree`

`ExposeIterator` contains the following attributes:

Attribute	Optional	Description
<code>applicable</code>	Yes	A Boolean value to indicate whether the panel actually exposes the iterator or not. Only use this if the panel is modal in the rare case that some modes do not define all the iterators.
<code>flags</code>	Yes	A comma separated list of flags that the iterator defines. Any iterator that defines one or more flags needs to be able to expose those flags. You use this attribute to set flags on this iterator.
<code>name</code>	Yes	The ID by which the widget is known outside of the PCF file. This is an optional attribute and it defaults to the widget ID if you do not set it explicitly.
<code>valueType</code>	No	The data type of the iterator.
<code>widget</code>	No	The ID of the widget. This can be an iterator or a list view panel. If it is a <code>ListViewPanel</code> , PolicyCenter references the first iterator in the list view panel.

`ExposeRowTree` contains a subset of the `ExposeIterator` attributes.

## Deprecated Unintended ListView ContentCell Configuration

In releases prior to PolicyCenter 7.0, it was permissible to embed a `DetailViewPanel` under a `ListViewPanel` `ContentCell` to achieve some level of formatting and layout for the cell content. In PolicyCenter 7.0, Guidewire deprecates and disallows the following usages:

Deprecated/Disallowed	Recommended alternative
A <code>DetailViewPanel</code> that is nested in a <code>ListViewPanel</code> <code>ContentCell</code> and that contains a <code>ListViewInput</code> (for example, a <code>ListView</code> inside a <code>ListView</code> cell).	Use a <code>PanelIterator</code> to iterate over a <code>ListViewPanel</code> instead.
A <code>DetailViewPanel</code> that is nested in a <code>ListViewPanel</code> <code>ContentCell</code> and that contains more than one editable Input.	Keep any composite <code>ContentCell</code> read-only and edit its content through a <code>Popup</code> .

## New Modal Cell and CellRef Widgets

In releases prior to PolicyCenter 7.0, it was permissible to create multiple modes for a `RowSet` (under a `ListViewPanel`) and reference the desired mode under certain condition through `RowSetRef`. However, this approach was not very efficient if only one `Cell` under the `Row` needed to be different under each mode. This configuration also resulted in duplicate PCF configuration as you needed to re-define all `Cell` widgets under each mode.

With the release of PolicyCenter 7.0, Guidewire supports modal `Cell` widgets, which allows you to reuse other `Cell` widgets in the `Row` under all the modes.

## New CheckedValuesMenuItem Iterator

With the release of PolicyCenter 7.0, Guidewire has added a `CheckedValuesMenuItem` iterator. This iterator functions in a similar manner to the `AddMenuItemIterator` iterator. Use it to perform iteration of a `CheckedValuesMenuItem` widget.

## Changes to the PolicyCenter Product Model

In PolicyCenter 7.0, Guidewire made the following changes to the product model:

- “Changes to Question Sets” on page 116
- “New Reloading Availability Data on a Running Server” on page 117
- “Quote Rounding Level Within a Product” on page 117

### Changes to Question Sets

You can configure your own question set types and attach them to an entity of your choice. For example, you can create a question set that PolicyCenter displays with accounts. There is an `AnswerContainer` delegate that indicates which of the entities can store answers.

In Studio, there is a **Blocking Action** for each question. The blocking action determines the action to take if someone provides an incorrect answer to a question. You can use the blocking action to block further progress or simply to warn the user.

#### See also

- “Types of Question Sets” on page 60 in the *Product Model Guide*
- “Defining New Questions” on page 62 in the *Product Model Guide*
- “Question Set Object Model” on page 69 in the *Product Model Guide*

## New Reloading Availability Data on a Running Server

In PolicyCenter 7.0, you can make changes to availability data in Studio and upload these changes to a running PolicyCenter server or clustered group of servers. The types of availability data you can upload are:

- Lookup tables
- Availability scripts
- Grandfather states

For example, a carrier needs to discontinue the personal auto product in Alaska one month from now. Or the carrier needs to grandfather collision coverage in Texas. You can make these changes to the product model in Studio, and upload the changes dynamically to PolicyCenter.

### See also

- “Reloading Availability Data” on page 95 in the *Product Model Guide*

## Quote Rounding Level Within a Product

In Studio, you set the **Quote Rounding Level** on the **Basics** tab of a product defined in **Product Model → Products**. If necessary, you can configure PolicyCenter to use different rounding levels within the product. For example, in the default configuration for personal auto, the quote rounding level is set to 0 for the product. However, the quote rounding level for taxes is 2 in New York and Florida. The `rateTaxes` method in `gw.lob.pa.rating.PARatingEngine` sets the quote rounding mode for taxes in those states. You can use this method as an example.

### See also

- “Defining a Product” on page 19 in the *Product Model Guide*

## Changes to Configuring Lines of Business

In PolicyCenter 7.0, Guidewire made the following changes to configuring lines of business:

- “Changes to Studio Icons for a Line of Business” on page 117
- “Configuring Copy Data for a Line of Business” on page 118

## Changes to Studio Icons for a Line of Business

There have been changes to the icons that Studio uses for a product and policy line:

- Previously, the icons were stored in the following directory:

`PolicyCenter_installation/modules/configuration/classes/com/guidewire/studio/resources/images`

In this version, the location of icons for the product is:

`PolicyCenter_installation/modules/configuration/config/resources/productmodel/products/images`

The location of icons for the policy line is:

`PolicyCenter_installation/modules/configuration/config/resources/productmodel/policylinepatterns/images`

- The tab icon is no longer necessary. Studio uses the same icon for the **Resources** panel and the tab on the product or policy line screen.

The instructions describing how to add product and policy line icons in Studio for a new line of business have been updated.

### See also

- “Adding Icons for the Product and Policy Line” on page 156 in the *Product Model Guide*

## Configuring Copy Data for a Line of Business

Using copy data, you can quickly and accurately take information and copy it from one policy to another. You can also take information and copy it from a prior work order of one policy to the current work order. Copy data provides:

- A mechanism for searching for policies and work orders and selecting the source period.
- A user interface that controls which items you can copy from that source period.
- Copier Gosu classes that copy the information from the source period data to the target period.

### See also

- “Configuring Copy Data in a Line of Business” on page 191 in the *Product Model Guide*

## Improvements in Localization

In PolicyCenter 7.0, Guidewire has made the following changes to improve and simplify localization:

- Address Localization
- Typelist Localization
- New Jurisdiction Typelist
- Display Key Localization
- Command Line Export Translation File Import Tool

### Address Localization

PolicyCenter uses addresses for the following data model entities:

- Contacts
- Account Locations
- Policy Locations
- Policy Addresses
- Producer Addresses

PolicyCenter also bases Organizations and Users on Contacts. Therefore, these objects use addresses as well.

With the release of PolicyCenter 7.0, Guidewire has consolidated address format and construction into a small number of address methods and PCF files. You can modify or alter these methods to handle address formats for different countries.

In PolicyCenter 7.0, Guidewire uses the following for address localization:

- A single `AddressInputSet` PCF that is modal by country.
- A single `Address` search input set that is also modal by country, along with a single set of modal search input sets.
- A common `addressString` method (in class `gw.address.AddressFormatter`) that creates a delimited address string. You use this method to manage read-only addresses.

See “Configuring Address Information” on page 103 in the *Globalization Guide* for details.

### Typelist Localization

With the release of PolicyCenter 7.0, Studio provides a **Typelist Localization** editor to facilitate typelist localization. The **Typelist Localization** editor presents typelists and typecodes in a tree similar to the existing **Entity Names** editor. After you select a typecode from the tree, the editor presents you with an editable table of localized names and

descriptions for that typecode, for each locale. Guidewire locates the **Typelist Localization** editor under a new **Localizations** node in the **Resources** tree.

Guidewire stores typelist typekey localizations in **typelist.properties** files, one file for each locale. This modification removes typelist localizations from the typelist definition files (files with extension **.tti**, **.ttx** and **.tix**). Instead, the typekey localizations exist in separate property files in manner similar to the **display.properties** file. For example, Guidewire stores the English (US) typelist localizations in file:

```
.../modules/configuration/config/locale/en_US/typelist.properties
```

Studio stores typelist typekey localizations using the following format for name and description respectively:

```
TypeKey.<typelist>.<typekey> = <localized name>
TypeKeyDescription.<typelist>.<typekey> = <localized description>
```

#### See also

- “Localizing Typecodes” on page 43 in the *Globalization Guide*

## New Jurisdiction Typelist

In PolicyCenter 7.0, Guidewire has added a new **Jurisdiction** typelist to facilitate localization and internationalization. See “Modifications to Typelists” on page 112 for details.

## Display Key Localization

In PolicyCenter 7.0, as PolicyCenter initializes the display key system, it scans all the **config/locale** directories of all modules for display key property files. Using these files, PolicyCenter generates a master list of display keys. PolicyCenter uses these values for type information, not for the actual localized display value.

#### Display Difference Utility Tool

Guidewire provides a display key difference tool that does the following:

- It compares each locale configured on the server against the master list of display keys.
- It generates a file containing the missing key and value.

To generate a display key difference report, run the following command from the application **bin** directory:

```
gwpc displaykey-diff
```

#### See also

- “Localizing Display Keys” on page 41 in the *Globalization Guide*

## Command Line Export Translation File Import Tool

In PolicyCenter 7.0, Guidewire provides a command line tool that you can use to manually import a translation file into Studio. To use this tool, do the following:

1. Navigate to your installation **bin** directory, for example:

```
PolicyCenter/bin
```

2. Run the following command:

```
gwpc import-l10ns -Dimport.file="translation_file" -Dimport.locale=destination_locale
```

The command takes the following parameters:

Dimport.file	Specifies the file that contains the translations. It must be in the same format as an export file from Studio. By default, Studio places the export file in the root of the installation directory: <ul style="list-style-type: none"><li>• If you leave the import translation file in the same location, then you need enter only the name of the file to import.</li><li>• If you move the translation file to a different location, then you enter an absolute path or a relative path to the file from the root of the installation directory.</li></ul>
Dimport.locale	Specifies the destination locale for the translations. The locale must match a PolicyCenter locale that you have defined through Studio, for example: fr_FR or ja_JP.

#### See also

- “Exporting and Importing String Resources” on page 39 in the *Globalization Guide*

# New and Changed in Gosu in 7.0

## New in Gosu in 7.0

This topic describes new features in Gosu for PolicyCenter 7.0. These include:

- “Gosu Support for Intervals” on page 121
- “Gosu Support for Numeric Literals” on page 121
- “New Null-safe Operators” on page 122
- “New Named Function Arguments and Argument Defaults” on page 122

### Gosu Support for Intervals

An interval is a sequence of values of the same type between a given pair of endpoint values. Gosu provides native support for intervals. For instance, the set of integers beginning with 0 and ending with 5 is an integer interval containing the values 0, 1, 2, 3, 4, 5. The Gosu syntax for this is `0..5`. Intervals are particularly useful for concise easy-to-understand `for` loops. Intervals could be a variety of types including numbers, dates, dimensions, and names. You can add custom interval types. In other programming languages, intervals are sometimes called *ranges*. For details, see “Intervals” on page 111 in the *Gosu Reference Guide*.

There is no upgrade impact for this change. You can optionally use intervals for new code if you wish.

### Gosu Support for Numeric Literals

Gosu now natively supports numeric literals of the most common numeric types, as well as a binary and hex syntax. Gosu uses the special syntax to infer the type of the object.

For example:

- Gosu infers that the following variable has type `BigInteger` because the right side of the assignment uses a numeric literal “`1bi`”. That numeric literal means “1, as a big integer”

```
var aBigInt = 1bi
```

- Gosu infers that the following variables have type `Float` because the right side of the assignment uses a numeric literal with an "f" after the number.

```
var aFloat = 1f
var anotherFloat = 1.0f
```

For details, see “Numeric Literals” on page 55 in the *Gosu Reference Guide*.

## New Null-safe Operators

Gosu has new null-safe operators. In general, these operators check whether an operator is `null`. They return `null` if the value to the left of the operator is `null`, or in the case of the null-safe default operator return some other arbitrary value.

The operators are:

- The null-safe property and method operator (`?.`). See “Null-safe Property Access” on page 83 in the *Gosu Reference Guide*.
- The null-safe default operator (`?:`). See “Null-safe Default Operator” on page 85 in the *Gosu Reference Guide*.
- The null-safe index operator (`?[]`). See “Null-safe Indexing for Arrays, Lists, and Maps” on page 85 in the *Gosu Reference Guide*.
- The null-safe math operators (`?+, ?-, ?*, ?/, and ?%`). See “Null-safe Math Operators” on page 86 in the *Gosu Reference Guide*.

## New Named Function Arguments and Argument Defaults

In code that calls functions, you can specify argument names explicitly rather than relying on matching the declaration order of the arguments. This helps make your code more readable. For example, typical method calls might look like the following:

```
someMethod(true, false) // what do those values represent? difficult to tell visually
```

Instead of passing simply a series of one or more comma-separated arguments, pass a colon, then the argument name, then the equals sign, then the value.

For example:

```
someMethod(:redisplay=true, :sendUpdate=false) // easy to read code!
```

Additionally, this feature lets you provide default argument values in function declarations. The function caller can omit that argument. If the function caller passes the argument, the passed-in value overrides any declared default value. To declare a default, follow the argument name with an equals sign and then the value.

For details, see “Gosu Functions” on page 96 in the *Gosu Reference Guide*.

## Changed in Gosu in 7.0

This topic describes changes in Gosu for PolicyCenter 7.0. These include:

- “Gosu Compiles to Java Virtual Machine Bytecode” on page 123
- “Gosu Syntax Changes (Compile Time Issues)” on page 123
- “Gosu Behavior Changes (Run Time Issues)” on page 128
- “New XML APIs and Improved XSD Support Using ‘XmlElement’” on page 132
- “Guidewire XML Modeler (GX Modeler) Upgraded to Use New XML API” on page 133

## Gosu Compiles to Java Virtual Machine Bytecode

Gosu now compiles directly to Java Virtual Machine (JVM) bytecode. For you, the most important thing about this improvement is that Gosu run time performance significantly improved. This affects all Gosu code, including Rules, templates, internal Gosu code, and your own custom Gosu code.

For changes to the syntax or the run time behavior due to this change, see the following sections:

- “Gosu Syntax Changes (Compile Time Issues)” on page 123
- “Gosu Behavior Changes (Run Time Issues)” on page 128

## Gosu Syntax Changes (Compile Time Issues)

Because of the change for Gosu to compile directly to Java bytecode, the Gosu compiler and the Gosu run time behavior changed in some ways. The following sections list side effects of this change that change Gosu syntax slightly in ways that some code syntax is no longer supported.

### Mismatched property Getter/Setter Types

You can no longer define properties on a Gosu class in which the getter and the setter return different types. For example, do not define both of the following:

- a property getter (`property get` function) that takes an `Integer`
- a property setter (`property set` function) that takes a `String`.

A property can only have one type in Gosu. A property is an abstraction for an underlying field.

Gosu detects this issue at compile time. Gosu flags it as a compile error. Refactor the code to make the types match. To minimize compatibility issues, change the one that was declared second, since previous Gosu releases were ignoring the type of the second mismatched getter or setter.

---

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Change properties that use different types for getters and setters to use the same types.

---

### Overriding a Generic Function with a non-Generic function

A generic function is a function with a generic declaration in the function like `function foo<T>(arg : T) : T`. For more information about generics, see “Gosu Generics” on page 239 in the *Gosu Reference Guide*.

If you override a generic function, you must override it with a function with exactly the same parameterization. This was previously not a requirement.

Gosu detects this issue at compile time. Gosu flags it as a compile error. To fix it, change the child declaration to exactly match the overridden method declaration.

---

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Change overridden generic functions that have a different parameterization to match the parameterization of the overridden method declaration.

---

### Covariantly Overriding the Getter Half of a Writable Property

For properties defined as writable on a superclass, you cannot covariantly override getters (`property get` functions) on a subclass. A property must only have one type.

For example, suppose the parent class has a writable property of type `Object` but the child overrides the getter to return the more specific type `String`. You cannot override the setter on the child to take a `String`. Even if you could, you could cause run time errors in other code. For example, other code that had an instance of the object typed as the superclass and tried to set the property to an `Integer`. That could generate an error on the setter call or an error on subsequent getter calls.

Gosu detects this issue at compile time. Gosu flags it as a compile error.

There are two possible fixes:

- You can make the property read-only and define a separate, explicit setter method. You can then override the setter on the subclass to verify the argument type so things at least fail fast with a reasonable error message.
- You can generify the class and have the property be of that type. This approach ensures that the getters and setters continue to have the same type. For example:
  - a. Generify the superclass as `<T>`.
  - b. Define the property as of type `T`.
  - c. In the subclass, declare `MySubclass extends MyParentClass<String>` to ensure the property has the type `String`.

---

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Change overrides of getters that use different types than the superclass to use the same types.

---

## Overriding an Enhancement Method

Enhancement methods cannot be overridden on a subclass of the enhanced class.

Gosu statically dispatches enhancement methods at compile time. In Java bytecode, enhancements link to actual static methods. As a result, subclasses cannot override them, nor can enhancements override other methods.

Gosu detects this issue at compile time. Gosu flags it as a compile error if you marked the method as an override. Otherwise, Gosu flags this code as a compile warning rather than an error.

To fix this code, put the method on the superclass directly, instead of using an enhancement. If that is impossible, put `instanceof` calls into the enhancement on the superclass to simulate dynamic dispatch.

---

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Change enhancement methods that override methods defined on superclasses of the enhancements.

---

## Method Overloading Involving Non-Java-backed Types as the Arguments

You cannot overload a method signature in which the only difference is in non-Java-backed types from a type loader. Although this does not necessarily affect all types from a type loader, it affects many types from a type loader. For example, SOAP APIs from an external system or PCF types are non-Java-backed.

Normally, for Java bytecode for the JVM, at run time the JVM looks up the class on disk by its name.

For non-Java-backed types, there may not be a one-to-one correspondence between custom types and Java classes on disk. So, the Gosu compiler finds the common denominator between the custom type and the Java representation, typically `Object`. In some sense, the generated bytecode flattened down to that class, typically `Object`.

For example, means that two methods named `goToPage()` on two different PCF types override compile to `goToPage(PCFElement)`, resulting in duplicate method definitions on the `PCFElement` class.

Gosu detects this issue at compile time. Gosu flags it as a compile error.

There are two options to fixing this issue:

- Create a single method that takes a generic parameter (or even just `Object`) and then dispatch within it by using `typeis` on the argument. Note that `typeis` on a `null` argument always returns `false`, so you must handle `null` arguments specially.
- Put each overloaded variant in a separate enhancement class. Because each enhancement class compiles to its own class, there are no duplicated methods. A downside of this approach is that you cannot override any methods.

---

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Change overloaded methods that vary by non-Java-backed types as arguments.

---

## Variables With No Type Cannot Initialize to Null

Generally speaking, the type inference feature in Gosu allows you to declare a new variable and omit the type declaration. Gosu uses the type of the object on the right hand side of the equals sign to determine the type of the variable.

However, Gosu now forbids declaring a variable to have the value `null` without declaring the type. In other words, do not write code such as:

```
var x = null
```

A variable has to have a type on the stack. Implicitly or explicitly using the type `void` (technically the type of `null`) does not fix this problem for the Java Virtual Machine. This was previously a compiler warning, but it really indicates a programming error. Gosu now flags this as a compile error.

To fix this, explicitly assign the appropriate type to the variable and do not use the type `void`.

For example, you could change the previous example to:

```
var x : String = null
```

---

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Add type declarations to variables that you initialize to no value.

---

## Properties Must Not Conflict with Explicit Getter or Setter Methods

Gosu now strictly prevents Gosu properties from creating implicit methods that conflict with methods on a type. Java has no concept of what Gosu calls *properties*. Gosu compiler generates method signatures in Java bytecode from the Gosu properties. This means that Gosu property `get` and `set` functions and regular methods now can collide.

Gosu properties generate implicit get/set methods, depending on the readability and writability of the property. For example, a `Name` property implicitly generates a `getName` and `setName` method, assuming the property is both readable and writable. A method with the explicit declared name `getName` or `setName` would conflict with the generated methods from the parameter if the argument numbers and argument types also match.

Gosu detects this issue at compile time. Gosu flags it as a compile error.

To fix this issue, remove the extra functions (methods), which in typical code are redundant and accidental.

---

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Remove explicit methods that collide with implicit getter and setter methods for Gosu properties.

---

### Do Not Use the Return Value of a Void Function In an Expression

Functions that return nothing can no longer participate in other expressions. In other words, functions that technically have the return type `void` literally return no value. You cannot call a `void` function and use the result in some expression. The call to the method must be a stand alone Gosu statement. When compiled to Java, the value `void` is not a valid value on the stack, so the expression has an undefined result.

For example, if the function `syncObject` returns no value, the following code is invalid:

```
if (syncObject() > 0) {
    print("hello?")
}
```

Gosu detects this issue at compile time. Gosu flags it as a compile error. Typically this indicates a programming error. To fix this issue, change the method call to a stand alone statement.

---

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Remove void functions from expressions and put them in stand alone statements.

---

### Checked Exceptions Changes in Gosu

Exceptions are wrapped differently in PolicyCenter 7.0.

Previously, you could write exception code where the exception's type is undeclared:

```
try {
    ...
}
catch (e) {
    ...
}
```

That caught exceptions wrapped as an evaluation exception. The actual exception object was in that object's `Cause` property. The `Cause` property was `null` if the exception has no checked exception associated with it.

Starting in PolicyCenter 7.0, that undeclared type syntax is deprecated.

---

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** In try/catch statements, rewrite undeclared exception types to catch type `Exception` or a more specific subtype. Review code in the catch block to ensure it does not unwrap the exception.

---

Change your code to use the syntax

```
try {
    ...
}
catch (e : Exception) {
    ...
}
```

You can alternatively specify a more specific subtype of `Exception`, such as `IOException`.

After making this change in a catch block, carefully review the code in the catch block in case it was trying to do any unwrapping of the exception. For example, by accessing the `Cause` property, which is known as unwrapping the exception. Unwrapping the exception is no longer necessary after you change the catch block to explicitly declare `Exception` or a subtype of `Exception`.

For more information, including why to catch `Exception` and why not to catch the supertype `Throwable`, see “Catching Exceptions in Gosu” on page 103 in the *Gosu Reference Guide*.

For a related change that affects Java code that interacts with Gosu, see “Catching Exceptions in Java When Gosu throws Exceptions” on page 131.

### Higher Precedence for ‘as’ and ‘typeas’ Operators

The precedence of the coercion/conversion operators `as` and `typeas` has increased to match the more conventional priority in other programming languages.

Consider the Gosu expression `1 + 1 as String` with no explicit parentheses.

- In PolicyCenter 4.0, Gosu evaluated it as `((1 + 1) as String)`, which evaluates to 2
- In PolicyCenter 7.0, Gosu evaluates it as `(1 + (1 as String))`, which evaluates to 11

Gosu detects this issue at compile time. Gosu flags it as a compile error.

To fix this error, explicitly parenthesize the expression in question. A common situation is adding parentheses to any cast to `String` that occurs within an additive `+ expression` that is not already a `String`.

### Case-sensitive Language Keywords

Gosu is now more strict with the proper capitalization of Gosu language keywords.

Gosu detects this issue at compile time. Gosu flags it as a compile error.

To fix this, simply fix the capitalization to match the keyword capitalization as indicated in the compile error.

### Relative Imports Discouraged, and Now Sometimes Require Fully-Qualified Type Names

In general, to import a type into Gosu, use the `uses` keyword at the top of the file. In some cases, you can omit the fully-qualified name of a type even without the `uses` keyword and the Gosu type system can find the type unambiguously.

One variant of this is called a relative path, which means to use a path relative to the current package of the code that is running. If you use relative paths to refer to a type, in some cases this does not work in Gosu, which causes a run time error.

**IMPORTANT** It is best to avoid all relative-path imports of types in any new Gosu code.

Gosu detects this issue at compile time. Gosu flags it as a compile error.

To fix this issue, simply fully qualify the type you want to import. For example, in a Gosu class `com.mycompany.MyClass`, suppose you referenced the type `com.mycompany.subpackage.MyUtil` by typing `subpackage.MyUtil`.

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Change imports of types that use relative paths to use fully qualified paths.

## Gosu Behavior Changes (Run Time Issues)

Due to the changing Gosu to compile directly to Java bytecode, there were changes in the Gosu compiler and the Gosu run time behavior. The following sections list changes in the run time behavior.

**IMPORTANT** The following topic describes changes that affect run time behavior that Guidewire Studio does not flag at compile time as compile errors. Review your code for these issues.

### Accessing Package-local Java Classes from Gosu classes in the Same Package

You cannot create a package-local Java class.

If you do the following tasks in order, Gosu disallows it:

1. Create a Java class with *no explicit access modifier*. In other words, do not add public or private access modifier. If you omit an access modifier, the default is called *package protected* (*package-protected*).
2. Create a Gosu class in the same package.
3. From the Gosu class, access the Java class by name and access properties or call methods on it.

In Java, a matching package implies a matching package name and a matching class loader. Since the Gosu classes are in a different class loader from the Java classes, the Java Virtual Machine (JVM) considers them different packages. The JVM throws the `IllegalAccessExceptions` exceptions.

**WARNING** Gosu does not detect this upgrade issue at compile time. Avoid relying on run time errors to find this problem. Review your project for this issue.

To fix this issue, mark the Java class with the `public` modifier.

---

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Review code for package-local Java classes and how behavior might change.

---

### Standard Arithmetic Operators Are No Longer Null-safe

Arithmetic expressions of reference types (non-primitive types) are no longer null-safe. In PolicyCenter 4.0, if either side of an expression was `null`, the entire expression evaluated to `null`. This now causes an null pointer exception.

Although the old behavior is sometimes the intended behavior, it often is a logical error. This change encourages coders to make the null-safe or non-null-safe choice explicitly.

Starting in PolicyCenter 7.0, Gosu has more null-safe operators, for example the null-safe addition operator: `?+`. The null-safe arithmetic operators have the same semantics that the simple operator `+` used to have with `null` values. For more details, see “Null-safe Math Operators” on page 86 in the *Gosu Reference Guide*.

At run time, Gosu throws a `null` pointer exception if the object on either side of the operator is `null`. The exception text for this particular issue contains details of how to make an informed decision about how to handle `null` values.

**WARNING** Gosu does not detect this upgrade issue at compile time. Avoid relying on run time errors to find this problem. Review your project for this issue.

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Review code for changes to null safety of arithmetic operators.

## Array Casts

Do not use the `as` keyword to cast an array to a different array type. If you attempt it, Gosu throws an exception at run time.

**WARNING** Gosu does not detect this upgrade issue at compile time. Avoid relying on run time errors to find this problem. Review your project for this issue.

To fix this, call the `cast` method that exists on all arrays and pass the type of each element. For example, if you want to cast the array to an array of `Integer` objects, pass the type name `Integer` as an argument.

For example:

```
uses java.lang.Integer

var arrayOfInt : Number[] = {3, 4, 5}

// do not directly cast -- it does NOT work!
// var a = arrayOfInt as Integer[]

// cast from Number[] to Integer[]
var a = arrayOfInt.cast(Integer)
```

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Review code for changes to array casts.

## Primitive Property Short-Circuiting

In PolicyCenter 4.0, property access was null-safe even with primitive properties. For example, if the type of a property is `int`, code similar to `Obj.IntProp` gets the integer property. However, if `Obj` has the value `null`, the expression returns `null` rather than throw a `null` pointer exception. This created the unusual behavior of letting primitive types contain the `null` value at least temporarily within an expression. This is unusual, since `null` properly can be the value only for a reference type (a non-primitive type). In most cases, this worked, especially for `boolean` types, since `null` coerced to `false`, and typically this returned the expected answer.

In PolicyCenter 7.0, Gosu is more strict regarding null-safety of operators. The standard period operator for property access and method invocation is not null-safe. If the object to the left of the period has the value `null` and on the right side there is a primitive property, the expression throws a `null` pointer exception.

There are several exceptions:

- The array property called `length`, which short-circuits to the value 0 if a `null` value is on the left side.
- All boolean properties, which always short-circuit to `false` if a `null` value is on the left side

- The Count enhancement property some collection types.

**WARNING** Gosu does not detect this upgrade issue at compile time. Avoid relying on run time errors to find this problem. Review your project for this issue.

To fix this issue, use the `?.` operator to force null-short-circuiting a property access for primitive properties. If you use the `?.` operator, Gosu uses the default value (0.0, 0, or `false` as appropriate) for a primitive expression instead of throwing a null pointer exception.

**IMPORTANT** For more information about null-safe property operators, see “Handling Null Values In Expressions” on page 83 in the *Gosu Reference Guide*.

---

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Review code for changes to primitive property short-circuiting

---

## Class Loading and Initialization Ordering

Due to the changes to make Gosu compile down to Java bytecode, the order which classes load and initialize by the JVM might change. For example, certain classes might load earlier than they did before.

You must never rely on class load ordering for any functionality. For example, do not do something like:

- Have a static initializer for one class called A that puts something in a static hash map on another class B.
- Have another class C use that map in another static initializer on class C.

If suddenly C starts loading before A, there will be run time errors.

**WARNING** Never rely on class loading ordering for any functionality.

---

If you relied on class load ordering in previous versions, you might notice errors that are difficult or impossible to debug when the order changes, like in the PolicyCenter 7.0.

If you see any null pointer exceptions in the `c1n1t` method on a class, that refers to the internal method that statically initializes a class. Gosu calls this when any code first references the type. That error implies this type of loading and initialization error. The kinds of errors that could occur because of relying on class loading order could be much more subtle and impossible to debug. Be extremely careful not to rely on class loading order.

**WARNING** Gosu does not detect this upgrade issue at compile time. Avoid relying on run time errors to find this problem. Review your project for this issue.

---

If you have this type of error, refactor the code so that you do not rely on class load order.

---

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Review code for changes to class loading.

---

## Arrays of Non-Java-backed Types with ‘typeof’ or ‘TypeSystem.getFromObject(o)’

If an array contains elements of a native Java or Gosu type, including Java classes and Gosu classes, the `typeof` operator and the `TypeSystem.getFromObject(Object)` method work as expected.

However, the behavior is more complex for non-Java-backed types.

**Note:** For more information about non-Java-backed types, see “Method Overloading Involving Non-Java-backed Types as the Arguments” on page 124.

This issue affects:

- arrays of Guidewire entity types
- arrays of dynamically-generated SOAP (web service) types
- arrays of dynamically-generated XSD types
- arrays of PCF types

In Java bytecode, these types do not directly correspond to the Gosu equivalent. At run time, the type information is unavailable.

For example, the following code is valid because the types are standard types (built-in types and Gosu classes, in this example):

```
uses mycompany.MyClass
var t1 = typeof(new Integer[])
var t2 = typeof(new MyClass[])
```

However, the following code does not work as expected if the array type uses an XSD type:

```
var t3 = typeof(new xsd.myxsd.MyElementType[])
```

In this example, the objects in the package hierarchy return the less useful run time type `XmLElement[]`. This result is semantically incorrect but corresponds to the run time implementation type of the object as seen by the Java Virtual Machine.

---

### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Review code for changes to arrays of non-Java-backed types with `typeof` or `TypeSystem.getFromObject(o)`.

---

## Catching Exceptions in Java When Gosu throws Exceptions

Exceptions are wrapped differently in PolicyCenter 7.0. Exceptions thrown from a `throw` statement in Gosu are no longer wrapped in an `EvaluationException` exception. Any Java code that calls into Gosu and expects to catch a wrapped `EvaluationException` must instead change to catch the actual exception type.

For example:

```
function someGosu() {
    throw new IllegalArgumentException("test")
}
```

In previous releases, in Java code you could catch exceptions with the following code

```
try {
    // call code via ReflectUtil, or invoke a method on a Java interface that's implemented in Gosu
    callSomeGosu()
} catch (EvaluationException e) {
    IllegalArgumentException iae = (IllegalArgumentException) e.getCause();
    // Handle illegal argument exception
}
```

In PolicyCenter 7.0, that Java catch statement does not trigger. A similar issue happens if the catch statement caught the exception types `Exception` or `RuntimeException`. In that case, the catch statement would trigger, but Gosu would invoke it directly with the `IllegalArgumentException` rather than wrapping it.

This means that it is now better to write Java exception-handling code like:

```
try {
    // call code via ReflectUtil, or invoke a method on a Java interface that's implemented in Gosu
    callSomeGosu()
} catch (IllegalArgumentException e) {
    // Handle the illegal argument exception variable "e"
```

```
}
```

**WARNING** Gosu does not detect this upgrade issue at compile time. Avoid relying on run time errors to find this problem. Review your project for this issue.

If you have `try/catch` statements in Java code that uses the Guidewire entity libraries, audit your `catch` statements. Ensure your `catch` statements do not assume that exceptions coming from Gosu are wrapped in an `EvaluationException` exception.

For a related change that affects your Gosu code also, see “Checked Exceptions Changes in Gosu” on page 126.

---

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Review code for changes to Java code that catches exceptions from Gosu.

---

## Private Variables on Superclasses with the Same Name as a Variable on the Subclass

If a parent class has a private variable, a subclass cannot access it.

In PolicyCenter 4.0, if a subclass declared a private variable with the same name as the parent class private variable, Gosu was inconsistent about which one to get or set.

In PolicyCenter 7.0, the subclass can create its own private variable or non-private variable with the same name as a private variable on the parent class. Gosu treats as separate the superclass and subclass versions of fields with matching names. The subclass cannot access the private variable of the superclass, which is the correct behavior for `private` keyword.

No compile-time semantics have changed, so this problem cannot be detected at compile time. The private variable is a legal symbol in both classes regardless of the runtime behavior. It is solely a run time change such that those fields are now properly separate.

**WARNING** Gosu does not detect this upgrade issue at compile time. Avoid relying on run time errors to find this problem. Review your project for this issue.

---

---

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Review code for changes to private variables on superclasses with the same name as variables on subclasses.

---

## New XML APIs and Improved XSD Support Using ‘XmlElement’

There is a new set of XML APIs based on the root class `XmlElement` instead of `XMLNode`. The main class that represents an XML element is the class `XmlElement`. The most important feature in the new APIs is improved support for XSDs:

- The `XmlElement` APIs handle a wider set of standard-compliant XSDs.
- The `XmlElement` APIs handle namespaces in a more standards-compliant way.
- The `XmlElement` APIs use improved algorithms for adding properties to Gosu objects based on properties and attributes in an XSD.

For more information about the new APIs, see:

- “Manipulating XML Overview” on page 268 in the *Gosu Reference Guide*
- “Introduction to the XML Element in Gosu” on page 271 in the *Gosu Reference Guide*.

You can still use the older APIs that use `XMLNode`, particularly for code you have already written. This documentation refers to as the *legacy XML APIs*. The XML topic still contains documentation for the legacy XML APIs.

For detailed differences between the new XML APIs and the legacy XML APIs, see “Differences Between Legacy XML APIs and Current XML APIs” on page 269 in the *Gosu Reference Guide*.

## Backward Compatibility

You can continue to use the legacy `XMLNode` APIs for specific code you already wrote. They are not deprecated. However, for new code, use the newer `XmLElement` APIs.

There is an additional upgrade-specific feature. For backward compatibility only, you can optionally import an XSD into the Gosu type system using the legacy XML system’s XSD loader. To do this, modify the `compatibility-xsd.xml` file. Specifically, copy the file `PolicyCenter/modules/p1/config/registry/compatibility-xsd.xml` into the configuration module at the same path. Next, add an entry for your XSD. Set the `namespace` attribute to the Gosu package name of the schema. For example, if the schema is in the package location `example.pkg` and is called `myschema.xsd`, set the namespace to `example.pkg.myschema`.

## Guidewire XML Modeler (GX Modeler) Upgraded to Use New XML API

All APIs in the Guidewire XML Modeler (GX modeler) now use the new XML API, whose primary class is `XmLElement`. This might require changes to your code that interacts with an XML model.

### See also

- “Manipulating XML Overview” on page 268 in the *Gosu Reference Guide*
- “Introduction to the XML Element in Gosu” on page 271 in the *Gosu Reference Guide*
- “Differences Between Legacy XML APIs and Current XML APIs” on page 269 in the *Gosu Reference Guide*.

## Interceptors Removed

Before PolicyCenter 7.0, there was a feature called interceptors that allowed you to trigger custom code before or after method invocation. The mechanism was through annotations that implement the `IPreMethodInterceptor`, and `IPostMethodInterceptor` interfaces. This mechanism was removed as part of the change to byte-code-compiled Gosu in PolicyCenter 7.0.

---

### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Review code to remove any Gosu interceptors.

---



# New and Changed in Integration in 7.0

## New in Integration in 7.0

This topic describes what is new for integration with PolicyCenter 7.0. This includes:

- “Motor Vehicle Record (MVR) Plugin” on page 135
- “Reinsurance Architecture and Plugin” on page 135
- “New ClaimCenter PolicyCenter Integration” on page 135
- “Policy Hold Evaluation Plugin” on page 136
- “New Populator Plugin” on page 136
- “New Methods in ProductModelAPI” on page 136

### **Motor Vehicle Record (MVR) Plugin**

This new plugin generates a request to the motor vehicle record (MVR) provider and returns the MVR data. See “Motor Vehicle Record (MVR) Plugin” on page 228 in the *Integration Guide*.

### **Reinsurance Architecture and Plugin**

PolicyCenter 7.0 include significant support for reinsurance. As part of this new feature, there is a new plugin called the reinsurance plugin (`IReinsurancePlugin`). For more information:

- For overview information about reinsurance, related terminology, and the role of reinsurance in the PolicyCenter product, see “Reinsurance Management Concepts” on page 591 in the *Application Guide*.
- For implementation details of the reinsurance plugin, see “Reinsurance Integration” on page 453 in the *Integration Guide*.

### **New ClaimCenter PolicyCenter Integration**

If you use PolicyCenter 7.0 and ClaimCenter 7.0, a new tool in PolicyCenter helps you synchronize your PolicyCenter product model with your ClaimCenter lines of business model. For details, see “PolicyCenter Product Model Import into ClaimCenter” on page 560 in the *Integration Guide*.

## Policy Hold Evaluation Plugin

There is a new policy hold evaluation configuration plugin. For details, see “Policy Hold Job Evaluation Plugin” on page 231 in the *Integration Guide*.

## New Populator Plugin

Use the `PopulatorPlugin` to populate entities from any `String` data format. For example, import XML data and populate a `PolicyPeriod` and its subobjects. There is a default plugin implementation that does this, but you can modify it to add line-specific information. Also you can modify the XML model that this default plugin uses, for example to add or remove properties from the XML. See “Populator Plugin” on page 230 in the *Integration Guide*.

## New Methods in ProductModelAPI

There are two new methods in the `ProductModelAPI` web service:

- To return the list of available questions for a specified policy period, call the `ProductModelAPI` web service method `getAvailableQuestions`. The return type is a list of question sets, as the type `List<QuestionSet>`.
- To return the list of available questions for a specified policy period, call the `ProductModelAPI` web service method `getAvailableClausePatterns`. The return type is a list of clause patterns, as the type `List<ClausePattern>`.

## Changes in Integration in 7.0

This topic describes what has changed for integration with PolicyCenter 7.0. This includes:

- “WS-I Web Services” on page 137
- “Setting Locale in WS-I Requests for Guidewire Servers Only” on page 142
- “IAccountPlugin Web Service Changes for Transferring Policies” on page 143
- “SubmissionAPI Changes” on page 143
- “PolicyAPI Changes” on page 144
- “PolicyRenewalAPI Changes” on page 144
- “Confirming Renewals Integration Flow” on page 145
- “Guidewire InsuranceSuite Plugin Implementations are Versioned” on page 145
- “SOAP Implementation Classes and WSDL Packages Include Version” on page 146
- “Geocoding Only Some Addresses” on page 147
- “Changes to PolicyCenter Contact-related Plugins” on page 147
- “ContactAPI Web Service Changes” on page 148
- “Improvements to Matching Classes” on page 148
- “Escalation and Exception Plugins” on page 150
- “Billing Plugin Changes” on page 151
- “For Messaging, a Contact is Now a Safe-ordered Object” on page 151
- “Integrating PolicyCenter 7.0.0 with ClaimCenter 6.0.0” on page 152
- “AccountSyncable Changes” on page 152
- “Changes to Messaging System Safe Ordering” on page 153

## WS-I Web Services

The web services implementation that previously existed uses the SOAP protocol using the Remote Procedure Call (RPC) encoding. This documentation refers to this older style of web services as RPC-Encoded web services or sometimes shortened to RPCE web services.

There is a new standard called WS-I web services. PolicyCenter provides a fully WS-I standard-compliant web services layer for both server (publishing) and client (consuming) web service APIs. This is the recommended approach for all new published web services.

If you previously published RPCE web services with the `@WebService` annotation, you can continue to do so. To reduce ambiguity about RPCE web services versus WSI web services, the old annotation name for RPCE web services changed from `@WebService` to `@RPCWebService`.

---

**WARNING** RPCE web services are deprecated as of version 8.0. Immediately begin to convert existing RPCE code to WS-I web services.

---

If you want to create new WS-I web services or convert your RPCE web services to WS-I, use the new annotation `@WsiWebservice`. The syntax of the annotation has changed compared to the older RPCE web services. Most notably, the main web service publishing annotation `@WsiWebService` takes no arguments. The RPCE web services used arguments on the annotation to add additional configuration settings, such as required permissions or minimum run level to access the web service. The newer `@WsiWebService` annotation relies on additional annotation types on the published Gosu class, for example `@WsiPermissions` to set the permissions on the service.

For a upgrade-specific information, see the following topics:

- “Converting Your Own Published Web Services from RPCE to WS-I” on page 140
- “Changes for Consuming Web Services in WS-I” on page 141

For full documentation about publishing or consuming web services, be sure to start with the right documentation for your task. See “Finding Documentation for Connecting to Web Services” on page 141.

Some but not all built-in web services converted to WS-I. You connect to WS-I web services differently than you connect to RPCE web services. To learn which built-in web services are RPCE and which are WS-I, see “Reference of All Built-in Web Services” on page 41 in the *Integration Guide*.

## Publishing RPCE Web Services Compared to WS-I Web Services

In addition to the differences in the web service publishing annotation itself there are deeper differences between RPCE and WS-I web services.

**WARNING** RPCE web services are deprecated as of version 8.0. Immediately begin to convert existing RPCE code to WS-I web services.

Behavior of publishing a web service	RPCE web service behavior	WS-I web service behavior
The basic annotation on a class to publish a web service	<p>@RPCWebService</p> <p><b>IMPORTANT:</b> This is just a new name for the annotation @WebService. The old name still works and refers to RPCE web services, but it is deprecated.</p> <p>This annotation supports many arguments to configure run level, permissions, and local WSDL creation.</p>	<p>@WsiWebService</p> <p>This annotation supports one optional argument. The argument is the namespace for the web service. See "Consuming WS-I Web Service Overview" on page 79 in the <i>Integration Guide</i>. If you do not declare the namespace, Gosu uses the default namespace "<a href="http://example.com">http://example.com</a>".</p>
Does PolicyCenter automatically generate WSDL files from a running server?	Yes	Yes, although the URL is different. See "Getting WS-I WSDL from a Running Server" on page 56 in the <i>Integration Guide</i>
Does PolicyCenter automatically generate WSDL files locally if you regenerate the SOAP API files?	Yes	Yes
Does PolicyCenter automatically generate JAR files for Java SOAP client code if you regenerate the SOAP API files?	Yes	<p>No, but it is easy to generate with the Java built-in utility <code>wstool</code>. The documentation includes examples. See "Calling a PolicyCenter WS-I Web Service from Java" on page 74 in the <i>Integration Guide</i>.</p> <p><b>IMPORTANT:</b> This is an important difference between RPCE web services and WS-I web services.</p>
Can it serialize a Java object as an argument type or return type?	Yes, in limited situations, such as some Java-based types inside the <code>gw.*</code> package namespace.	No
Can it serialize a Guidewire entity as an argument type or return type?	Yes, if the entity is marked in the data model as <code>Exportable</code> .	<p>No</p> <p><b>IMPORTANT:</b> All published WS-I web services must have no arguments or return types that are Guidewire entity instances</p>
Can it serialize a XSD-based type?	No	Yes, using the <code>XmlElement</code> APIs.
Can it serialize Gosu class instances (sometimes called POGOs: Plain Old Gosu Objects)?	Yes	<p>Yes, however the class must have the following two special qualities:</p> <ul style="list-style-type: none"> <li>• The class must be declared as <code>final</code>, which means it has no subclasses.</li> <li>• The class must have the annotation <code>@WsiExportable</code>.</li> </ul> <p><b>IMPORTANT:</b> Remember to add the <code>final</code> modifier and the <code>@WsiExportable</code> annotation. If you forget, you cannot use that type as a return type or argument type for a WS-I web services.</p>

Behavior of publishing a web service	RPCE web service behavior	WS-I web service behavior
How to specify the minimum run level?	As an argument to the basic web service publishing annotation	The annotation @WsAvailability
How to specify the required permissions to use the service?	As an argument to the basic web service publishing annotation	The annotation @WsPermissions
Automatically throw SOAPException exceptions?	Yes	<p>No. Declare the actual exceptions you want thrown. In general this requirement reduces typical WSDL size.</p> <p>If you want identical behavior to RPCE web services, add <code>Throws (ServerStateException, PermissionException, BadIdentifierException)</code> onto your Gosu class. However, that set of exceptions probably is not what you want. Carefully consider what exceptions your web service encounters.</p>
<p>Bundle handling</p> <p><b>Note:</b> a <i>bundle</i> is a container for entity instances. A bundle helps PolicyCenter track what changed in a database transaction. See “Bundles and Database Transactions” on page 331 in the <i>Gosu Reference Guide</i>.</p>	RPCE web services put any entity changes into a default bundle, which you manually commit as needed. Additionally, the bundle management is nonstandard. Any changes in a web service default bundle do not trigger message events or respect concurrent data change exception rules.	<p>There is no default bundle for WS-I web services. You define your own bundle. Use the <code>runWithNewBundle</code> API to create a new bundle. See “Running Code in an Entirely New Bundle” on page 341 in the <i>Gosu Reference Guide</i>.</p>
Logging in to the server	<p>There are two ways in the RPCE system to log into a server.</p> <ul style="list-style-type: none"> <li>For Java clients that used the generated libraries, the standard approach was to use the <code>APILocator</code> utility class to manage the connection and handling connection pooling.</li> <li>Other client languages (or Java if desired) could connect to the <code>ILoginAPI</code> web service interface.</li> </ul>	<p>Do not use either the <code>APILocator</code> utility class or the <code>ILoginAPI</code> web service interface. They do not apply to WS-I web services.</p> <p>Instead, each WS-I request embeds necessary authentication and security information in each request.</p>

<b>Behavior of publishing a web service</b>	<b>RPCE web service behavior</b>	<b>WS-I web service behavior</b>
Package name of web services from the SOAP API client perspective	The namespaces are flattened so that all web services appeared in the same package.	<p>The package in which you put your web service implementation class defines the package for the web service from the SOAP API client perspective. The biggest benefit from this change is reducing the chance of namespace collisions in general.</p> <p>For Guidewire InsuranceSuite customers, this makes it much easier to support backwards compatibility with previous (or future) Guidewire application SOAP APIs. The names could match but the package name specifies the application and version number. For example, the structure of the class name might look like:</p> <p><i>MY_PACKAGE_NAME.bc300.MY_CLASS_NAME.</i></p>
Calling a local version of the web service from Gosu	<p>Gosu creates types in the package <code>soap.local.api.APINAME</code></p> <p>All the references were in that package, independent of the package of the implementation class.</p> <p>You must call the following command to rebuild the local files:</p> <p><code>PolicyCenter/bin/gwpc regen-soap-local</code></p>	<p>Gosu creates types that use the original package name to avoid namespace collisions.</p> <p>API references in the package <code>wsi.local.ORIGINAL_PACKAGE_NAME</code></p> <p>You must call the following command to rebuild the local files:</p> <p><code>PolicyCenter/bin/gwpc regen-wsi-local</code></p>

## Converting Your Own Published Web Services from RPCE to WS-I

Refer to the previous table for major differences between RPCE and WS-I web services.

To convert an RPCE web service to WS-I might be a small or large project, depending on the type of service.

For a simple example, suppose all the parameters and return values of the RPCE web service are simple Gosu objects (sometimes called POGOs, Plain Old Gosu Objects). The challenges of converting the web service is probably low. You probably will need to accommodate changes such as the differences in bundle handling.

In contrast, if your RPCE web services use Guidewire entity instances as parameters or return values, you may need major refactoring to accommodate the conversion to WS-I. You must redesign the code to send and receive non-entity data.

---

**IMPORTANT** All published WS-I web services must have no arguments or return types that are Guidewire entity instances.

---

For example, you could change arguments and return types to the following:

- **Gosu class instances** – Write Gosu classes that include the important properties for that integration point. For example, a check printing service might only need the amount and the mailing address, but not any associated metadata or notes about the check. To work with WS-I web services, you must add the `final` keyword on the class, and also add the `@WsiExportable` annotation on the class. For example:

```
package example.wsi.myobjects
uses gw.xml.ws.annotation.WsiExportable

@WsiExportable
final class MyCheckPrintInfo {
    var checkID : String
```

```
    var checkRecipientDisplayName : String  
}
```

- **XML types created with the Guidewire XML modeler** – Use the Guidewire XML modeler tool (also called the *GX modeler*) to generate a model that contains only the desired subset of properties for each entity for each integration point. Then you can import or export XML data using that GX modeler as needed. See “The Guidewire XML (GX) Modeler” on page 302 in the *Gosu Reference Guide*.

## Changes for Consuming Web Services in WS-I

The syntax for consuming WS-I web services is different from RPCE (old style) web services.

There are important differences in how this works in PolicyCenter 7.0.0.

Note these important differences:

- In Studio, all RPCE web services appear as icons underneath a single icon. In previous releases, this was labelled **Web Services**. In version 7.0.0, the RPCE web services all appear as they used to appear, except that the **Web Services** icon now says **RPCE Web Services**.
- In Studio, WS-I web service resources now exist as a new resource type called a *web service collection*. The location of the web service collection in the package hierarchy defines the package for the types that Gosu creates from the associated WSDL.
- These WS-I web service resources appear in the same resource hierarchy as Gosu classes. For example, navigate to a package in the Gosu classes hierarchy. Then right-click, then choose **New → Web Service Collection**.

For more information on WS-I SOAP web services and web service collections, see “Consuming WS-I Web Service Overview” on page 79 in the *Integration Guide*.

## Finding Documentation for Connecting to Web Services

For both consuming and publishing web services, the WS-I syntax is different from the RPCE syntax. Be sure to read the correct documentation for the type of technology you are using, as well as whether you are publishing or consuming a web service from PolicyCenter.

Type of Web Service	Documentation for publishing web services	Documentation for consuming web services from Gosu
WS-I	“Publishing Web Services (WS-I)” on page 45 in the <i>Integration Guide</i>	“Consuming WS-I Web Service Overview” on page 79 in the <i>Integration Guide</i>
RPCE	“Publishing Web Services (RPCE)” on page 105 in the <i>Integration Guide</i>	“Calling RPCE Web Services from Gosu” on page 97 in the <i>Integration Guide</i>

### See also

- For an introduction to web services, see “Web Services Introduction” on page 39 in the *Integration Guide*.
- For the list of all built-in web services and whether each one is RPCE or WS-I, see “Reference of All Built-in Web Services” on page 41 in the *Integration Guide*.
- For background about the WS-I standard and its use of the Document Literal encoding, see “Consuming WS-I Web Service Overview” on page 79 in the *Integration Guide*.

## Suite Configuration File Overrides URLs to Guidewire Applications

PolicyCenter now has a suite configuration file for web service configuration. This only applies to inter-application integrations among Guidewire applications. This file provides a central place to configure (override) the URLs for other Guidewire products in the suite.

For inter-application Guidewire integrations that use WS-I web services, always use the suite configuration file to set the URLs. The suite configuration file is at the path:

`PolicyCenter/modules/platform/config/suite-config.xml`

Do not modify that file directly. Copy it to the configuration module and edit it there.

The file looks like the following:

```
<suite-config xmlns="http://guidewire.com/config/suite/suite-config">
  <product name="cc" url="http://localhost:8080/cc"/>
  <product name="pc" url="http://localhost:8180/pc"/>
  <product name="ab" url="http://localhost:8280/ab"/>
  <product name="bc" url="http://localhost:8580/bc"/>
</suite-config>
```

When importing WS-I WSDL, PolicyCenter does the following:

1. Checks to see whether there is a WSDL port of element type `<gwwsdl:address>`. If this is found, PolicyCenter ignores any other ports on the WSDL for this service.
2. If so, it looks for shortcuts of the syntax  `${PRODUCT_NAME_SHORTCUT}`. For example:  `${pc}`
3. If that product name shortcut is in the `suite-config.xml` file, PolicyCenter substitutes the URL from the XML file to replace the text  `${PRODUCT_NAME_SHORTCUT}`. If the product name shortcut is not found, Gosu throws an exception during WSDL parsing.

For web services that Guidewire applications publish, all WSDL documents have the `<gwwsdl:address>` port in the WSDL. The Guidewire application automatically specifies the application that published it using the standard two-letter application shortcut. For example, for PolicyCenter the abbreviation is pc. For example:

```
<wsdl:port name="TestServiceSoap11Port" binding="TestServiceSoap11Binding">
  <soap11:address location="http://172.24.11.41:8480/pc/ws/gw/test/TestService/soap11" />
  <gwwsdl:address location="${pc}/ws/gw/test/TestService/soap11" />
</wsdl:port>
```

Always use the `suite-config.xml` file to override the URL including the port number to connect to other Guidewire applications using this shortcut system.

The suite configuration file is used only for WS-I web services, not RPCE web services.

## Setting Locale in WS-I Requests for Guidewire Servers Only

For WS-I web services hosted on a Guidewire application, you can set (override) the international locale for this web service. See “Setting Locale in a Guidewire Application” on page 92 in the *Integration Guide* and “Setting Locale in a Guidewire Application” on page 92 in the *Integration Guide*.

For integrations among Guidewire applications, if the published web service is a WS-I service, the client application will set the locale as appropriate.

## Changes to Locally-Accessed RPCE Web Services

You can call RPCE web services published on the same server as an outgoing (published) SOAP endpoint. This is primarily used for testing your own SOAP APIs or calling PolicyCenter SOAP APIs that only exist as SOAP APIs.

Starting in version 7.0.0, before you use the `soap.local.api.*` syntax, you must regenerate the local files related to RPCE services, including after adding, changing, or removing web services. At a command prompt, change the working directory to `PolicyCenter/bin` and type the following command:

```
gwpc regen-soap-local
```

---

**IMPORTANT** To access local SOAP APIs for RPCE web services, you must call the `gwpc regen-soap-local` script to regenerate local files. Also, after you add, modify, or delete an RPCE web service, call `regen-soap-local` again.

---

## IAccountPlugin Web Service Changes for Transferring Policies

There is a new method on the `IAccountPlugin` web service called `transferPolicies`. It transfers policies to another account.

## SubmissionAPI Changes

In PolicyCenter 7.0, the `SubmissionAPI` web service follows the WS-I standard instead of the earlier RPC-encoded standard for publishing web services. The API methods throw a `SOAPException` if a communication error occurs. For complete information, see “Submission Web Services” on page 165 in the *Integration Guide*.

The most significant change to the API is how to pass information from the external system. The methods of `SubmissionAPI` accept a `policyPeriodData` parameter as a `String`. For the default implementation, this `String` data must be in XML format.

The Submission API methods call the `PopulatorPlugin`, new in PolicyCenter 7.0. The plugin method `populatePolicyPeriod` contains logic to parse the XML data passed in the `policyPeriodData` parameter. To implement an integration with your external system, you must modify the implementation of this method to process the XML format from your external system. The XML format and the parsing logic are defined by you and your external system, not by PolicyCenter.

For more information, see “New Populator Plugin” on page 136.

## Changes to the Start Draft Submission Method

The Submission API changes some parameters in the `startDraftSubmission` method to be WS-I compliant.

The method signature is:

```
function startDraftSubmission(accountNumber : String, productCode : String, producerCodeId : String,  
    policyPeriodData : String, parseOptions : String) : String
```

The parameters are:

- Account number as a `String`, the same is in PolicyCenter 4.0
- Product code as a `String`, the same as the third parameter in PolicyCenter 4.0
- Producer code public ID as a `String`, the same as the second parameter in PolicyCenter 4.0
- An XML representation of policy period data from the external system as a `String`, which replaces the fourth parameter in PolicyCenter 4.0
- A set of parse options as a `String` to control how your implementation parses the policy period data

The method returns a `String` that contains the job number of the draft submission.

## New Quote Submission Method

The Submission API adds the `quoteSubmission` method, which is WS-I compliant.

The method signature is:

```
function quoteSubmission(accountNumber : String, productCode : String, producerCodeId : String,  
    policyPeriodData : String, parseOptions : String) : String
```

The parameters are:

- Account number as a `String`
- Product code as a `String`
- Producer code public ID as a `String`, the same as the second parameter in PolicyCenter 4.0
- An XML representation of policy period data from the external system as a `String`, which replaces the fourth parameter in PolicyCenter 4.0
- A set of parse options as a `String` to control how your implementation parses the policy period data

The method returns a `String` that contains the job number of the quote submission.

---

#### UPGRADE TASK

**Priority:** Before deploying a production server

**Summary:** Review changes to the Submission API web service interface and change your implementations accordingly

---

## PolicyAPI Changes

In PolicyCenter 7.0, the `PolicyRenewalAPI` web service follows the WS-I standard instead of the earlier RPC-encoded standard for publishing web services. The API methods throw a `SOAPException` if a communication error occurs. For complete information, see “Policy Web Services” on page 168 in the *Integration Guide*.

## PolicyRenewalAPI Changes

In PolicyCenter 7.0, the `PolicyRenewalAPI` web service follows the WS-I standard instead of the earlier RPC-encoded standard for publishing web services. The API methods throw a `SOAPException` if a communication error occurs. For complete information, see “Policy Renewal Web Services” on page 172 in the *Integration Guide*.

### Changes to the Renewal Method

The `PolicyRenewalAPI` interface method `renewNewPersonalAutoPolicy` in PolicyCenter 4.0 is removed. It changed in PolicyCenter 7.0. The API interface now has a more general purpose method called `startNewRenewal`. Specify the type of policy to renew using the `productCode` method parameter, which is a `String` object. Put the new policy data in the `policyPeriodData` parameter, which is a `String` that contains an XML representation of policy data in the external system.

To implement an integration with your external system, you must modify the implementation of `PolicyRenewalAPI` to process the XML format from your external system. You do not need to modify the interface with methods for specific policy types, nor do you need to modify the method signature of `startNewRenewal`.

The method signature is:

```
function startNewRenewal(accountNumber : String, productCode : String, producerCodeId : String,  
    policyPeriodData : String, parseOptions : String) : String
```

The parameters include:

- Account number as a `String`
- Product code as a `String`
- Producer code public ID as a `String`
- An XML representation of policy period data from the external system as a `String`
- A set of parse options as a `String` to control how your implementation parses the policy period data

The method returns a `String` that contains the job number of the newly started renewal.

Also see “New Populator Plugin” on page 136.

### New Term Confirmation Method

The the `PolicyRenewalAPI` web service in PolicyCenter 7.0 includes a new method to confirm the term of a policy, `confirmTerm`. External systems call this method when a payment is received as part of the *Confirmed Renewals* flow.

The method signature is:

```
function confirmTerm(policyNumber : String, termNumber : int, transactionId : String)
```

The parameters include:

- Policy ID as a `String`
- Term number as an `int` that identifies the term to confirm
- Transaction ID as a `String` that identifies the payment transaction in the external system

The method returns nothing.

Also see:

- “New Populator Plugin” on page 136.
- “Confirming Renewals Integration Flow” on page 145.

---

#### UPGRADE TASK

**Priority:** Before deploying a production server

**Summary:** Review changes to the Policy Renewal API web service interface and change your implementations accordingly

---

## Confirming Renewals Integration Flow

PolicyCenter 7.0 has additional choices for renewals integration flow with a billing system.

In PolicyCenter 4.0, the available renewal flows were:

- *bind-and-cancel*
- *renewal offers*

These two are still available but there is a new optional renewal flow called *confirmed renewal*. For more information, see “Confirmed Renewals Flow” on page 529 in the *Integration Guide*. Also, see a table that compares and contrasts each flow in “Multiple Choices of Renewal Flow” on page 525 in the *Integration Guide*.

In the default configuration of PolicyCenter 7.0, all lines of business use the *bind-and-cancel* flow. For the business auto LOB, this is a change, since in PolicyCenter 4.0 it used renewal offers in the default configuration.

In all cases, you can configure any line of business to use any desired renewal flow. You can even choose which renewal flow at run time for each `PolicyPeriod`. Refer to “Confirmed Renewals Flow” on page 529 in the *Integration Guide* for details.

## Guidewire InsuranceSuite Plugin Implementations are Versioned

If you have more than one Guidewire application, be aware that some of the plugins exist primarily to connect to other Guidewire applications. If you want to use the built-in plugin implementation to connect to other Guidewire applications, ensure that you use the right version of the plugin implementation. The name of the classes are equivalent but are in different packages, which include application version numbers.

The last part of the package name is the Guidewire application two-digit abbreviation, followed by the application version number with no periods. For example, the shortened version of *BillingCenter 3.0.0* is bc300.

For example, the PolicyCenter plugin that connects to BillingCenter 3.0.0 is at the fully-qualified path:

`gw.plugin.billing.bc300.BCBillingSystemPlugin`.

---

**IMPORTANT** For suite integrations with other Guidewire applications, you must choose the plugin implementation class that matches the version of your applications. Choose the one with the proper version number of the other application (not the current application) in its package name.

---

For existing customers, this may be an upgrade task to update implementations classes in the Studio Plugins editor. Set the implementation class to the version of your other Guidewire application. If you have questions about this process, contact Guidewire Customer Support.

---

#### UPGRADE TASK

**Priority:** Before your next upgrade

**Summary:** Update your plugin implementation classes to the version of your Guidewire application.

---

## SOAP Implementation Classes and WSDL Packages Include Version

PolicyCenter changed the package of some SOAP API types to include the version number of the application.

There are two types of changes:

### Web Service Implementation Classes Are Organized by PolicyCenter Version

The web service implementation classes are now organized by the version number of the current application. This makes it easier over time for Guidewire applications to change their API but still maintain backward compatibility with previous versions of web services that still exist in the application.

The full package name includes one component with the Guidewire application two-digit abbreviation followed by its version number with no periods. For example, the shortened version of *BillingCenter 3.0.0* is *bc300*.

For example, the PolicyCenter 7.0.0 version of this API is at the fully qualified path:

`gw.webservice.pc.pc700.job.ICancellationAPI`

In contrast, the PolicyCenter 4.0.0 backward compatible version of the API is at the fully qualified path:

`gw.webservice.pc.pc400.job.ICancellationAPI`

### WSDL for Guidewire Applications Are Organized by Destination Application Version

WSDL files are files that describe how to use a web service. Starting in PolicyCenter 7.0.0, the WSDL files to another Guidewire application are organized by the version of the destination application. Additionally, all WSDL files are encapsulated as web service collections in Studio. Web service collections are the new way to encapsulate remote resources like WSDL and XSD files in Studio. Refer to “Consuming WS-I Web Service Overview” on page 79 in the *Integration Guide*.

The full package name includes one component with the Guidewire two-digit abbreviation for the application, followed by its version number with no periods. For example, the shortened version of *BillingCenter 3.0.0* is *bc300*.

This organization of the WSDL (and web service collections) makes it easier to ensure compatibility with a specific version of the destination application. A remote application’s web services may change between releases. If that happens, you can choose to connect to the older or newer version of that web service API in the destination application.

The files are in the package:

`wsi.remote.APP_ABBREVIATION.APP_ABBREVIATION_AND_VERSION`

For example, to connect to the 3.0 versions of BillingCenter, the web collection is at the Studio path in the Classes hierarchy:

```
wsi.remote.bc.bc300
```

#### UPGRADE TASK

**Priority:** Before starting the server

**Summary:** Change references to SOAP and WSDL packages to paths that include version numbers.

## Geocoding Only Some Addresses

PolicyCenter 7.0 added a property in the data model of the Address entity. This new property is called `BatchGeocode`. Callers of the `GeocodePlugin` can use this property to filter which addresses to pass to the plugin for geocoding. If you write your own implementation of the `GeocodePlugin`, ignore the `BatchGeocode` property. Callers of the plugin are responsible for determining which addresses to geocode.

## Changes to PolicyCenter Contact-related Plugins

The contact-related plugins changed in PolicyCenter version 7.0.

In PolicyCenter 4.0, there were the following:

- `IContactSystemPlugin` plugin – Does most of the work of communicating with the external contact manager.
- `IContactConfigPlugin` plugin – Configured some elements of how PolicyCenter deals with contacts. For example, configuring how PolicyCenter maps contact types to account contact roles.
- `AccountContactPlugin` plugin – Copies data model extensions between account contacts
- `AccountSyncable` plugin – Customizes how PolicyCenter synchronizes contact with accounts.

### Changes Related to Contact System Plugin

In PolicyCenter 7.0, the `IContactSystemPlugin` plugin is deprecated because there is a new version of the plugin with different method signatures. You can still use it with your legacy code in this release. However, start converting your code to the newer version called `ContactSystemPlugin`.

The two plugins have some differences:

- The `ContactSystemPlugin` plugin has a new method called `supportsFindingDuplicates`. It returns `true` if and only if the plugin supports finding duplicate contacts.
- There is another new method to overwrite contacts with latest values:  

```
void overwriteContactWithLatestValues(Contact contact, String addressBookUID);
```
- The `retrieveContact` method signature changed. Instead of returning a `ContactSoapWrapper` object, it simply returns a regular `Contact` entity. The basic behavior is otherwise unchanged.
- The `searchContacts` method signature changed. Instead of returning an array of `ContactSearchResultSoapWrapper` objects, it returns an array of objects of type `gw.plugin.contact.ContactResult`. The basic behavior is otherwise unchanged.
- The `addContact` method has an extra argument for a transaction ID string. There is an alternate method signature that also has a text-based payload argument. For example, sending an XML encoding of the contact.
- The `updateContact` method has an extra argument for a transaction ID string.
- The `removeContact` method has an extra argument for a transaction ID string.

For details, see “Contact Integration” on page 573 in the *Integration Guide*.

### Changes Related to Contact Configuration Plugin

The plugin `IContactConfigPlugin` has a minor change:

- There is a new method that checks the given `ContactSearchCriteria` to see if the minimum search criteria is set. The method signature is:

```
boolean minimumCriteriaSet(ContactSearchCriteria contactSearchCriteria);
```

For details, see “Configuring How PolicyCenter Handles Contacts” on page 579 in the *Integration Guide*.

### New Contact Account Role Plugin

In addition to the account contact plugin mentioned earlier, there is a new plugin interface called `IAccountContactRolePlugin`. Implement this plugin interface to configure how to copy data model extensions between account contact roles. For details, see “Account Contact Role Plugin” on page 581 in the *Integration Guide*.

---

#### UPGRADE TASK

**Priority:** Before your next upgrade

**Summary:** Replace deprecated contact system plugin with the latest version with a different name.

---

## ContactAPI Web Service Changes

The web service `ContactAPI` changed significantly. Refer to the implementation class `ContactAPI.gs` in Studio for the method signatures.

One major part of this change is changing to use XML (XSD-based) types for use with WS-I compliant web services. You can see which XML types are used by looking at the implementation class. For each one of these XSD-based types, you can find a Guidewire XML (GX) Model to represent a contact.

For example, the `addContact` method now uses this as an argument:

```
function addContact(externalContact :  
    gw.webservice.pc.pc700.gxmodel.contactmodel.types.complex.Contact) : String {
```

As you look at the fully-qualified path, look for the part of the path before “types”. That indicates the path of the XML model in Studio:

```
PolicyCenter/modules/pc/gsrc/gw/webservice/pc/pc700/gxmodel/ContactModel.gx
```

View and edit that model as desired to update the XML definition.

#### See also

- “The Guidewire XML (GX) Modeler” on page 302 in the *Gosu Reference Guide*

## Improvements to Matching Classes

There are a few parts of PolicyCenter that need specialized logic to answer the question, “do these two objects represent the exact same thing?”. These include:

- The policy difference page. For example, in a policy change job, comparing a policy to the original. Other situations use this same interface, such as for side-by-side quoting for multiple draft revision jobs.
- Out-of-sequence job handling
- Preemption job handling

To determine whether two objects represent the same real-world thing, PolicyCenter runs code that answers the question definitively. For example, to compare two cars, you might compare the unique Vehicle Identification Numbers (VIN numbers). If the VIN numbers match, they represent the same car, independent of the number of other properties on the car that might be different.

In PolicyCenter 4.0, the matching all happened within the `PCBeanMatcher.gs` file. See “Compare Individual Objects with Matchers” on page 487 in the *Integration Guide* for details of customizing that file.

In PolicyCenter 7.0, there are additional entity types that use this `PCBeanMatcher` style of matching.

Additionally, there are new matching classes that fill the same need as `PCBeanMatcher`, but are implemented in a different way. The method names are different in this new style of matcher, which are also called delegate-based matchers.

### How the Delegate-based Matchers Work

The new matchers use a delegate interface to define the required methods. The delegate interface depends on what type of object it is:

- For effective-dated subobjects in a `PolicyPeriod` graph, the matchers use the `EffDatedLogicalMatcher` interface.
- For all other entity types, the matchers use the `LogicalMatcher` interface.

To define a new matcher, use the right interface and define the necessary methods. You must be able to know the set of fields that uniquely identify an entity. For example, the `LogicalMatcher` interface has a method called `genKey` that generates a unique key that can be compared to determine if the objects match.

If the keys for two objects do not match, they are definitely not the same object.

If the keys for two objects match, they might or might not be the same object. To determine for sure, PolicyCenter calls the `isLogicalMatchUntyped(KeyableBean bean)` method, which may do additional checks.

This method might check some set of parent foreign keys that must also match. For example, a `VehicleDriver` would match if the foreign keys to `PersonalVehicle` and `PolicyDriver` also match.

About 100 entity types now implement one of the matcher delegates and did not have explicit code in `PCBeanMatcher`. In contrast, the `PCBeanMatcher` matches about 30 entity types, most notably including `Cost` objects.

The base comparison of comparing `Id` and `FixedID` properties for all entities remains in the `PCBeanMatcher` class.

The default implementation of the delegate-based matchers relies on defining columns on the entity, including parent foreign key links to determine whether two objects match.

General tips with writing new matchers:

- You can match on whatever you wish, but the safest approach is to use only true database-backed columns. generally speaking is dangerous.

---

**WARNING** Matching on non-database-backed columns, such as dynamically-generated Gosu properties, can be dangerous. Contact Guidewire Customer Support for guidance before implementing matchers that use anything other than database-backed columns.

---

- Columns must be very immutable, something that never changes over the life of the object. A car’s VIN number is a good example because VIN numbers do not change over time and editing it never make sense (other than correcting a typo).

### Which PolicyCenter Types Switched from BeanMatcher to the New Matchers?

The following types switched from the BeanMatcher system to the new matcher classes:

- `RateFactor`
- `CoverageSymbol`
- `Clause`
- `PolicyContactRole`
- `LocationAnswer`

- WCJurisdiction
- WCCoveredEmployee
- WCModifier
- PersonalVehicle
- PersonalVehicleCov
- CPLocation
- PolicyLocation
- BOPBuildingCov
- BOPBuilding
- BusinessVehicle
- BusinessVehicleCov
- BAJursModifier
- BAJursModifier

### **Customize Which Classes Perform Matching**

The policy difference system always initially uses the `PCBeanMatcher` class. Similarly, the PolicyCenter out-of-sequence handling system initially calls `PCBeanMatcher`.

However, the `PCBeanMatcher` always uses the `LogicalMatcher` delegate if one is available. The `EffDatedLogicalMatcher` class is a subtype of `LogicalMatcher`.

The `PCBeanMatcher` file is editable, so you change the matching behavior that you want to use when generating a list of differences.

### **Matchers and Copiers**

Related to the matcher classes are copier classes, which know how to completely duplicate an entity. In a typical policy difference operation, the difference generation is read-only. The difference generation itself does not need to copy data from one entity to a duplicate of that entity.

However, in some cases, PolicyCenter also uses a copier. For example, an out-of-sequence job or a preemption job might need to do the following process, which is sometimes called *DuplicateAdd*:

1. Generate a policy difference between branches to find added entity instances
2. Use matchers to see if new entity instances have a match somewhere in the graph
3. If they do match, use copiers to ensure that they are fully in sync as defined by each entity type.

Because of this, if you write a new matcher class, then you must write a copier class for your data. Otherwise, *DuplicateAdd* does not work as expected. PolicyCenter might extend an existing entity, but lose (not copy over) the new values.

---

**WARNING** If you write a new matching class, write a new copier class.

---

See “Configuring Copy Data in a Line of Business” on page 191 in the *Product Model Guide*.

## **Escalation and Exception Plugins**

There are several optional exception and escalation plugins. By default, they just call the associated rule sets and perform no other function. Implement your own version of the plugin and register it in Studio if you want something other than the default behavior. The following table lists the exception and escalation plugins:

Name	Plugin interface	The default action
Activity escalation	<code>IActivityEscalationPlugin</code>	Calls the activity escalation rule set
Group exceptions	<code>IGroupExceptionPlugin</code>	Calls the group exception rule set
User exceptions	<code>IUserExceptionPlugin</code>	Calls the user exception rule set

For more details, see “Exception and Escalation Plugins” on page 312 in the *Integration Guide*.

## Billing Plugin Changes

The billing system plugin, which is the main plugin for connecting to an external billing system, changed.

There are several new methods on the plugin:

- `addPaymentInstrumentTo` – Add a payment instrument to an account. and return a billing system instrument. For example, a customer credit card number or a token that represents it in an external payment system.
- `getExistingPaymentInstruments` – Return an array of billing system instruments
- `getCompatibilityMode` – The compatibility mode to use in billing integration. Set to 4.0.0 to get the behavior used in PolicyCenter 4.0. For example, if you return 7.0.0, PolicyCenter sends contact updates not to the billing system but to the contact system. If you enter 4.0.0, PolicyCenter sends contact updates directly to the billing system.

---

**IMPORTANT** The `getCompatibilityMode` method is essential for PolicyCenter integration. Choose your value for this setting carefully.

- `searchForAccounts` – Search for accounts in the billing system using a search criteria. The method signature is the following:

```
BillingAccountSearchResultJava[] searchForAccounts(  
    BillingAccountSearchCriteriaJava searchCriteria, Integer limit);
```

- `getInvoiceStreams` – Retrieve all invoice streams for a given account. The method signature is the following:

```
BillingInvoiceStreamInfoJava[] getInvoiceStreams(String accountNumber);
```

- `getSubAccounts` – Retrieve all subaccounts of the account with the given account number. If the account does not exist, returns an empty array. The search for subaccounts is recursive and returns subaccounts of subaccounts also. The method signature is the following:

```
BillingAccountSearchResultJava[] getSubAccounts(String accountNumber);
```

## Billing Summary Plugin Changes

The interfaces for the plugin and the special data types it returns changed. In PolicyCenter 4.0, the `IBillingSummaryPlugin` returned interfaces that defined a set of properties, for example `PolicyPeriodSoapWrapper`. To integrate with an external billing system, you extended these interfaces with properties unique to your external system.

In PolicyCenter 7.0, the method signatures of the plugin interface changed to return interfaces that define no properties, for example, `BillingPeriodInfoJava`. They are marker interfaces only. To implement your own plugin, create your own classes that implement the type interfaces and include the properties that you need to retrieve from your external billing system.

For more information, see “Implementing the Billing Summary Plugin” on page 548 in the *Integration Guide*.

## For Messaging, a Contact is Now a Safe-ordered Object

Starting in PolicyCenter 4.0.1, the primary entity for messaging was Account. This means that any safe ordering happens on the Account object. In PolicyCenter 7.0, both Account and Contact are both safe-ordered objects.

This means that any account-related message will be sent in strict order for each destination, and wait for acknowledgement before sending the next message for that same account. Also, any contact-related message will be sent in strict order for each destination, and wait for acknowledgement before sending the next message for that same contact.

For more information about safe-ordering, see “Message Ordering and Multi-Threaded Sending” on page 375 in the *Integration Guide*.

## Integrating PolicyCenter 7.0.0 with ClaimCenter 6.0.0

If you need to integrate PolicyCenter 7.0.0 with ClaimCenter 6.0.0:

1. In PolicyCenter, find the class `gw.webservice.pc.pc700.ccintegration.CCPolicySearchIntegration`
2. Comment out the RPC web service declaration so it looks like:  
`//@RpcWebService'`
1. In PolicyCenter, find the class `gw.webservice.pc.pc400.ccintegration.CCPolicySearchIntegrationV2`
2. Uncomment the RPC web service declaration so that it looks like:  
`@RpcWebService`

## AccountSyncable Changes

The `IAccountSyncablePlugin` changed. No methods from PolicyCenter 4.0 remain.

Instead, the only method is `refreshAccountInformation`, which must refresh any necessary account information to ensure the account syncable will use the most current data when calling the other methods. It takes an entity instance of a type that implements the main `AccountSyncable` interface.

The default implementation of this plugin simply calls the `refreshAccountInformation` method on the `AccountSyncable`.

On a related note, in PolicyCenter 7.0, the `AccountSyncable` interface (which account syncable entities implement) changed significantly. The only method from PolicyCenter 4.0 that remains is `refreshAccountInformation`.

Additionally, there are large number of new methods. The easiest way of seeing how to implement them in Gosu is to look at the Gosu class: `gw.api.domain.account.AbstractAccountSyncableImpl`. That is an abstract class that is the superclass for delegate adapters for account syncable objects.

### Account Syncable Adapters

PolicyCenter 7.0 changed not only the `AccountSyncable` delegate but also its adapters, which are the objects that implement the `AccountSyncable` delegate for each type of object. The changed adapters are as follows:

- `gw.account.AbstractAccountSyncableAdapter.gs`
- `gw.contact.PolicyContactRoleSyncableAdapter.gs`
- `gw.lob.pa.contact.PolicyDriverSyncableAdapter.gs`
- `gw.lob.wc.contact.PolicyOwnerOfficerSyncableAdapter.gs`
- `gw.policyaddress.PolicyAddressSyncableAdapter.gs`
- `gw.policylocation.PolicyLocationSyncableAdapter.gs`
- `gw.api.domain.account.AccountSyncableAdapter.java`

The old `AccountSyncablePlugin` used to have the methods `handleSyncableEvent` and `isSyncedToAccount`. In PolicyCenter 7.0, functionality is in the `AccountSyncable` interface and subsequent implementations. For example, `PolicyOwnerOfficer.eti` in PolicyCenter 4.0 look like,

```
<subtype
  xmlns="http://guidewire.com/datamodel"
  entity="PolicyOwnerOfficer"
  supertype="WCPolicyContactRole">
  <implementsEntity
    name="AccountSyncable"
    adapter="gw.lob.wc.contact.PolicyOwnerOfficerSyncableAdapter"/>
  ...
</subtype>
```

In PolicyCenter 7.0 it looks like:

```
<subtype
  xmlns="http://guidewire.com/datamodel"
  entity="PolicyOwnerOfficer"
  supertype="WCPolicyContactRole">
  <implementsInterface
```

```
iface="gw.api.domain.account.AccountSyncable"
impl="gw.lob.wc.contact.PolicyOwnerOfficerAccountSyncableImpl"/>
...
</subtype>
```

Generally speaking, the code that was implemented separately in each adapter is now implemented in the various `AccountSyncable` implementations. For example, the `PolicyOwnerOfficerSyncableAdapter` had to specifically override the `copyAllAccountFields` method to add its special fields to the list of properties to copy:

```
override function copyAllAccountFields() {
    super.copyAllAccountFields()
    _policyOwnerOfficer.setFieldValue("RelationshipTitleInternal",
        _policyOwnerOfficer.RelationshipTitle)
}
```

In PolicyCenter 7.0, however, each account syncable provides a set of `AccountSyncedFields` that it uses. These fields define the binding between the account-level entity and its field with the policy-level account syncable and its field. The equivalent functionality is now implemented more generically in the `AbstractAccountSyncableImpl` class mentioned earlier in this topic.

```
override function copyPolicyContractDataUnchecked() {
    AccountSyncedFields.each(\ field -> field.copyFromAccountToPolicy(_accountSyncable))
}
```

As a result, converting an adapter from PolicyCenter 4.0 to PolicyCenter 7.0 mostly involves introducing new `AccountSyncedFields` objects as necessary if new account-level data is shared but then revised. Next, ensure that the `AccountSyncable` implementation returns the correct list of `AccountSyncedFields`.

### Responsibilities Change for the Plugin

In PolicyCenter 4.0, the `AccountSyncablePlugin` was responsible for two things:

- Deciding when an account syncable was “synced to the account”
- Handling the various `AccountSyncableEvents` options, such as quoting, out of sequence merge, and difference generation.

In PolicyCenter 7.0, this behavior became complex. It was worthwhile to create a rich subtype hierarchy that shared behavior as needed. The `isSyncedToAccount` method now exists directly on the account syncable implementation. Instead of the PolicyCenter 4.0 `AccountSyncablePlugin` method called `handleAccountSyncableEvent` taking a typecode from a typelist, account syncables now have a `prepareForEVENT_NAME` methods. As a result, instead of calling `handleSyncableEvent("Promote")`, the delegate has a method `prepareForPromote`.

The `refreshAccountInformation` method remains on the `AccountSyncablePlugin` interface. It ensures that changes to the account-level entities in a different bundle become immediately visible. You can easily replace this method in certain circumstances, such as during performance testing data generation.

## Changes to Messaging System Safe Ordering

Starting in PolicyCenter 7.0.1, the messaging system supports safe-ordering of messages by `Account` or `Contact`. In PolicyCenter 4.0 and PolicyCenter 7.0.0, the messaging system supported safe ordering only by `PolicyPeriod` as the primary entity for the application.

Safe ordering is a messaging feature that prevents related messages from sending out of order from each other. This is called *safe order*. Messages are grouped by their related *primary object* for each messaging destination.

**Note:** For definitions, see “Primary Entity and Primary Object” on page 343 in the *Integration Guide*.

Messages send in creation order with other messages associated with that same primary object for that destination. Any messages associated with a primary object for that destination are called *safe-ordered* messages. The application waits for an acknowledgement before processing the next safe-ordered message for that same primary object for that destination. In other words, delays or errors for that destination blocks further sending of messages for that same destination for that same primary object.

In PolicyCenter, by default messages associated with an account send grouped by account for each messaging destination and sent in creation order. These are *safe-ordered* messages. If a messaging destination sets **Contact** its *alternative primary entity*, messages associated with a contact send grouped by contact for each messaging destination and sent in creation order.

There are PolicyCenter objects that are not associated with a primary entity. For example, **User**. All messages that are not associated with a claim (the primary object) are called *non-safe-ordered* messages.

Not all messages are associated with a primary object. The logic for how and when to send these *non-safe-ordered messages* is different from the logic for safe-ordered messages. Additionally, the behavior varies based on the messaging destination configuration setting called **Strict Mode**. For details, see “Message Ordering and Multi-Threaded Sending” on page 375 in the *Integration Guide*.

# New and Changed in System Administration in 7.0

## New in System Administration in 7.0

This topic describes what is new for system administration of PolicyCenter 7.0. This includes:

- “DCE VM Strongly Recommended for Development Environments” on page 155
- “Support for Importing and Exporting Policy Forms and Policy Holds” on page 157

### DCE VM Strongly Recommended for Development Environments

The Dynamic Code Evolution Virtual Machine (DCE VM) is a modified version of the Java HotSpot Virtual Machine (VM). The DCE VM supports any redefinition of loaded classes at runtime. You can add and remove fields and methods and make changes to the super types of a class using the DCE VM. The DCE VM is an improvement to the HotSpot VM, which only supports updates to method bodies.

Guidewire strongly recommends the use of the DCE VM for development. Guidewire does not support the DCE VM in a production environment. Performance of the Java Virtual Machine might be impacted by the addition of the DCE VM.

Starting in PolicyCenter 7.0, Gosu becomes a compiled instead of an interpreted language. In prior releases, Gosu code ran on top of the JVM through an intermediate layer. So, users could change code (and alter execution) at any time, even in a running system. In PolicyCenter 7.0, Gosu code is translated to Java bytecode instructions and registered directly with the JVM (in class form) for execution.

This provides two significant benefits. First, Gosu code runs natively on the VM. Native execution is much faster than interpreted execution. Second, Gosu classes can be directly inspected, instrumented, and tuned just like their Java-based equivalents. Development environments, debugging tools, stack traces, memory analysis tools, and many other tools are able to work directly with Gosu classes and treat them as first-class objects.

However, since Gosu classes are now registered directly with the JVM, a new limitation is imposed by JDks. Just like Java classes, Gosu classes cannot be modified once they have been loaded. There are three ways to make a change to a compiled class in a JVM:

- Restart the JVM. This is easy, but requires a full restart.
- Drop the classloader that loaded the class and force a reload from disk. Only supported if the software is designed to do this for special classes and only performs well for small numbers of loaded classes.
- Use JVM hot swapping. This is the option preferred by Guidewire.

Recently, the Java language specification has added a "swap API" to allow VMs to implement class changes while running. This API provides for the ability to reload changed classes on an as-needed, request basis. Development environments, such as Eclipse, WSAD, or Guidewire Studio, can invoke requests against a running VM to ask the VM to reload a new version of a class. Individual vendor VMs are not required to implement this API. VMs can implement a subset of the API, or they can provide a full, general reload facility.

Following is a summary of reload support for the two Guidewire supported JDks:

#### **IBM JDK**

The only types of changes you can reload are:

- Adding or editing (but not deleting) methods
- Adding and deleting classes

New limitations compared to PolicyCenter 4.0: Not being able to delete methods, change the method name, or change a method signature

#### **Sun JDK**

The only types of changes you can reload are: (limited)

- Editing (but not adding or deleting) methods
- Adding and deleting classes

New limitations compared to PolicyCenter 4.0: Not being able to add or delete methods, change the method name, or change a method signature

In addition to these supported VMs, Guidewire is providing a third option, DCE VM applied to the Sun JDK. DCE VM is only available for the Sun JDK and not the IBM JDK. The DCE VM provides full, generalized reload support for all types of class changes. DCEVM 0.2 has been tested and is certified for use with PolicyCenter 7.0. This VM is optional but Guidewire strongly recommends using the VM for individual development workstations because it provides automatic support for all forms of Gosu and Java code changes.

Guidewire does not certify use of the DCE VM for shared development environment and integration testing, sandbox, test, staging, and production environments. Reloading is not relevant for these environments.

Guidewire certifies use of the DCE VM only for individual development workstations.

If you elect to use the Sun or IBM VM for individual development workstations, without the DCE VM, Studio still attempts to instruct the VM to reload Gosu class changes. In many cases, such as business rules and PCF files, the reloading works fine with these VMs, since the changes that can be made fall within the native reload capabilities. However, in some cases, changes to Gosu code fall outside the reload capabilities of the VM. If this happens, it is necessary to restart the server to reload the change in much the same way that the server is restarted for data model changes. This could lead to confusion since some Gosu changes are reloaded and others are not depending on the JDK.

#### **See also**

- “Installing the Dynamic Code Evolution Virtual Machine” on page 41 in the *Installation Guide*
- “Studio and the DCE VM” on page 91 in the *Configuration Guide*
- <http://java.net/projects/dcevm/>

## Support for Importing and Exporting Policy Forms and Policy Holds

PolicyCenter 7.0 adds support for importing and exporting Policy Form and Policy Hold entities. For instructions, see “Importing and Exporting Administrative Data” on page 87 in the *System Administration Guide*.

## Changed in System Administration in 7.0

This topic describes what has changed for system administration of PolicyCenter 7.0. This includes:

- “Updated System Requirements” on page 157
- “Database Configuration Changes” on page 157
- “Maximum Contact Search Results Parameter Name” on page 157

### Updated System Requirements

Guidewire has updated the system requirements to install PolicyCenter. See the *Guidewire Platform Support Matrix* for current system and patch level requirements. The *Guidewire Platform Support Matrix* is available from the Guidewire Resource Portal at <http://guidewire.custhelp.com>.

Guidewire has updated the supported version of the H2 database engine used for the QuickStart development environment configuration. The supported version of H2 uses a different file format than prior supported versions. H2 does not support an upgrade from the old file format to the new format. Guidewire does not support upgrading PolicyCenter on H2. You can use the `gwpc dev-dropdb` command to remove your old PolicyCenter H2 database before you upgrade. PolicyCenter creates a new H2 database if one is not found. Alternatively, you can change the `jdbcURL` in `config.xml` to point to an unused location for H2 files.

If you attempt to use PolicyCenter 7.0 with an old H2 database, you will encounter an error similar to the following:

```
Caused by: org.h2.jdbc.JdbcSQLException: Unsupported database file
version or invalid file header in file "Old database:
C:\tmp\guidewire\pc.data.db - please convert the database to a
SQL script and re-create it." [90048-147]
```

### Database Configuration Changes

The `maxWait` and `stmtpool.enabled` parameters are removed from the database configuration in PolicyCenter 7.0.

The `stmtpool.enabled` parameter has been removed completely. This parameter enabled Oracle statement caching and statement pooling when set to `true`. Guidewire has found that enabling statement caching and pooling leads to issues with too many open cursors.

The `maxWait` parameter has been removed from the default `config.xml` file. This parameter defines the maximum time in milliseconds that the data source waits for a connection before one becomes available in the pool to service. The default value of 30000 is usually sufficient. You can add the `maxWait` parameter to the `<database>` element in `config.xml`.

### Maximum Contact Search Results Parameter Name

The parameter that defines the maximum number of results returned from a contact search has been renamed. The new name is `MaxContactSearchResults`. In releases of PolicyCenter prior to 7.0, this parameter was named `ContactSearchMaxResult`. Guidewire renamed this parameter to be consistent with the name in ClaimCenter.



# New and Changed in Rules in 7.0

This topic covers what is new and changed with the PolicyCenter business rules PolicyCenter 7.0.

This topic includes:

- “Changes in Rules in PolicyCenter 7.0” on page 159

## Changes in Rules in PolicyCenter 7.0

This topic describes what has changed with rules in PolicyCenter 7.0.

### Assignment Rules Using Proximity Search

Guidewire has improved the performance of proximity search. To support this change, Guidewire has added the following:

- A new configuration parameter—`ProximityRadiusSearchDefaultMaxResultCount`—that is the maximum number of results to return if performing a radius ( $n$  miles or kilometers) search from PolicyCenter. The default is 1000. This parameter has no effect on ordinal (nearest  $n$ ) proximity searches. This parameter does not have to match the value of the corresponding parameter in the Guidewire ContactManager `config.xml` file. This parameter exists within ContactManager as well, but behaves slightly differently in that application. See the ContactManager documentation for details.
- A new field to `ProximitySearchParameters`—`RadiusSearchMaxResults`—that indicates the maximum number of items to return on a radius search:
  - If `DistanceBasedSearch` is `false`, then PolicyCenter ignores this value
  - If `DistanceBasedSearch` is `true`, and the value of `RadiusSearchMaxResults` is `null`, zero, or negative, PolicyCenter limits the search by the value of `ProximityRadiusSearchDefaultMaxResultCount`.

In addition, Guidewire has changed the default value of `ProximitySearchOrdinalMaxDistance` from 100 to 300.



# 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 4.0.0 Release Notes

## Release 4.0.0.107

Copyright © 2009 Guidewire Software, Inc. All rights reserved. Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire ContactCenter, Guidewire Insurance Suite, and the Guidewire logo are trademarks or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries. All other trademarks are the property of their respective owners.

This documentation is published as **Guidewire Confidential**. The contents of this documentation, including product architecture details and APIs, are considered confidential and are fully protected by customer licensing confidentiality agreements and signed Non-Disclosure Agreements (NDAs).

This product includes software developed by the Apache Software Foundation (<http://www.apache.org>).

## Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Changes in This Release
- Known Issues and Limitations

## Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

## Version Number

This release of Guidewire PolicyCenter is 4.0.0.107.

## Installation Requirements

For installation requirements, see the *PolicyCenter Installation Guide*.

## Support

This document includes important information about the current release of Guidewire PolicyCenter. For assistance with this software release, contact Guidewire Customer Support. You can contact support on the Web at <http://portal.guidewire.com>, by email at [support@guidewire.com](mailto:support@guidewire.com), or by phone at +1-650-356-4955.

## Installing This Release

For installation information, see the *PolicyCenter Installation Guide*.

This section describes the following changes to PolicyCenter:

- InetSoft Evaluation
- Upgrade Issues

### InetSoft Evaluation

You can evaluate InetSoft Style Report Enterprise Edition Version 10.1 using the following license key:

L000-798-ERX-0000BF00100001F-F2AFC9DD34ED

If you want to use InetSoft and the PolicyCenter reporting module in a production environment, contact Guidewire Customer Support to request a full license key. For more information, see the *PolicyCenter Reporting Guide*.

### Upgrade Issues

This section describes changes to the PolicyCenter base configuration that may cause upgrade issues.

#### Workflows removed (CC-53856)

- **Issue:** The IssueRenewalWF, PendingNonRenewalWF, PendingNotTakenWF, and PendingRenewalWF workflows were removed. There is no configuration upgrade step available at this time. There is also no mechanism to migrate in-process renewals from the previous workflow process to the updated renewal workflow process in PolicyCenter 4.0.0.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

#### Database upgrade not supported (CC-56175)

- **Issue:** Database upgrade from previous versions of PolicyCenter is not supported in PolicyCenter 4.0.0.
- **Workaround:** Guidewire is aware of this issue. Database upgrade will be supported in a future release.

## Changes in This Release

This section describes the product changes in this release.

- Base PCF File Changes
- Configuration Changes

## Base PCF File Changes

All links below require the `readme_files` directory on your local disk.

### PolicyCenter release 3.0.3 to 4.0.0

- To view a report of the changes in the base PCF files in the `modules/pc` directory, [click here](#).
- To view a report of the changes in the base PCF files in the `modules/p1` directory, [click here](#).

## Configuration Changes

For a list of configuration and GScript API changes between PolicyCenter 3.0.3 and PolicyCenter 4.0.0, [click here](#) (requires the `readme_files` directory on your local disk). This is a Microsoft Excel file that details changes in the following:

- Entity model
- GScript API
- Typelists
- Display keys

## 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 if they 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 login remains active in new browser window (CC-4152)

- **Issue:** When you log into PolicyCenter in a browser window, then you close that window and open a new one, in some instances you can access PolicyCenter pages without needing to log in again.

This is an issue with some versions of Microsoft Internet Explorer. When multiple browser windows are running in the same process, they can share session information, and this allows them to share a single login with PolicyCenter. Early versions of Internet Explorer provided a **Launch browser windows in a separate process** option, which, if enabled, would avoid this situation. Some versions of Internet Explorer do not allow you to set this option, and enable or disable it automatically. You can review the relevant Microsoft Knowledge Base Article 240928.

However, you may experience a situation in which Internet Explorer does not correctly enable this option, especially if you manually disabled it in an earlier version and then later upgraded. In this case, the multiple browser windows are run in the same process, exhibiting the undesired behavior.

- **Workaround:** The simplest solution is to always click **Logout** at the top of the PolicyCenter window before closing your browser.

If desired, you can also take steps to manually enable the Internet Explorer option. This requires you to edit the Windows registry on the affected computer.

**Warning:** Editing the registry is a risky procedure, and any improper changes may result in permanent data loss and require Microsoft Windows to be completely reinstalled on that computer. To make this change, create or edit the `[HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\BrowseNewProcess]` registry entry. Set the data type to `REG_SZ`, and then set the data value to `yes`.

### WebSphere timeout during initial startup (CC-4904)

- **Issue:** The initial creation of the PolicyCenter database may take longer than the default WebSphere startup timeouts will allow, and WebSphere may abort the startup before the database upgrade has completed. In this

situation, the `Default_Server_stderr.log` file will show an interruption in the `Applying upgrade step` `### of ####` log messages.

- **Workaround:** Increase the Ping Timeout and Ping Initial Timeout values for WebSphere. These can be found in the WSADMIN Console; click **Nodes**, drill down to your application server, click the **Advanced** tab, and increase the values.

#### On AIX without an X11 server, PolicyCenter may report that an X11 resource is missing (CC-5889)

- **Issue:** PolicyCenter does not require an X11 server; however other components of the PolicyCenter environment (such as Java) may inadvertently have this requirement. If so, PolicyCenter may report that an X11 resource is missing when it is starting up.
- **Workaround:** Install an X11 server, or contact Guidewire support to determine if the dependency can be removed.

#### Port conflict when starting PolicyCenter with JMX RMI adapter enabled and Microsoft Outlook running (CC-6964)

- **Issue:** When Microsoft Outlook starts, it chooses several ports to use for its connection to an Exchange server. In some instances, Outlook may use port 1099, which is a port that the PolicyCenter JMX RMI adapter can use, *if enabled*. The PolicyCenter server will then report a port conflict during startup when it tries to use that port for the RMI adapter.
- **Workaround:** Close Outlook to free the port, and then start the PolicyCenter server again. You can also change the port that the RMI adapter uses by editing the `JMXManagementPlugin` through Guidewire's Studio.

#### Studio: "Verify Changed Resources" is not implemented in the product model (CC-42574)

- **Issue:** The Studio Tools → Verify → Verify Changed Resources command does not work with the PolicyCenter product model.
- **Workaround:** Use one of the other verification commands instead. See "Validating Studio Resources" in the *PolicyCenter Configuration Guide* for details.

#### Unused display keys (CC-51235)

- **Issue:** PolicyCenter has display keys that are not used. If you are localizing PolicyCenter to other languages, you do not need to translate these unused display keys.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

#### Issues with producer permissions when using contact search (CC-53399)

- **Issue:** There are known issues with producer permissions when accessing accounts or policies associated with a contact from the `Search → Contacts` drop-down list:
  - Producers who do not have permissions to access specific accounts should see the account but not be able to click on the account when viewing a contact from the `Contacts` search tab.
  - Producers who do not have permissions to access specific policies should not see the policy when viewing a Contact from the `Contacts` search tab.
  - The availability of the account or policy link should be based on the producer code permission.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

#### Personal Auto: Exception thrown with future dated garaging location change (CC-53716)

- **Issue:** PolicyCenter throws an exception when you have a future dated garaging location change in the following situation. In a personal auto policy change job, add a new garaging location and set this as the garaging location for all vehicles on the policy. Then enter an out-of-sequence policy change adding a new vehicle and associating it with the original garaging location. PolicyCenter throws an exception when you quote the policy change. This is similar to CC-55474 which is for business auto.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Duplicate policy contact roles in preempted policy change jobs block quoting (CC-54278)**

- **Issue:** On preempted policy change jobs, you can get duplicate policy contact roles. Although you click **Handle Preemptions**, and **Apply all changes**, the duplicate policy contact role still exists. This results in a validation error and blocks quoting.

In some cases, the screen displays both contact roles, and you can delete one of them. In other cases, the screen displays only one contact role, but you get the validation error when you click **Quote**.

- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Commercial Package Policy: Stack trace when removing a line of business in a future dated policy change (CC-54329)**

- **Issue:** In a commercial package policy, removing a line of business in a future dated policy change, then doing an out-of-sequence change, results in a stack trace. This does not stop you from binding the policy change.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Commercial Package Policy: Exception when quoting a preempted policy change job (CC-54333)**

- **Issue:** In a commercial package policy, quoting a preempted policy change job results in an exception. In a submission job, create a policy that contains all lines of business. Quote and issue the submission. start a policy change job. Remove a line and make some other changes. Quote the policy change. Start another policy change job with the same effective date. Do not remove the line but make the same changes. Quote the policy change. Bind the submission job (which has removed commercial property line) In the second policy change job, handle preemptions and apply all changes. When you try to quote, PolicyCenter throws an exception.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Inland Marine Policy: Exception when adding a coverage part in future and out-of-sequence policy changes (CC-54338)**

- **Issue:** In an inland marine policy, adding coverage parts in both future and out-of-sequence policy changes causes an exception. Quote and bind both policy changes. PolicyCenter throws an exception when starting a third policy change.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Commercial Package Policy: Exception when adding line of business in future and out-of-sequence policy change (CC-54372)**

- **Issue:** In a commercial package policy, if you add a line of business in a future policy change, then add the same line of business in an out-of-sequence policy change, PolicyCenter throws an exception when you quote the second policy change.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Workers' Compensation: Incorrect premium if same jurisdiction added to preempted policy change job (CC-54937)**

- **Issue:** In a workers' compensation policy, PolicyCenter returns an incorrect premium if the same jurisdiction is added to both the preempting and preempted policy change jobs.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Concurrent modifications to underwriter issue approvals do not generate an error (CC-54944)**

- **Issue:** Under some circumstances, concurrent modification of the approval for an underwriter issue by two users on two different workstations will not create a system error, as expected.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Participants added multiple times to the policy (CC-55203)**

- **Issue:** The Participants screen displays the participants from more than one job. For example, this occurs in the following situation. Quote and bind a submission (using any line of business). Then start a policy change and bind the policy change. The Participants screen displays the submission participants as well as the policy change participants.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Workers' Compensation: Exception when policy change job with split on basis amount preempts renewal job (CC-55303)**

- **Issue:** In a workers' compensation policy, PolicyCenter throws an exception if a renewal job is preempted by a policy change job that has a split on basis amount based on anniversary date.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Non-deterministic validation errors on server startup (CC-55314)**

- **Issue:** When starting up the server, you may receive non-deterministic validation errors caused by the UWIssueValueType Gosu class.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release. The solution will likely remove your access to UWIssueValueType and change the constructor parameters for UWIssueValueComparatorWrapper.

**Workers' Compensation: Exception if jurisdictions removed from an earlier period (CC-55416)**

- **Issue:** In a workers' compensation policy, PolicyCenter throws an exception when quoting a renewal job if jurisdictions were removed from an earlier period. This occurs in a policy change that preempts a renewal job and in an out-of-sequence policy change.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Exception thrown with future dated garaging location change in a business auto policy (CC-55474)**

- **Issue:** PolicyCenter throws an exception when you have a future dated garaging location change in the following situation. In a business auto policy change job, add a new garaging location and set this as the garaging location for all vehicles on the policy. Then enter an out-of-sequence policy change adding a new vehicle and associating it with the original garaging location. PolicyCenter throws an exception when you quote the policy change. This is similar to CC-53716, which is for personal auto.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Removing location at a future date results in null constraint violations in a commercial property policy (CC-55475)**

- **Issue:** In commercial property policy with out-of-sequence policy changes, removing a location at a future date results in null constraint violations at quote.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Editing underwriter issues in slice mode can result in invalid policy data (CC-55501)**

- **Issue:** Editing underwriter issues in slice mode can introduce invalid data into the policy.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release. Until the issue is addressed, avoid editing underwriter issues in slice mode.

**PolicyCenter displays many warnings when regenerating the toolkit for regenerating Javadoc (CC-55564)**

- **Issue:** This issue occurs when running the regen-toolkit command.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Multiple DiffProperties create for a single change (CC-55596)**

- **Issue:** In some corner cases, the PolicyPeriodDiffPlugin creates multiple DiffProperties for a single change when it should only create one.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Overlapping coverages in out-of-sequence slices (CC-55658)**

- **Issue:** In a rewrite job, setting the effective date beyond the effective date of the last out-of-sequence slice creates overlapping coverages of the same pattern.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Missing required coverages added in a business auto policy (CC-55675)**

- **Issue:** In business auto, the quote process adds missing required coverages instead of reporting them as errors.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Hard-coded strings (CC-55718)**

- **Issue:** There are some hard-coded strings that need to be changed into display keys for easier localization. The remaining hard-coded strings are primarily in Gosu code.
- **Workaround:** In some cases, the hard-coded string is in Gosu code that you can modify. If this is the case, you can modify the code to use a display key rather than the hard-coded string. Guidewire is aware of this issue and will address it in a future release.

**Exception thrown when trying to view slice date with out-of-sequence policy changes (CC-55760)**

- **Issue:** Exception thrown when trying to view the slice date in a policy with out-of-sequence policy changes and underwriting issues. The exception is: Unhandled exception during RENDER.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Issue with submission state when underwriter approves an issue (CC-55770)**

- **Issue:** If a user approves an underwriting issue without editing or locking the submission, the submission is in a state where it is editable by the producer, but the quote is hidden.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Unable to quote policy change because pre-qualification question blocks quote (CC-55863)**

- **Issue:** If you have a pre-qualification question set with a required question that blocks quote, you will get an “Answer is missing for required question...” exception when you try to quote the policy change.
- **Workaround:** In Studio, for all questions sets of type PreQual, set JobType to Submission on the Availability tab. Guidewire is aware of this issue and will address it in a future release.

**Differences screen not implemented in some lines of business (CC-56176)**

- **Issue:** The Differences screen, which displays differences between a policy change and the original policy, or between different versions on a work order, is not yet implemented in the workers' compensation or business-owners lines of business.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Duplicate contact when merging contacts through API (CC-56177)**

- **Issue:** When you merge two instances of a contact through the API, you may get duplicate contacts for the following contact roles:
  - Driver
  - Secondary Named Insured
  - Primary Named Insured

- Billing Contact
- Labor Contact
- Named Insured
- Additional Interest
- Additional Insured
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

#### **Classes missing from Java API Javadoc (CC-56185)**

- **Issue:** In the Java API Javadoc, some of the classes in the `com.guidewire.pl.plugin.*` packages are missing.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

#### **New configuration parameter to synchronize contacts (PL-1321)**

- **Issue:** There is a new configuration parameter, `ShouldSyncUserRolesInLDAP`, in the `config.xml` file. If its value is `true`, the application synchronizes contacts with the roles they belong to after authenticating with the external authentication source.
- **Workaround:** None required.

#### **Incomplete implementation of non-English locale-specific database search and sort (PL-8108)**

- **Issue:** Guidewire has begun implementation of database searching and sorting appropriate to non-English locales. This release of PolicyCenter contains an incomplete implementation of this work. The changes will not affect customers implementing case-insensitive searches using standard English sort criteria.  
As part of this work, the current release renames the `supportsCaseInsensitiveSearch` attribute on column to `supportsLinguisticSearch`.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

#### **Zone data with missing values cannot be loaded (PL-9123)**

- **Issue:** Zone data, which is used for associating postal codes with cities and regions, does not load if it is incomplete.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

#### **Errors from startable Gosu plugins (PL-9313)**

- **Issue:** This release includes a new feature called startable plugins, which let you run custom code at server startup. Your implementation of `IStartablePlugin` can spawn a listener, such as for pulling messages from a JMS queue. Some startable plugins written in Gosu throw exceptions if they expose a listener. This issue does not affect Gosu implementations of other plugin interfaces.
- **Workaround:** Implement the `IStartablePlugin` plugin interface and listener code in Java. Guidewire is aware of this issue and will address it in a future release.

# Guidewire PolicyCenter 4.0.1 Release Notes

## Release 4.0.1.14

Copyright © 2001-2010 Guidewire Software, Inc. All rights reserved. Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Insurance Suite, Guidewire ContactCenter, Guidewire Studio, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries. All other product names are trademarks of their respective companies.

**This documentation is published as Guidewire Confidential.** The contents of this documentation, including product architecture details and APIs, are considered confidential and are fully protected by customer licensing confidentiality agreements and signed Non-Disclosure Agreements (NDAs).

This product includes software developed by the Apache Software Foundation (<http://www.apache.org>).

## Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Changes in This Release
- Known Issues and Limitations

## Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

## Version Number

This release of Guidewire PolicyCenter is 4.0.1.14.

## Installation Requirements

For installation requirements, see the *PolicyCenter Installation Guide*.

## Support

This document includes important information about the current release of Guidewire PolicyCenter. For assistance with this software release, contact Guidewire Customer Support. You can contact support on the Web at <http://portal.guidewire.com>, by email at [support@guidewire.com](mailto:support@guidewire.com), or by phone at +1-650-356-4955.

## Installing This Release

For installation information, see the *PolicyCenter Installation Guide*.

This section describes the following changes to PolicyCenter:

- Software Updates
- InetSoft Evaluation
- Upgrade Issues

## Software Updates

The JBoss EAP 5.0 application server is now supported.

## InetSoft Evaluation

You can evaluate InetSoft Style Report Enterprise Edition Version 10.1 using the following license key:

L000-798-ERX-0000BF00100001F-F2AFC9DD34ED

If you want to use InetSoft and the PolicyCenter reporting module in a production environment, contact Guidewire Customer Support to request a full license key. For more information, see the *PolicyCenter Reporting Guide*.

## Upgrade Issues

This section describes changes to the PolicyCenter base configuration that may cause upgrade issues.

### Message safe-ordering at the account level (PC-10760)

- **Issue:** In previous versions, PolicyCenter retrieved messages from the send queue in safe order by policy. After PolicyCenter got the message acknowledgement for a message, it sent the next message for that policy for that destination. In PolicyCenter 4.0.1, messaging safely orders messages by account (including any of its policies) rather than by individual policies. The *PolicyCenter Integration Guide* does not yet reflect this change. Prior to upgrading, you must ensure that there are no in-progress messages, such as pending, failed, sent but not acknowledged, or other such statuses.
- **Workaround:** Prior to upgrading, you must ensure that there are no in-progress messages. The Message table must be empty.

### Workflows removed (CC-53856)

- **Issue:** The IssueRenewalWF, PendingNonRenewalWF, PendingNotTakenWF, and PendingRenewalWF workflows were removed. There is no configuration upgrade step available at this time. There is also no mechanism to

migrate in-process renewals from the previous workflow process to the updated renewal workflow process in PolicyCenter 4.0.0.

- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

#### Database upgrade not supported (CC-56175)

- **Issue:** Database upgrade from previous versions of PolicyCenter is not supported in PolicyCenter 4.0.1.
- **Workaround:** Guidewire is aware of this issue. Database upgrade from PolicyCenter 3.0.x to 4.0.x will be supported in a future release. Database upgrade from PolicyCenter 4.0.0 to 4.0.1 is supported.

## Changes in This Release

This section describes the product changes in this release.

- Base PCF File Changes
- Configuration Changes

### Base PCF File Changes

All links below require the `readme_files` directory on your local disk.

#### PolicyCenter release 4.0.0 to 4.0.1

- To view a report of the changes in the base PCF files in the `modules/pc` directory, [click here](#).
- To view a report of the changes in the base PCF files in the `modules/p1` directory, [click here](#).

### Configuration Changes

For a list of configuration and GScript API changes between PolicyCenter 4.0.0 and PolicyCenter 4.0.1, [click here](#) (requires the `readme_files` directory on your local disk). This is a Microsoft Excel file that details changes in the following:

- Entity model
- GScript API
- Typelists
- Display keys

## Improvements and General Issues

This topic describes the primary improvements and issues corrected in this release.

ID	Description
<b>Account</b>	
CC-47001	<p>Improved minimum search criteria error messages. The minimum account search criteria now shows more accurate error messages such as:</p> <p><i>Please specify: At least the first four characters in the last name.</i></p> <p>or</p> <p><i>Please specify: At least the first two characters in the last name and either the first name, address, city, or state.</i></p>
CC-49119	Fixed an issue where the ProductFilter on the Work Orders screen displayed the product description instead of the product name. Fixed by changing <code>gw/job/JobFilters.gs</code> .

<b>ID</b>	<b>Description</b>
CC-49131	Improved the filters on the Account File Work Orders screen. Previously, some filters did not have a label. The screen now has the following filters: Status, Work Order Type, and Product.
CC-53612	The Policies listview on the Account File Summary screen has been relabeled Policy Terms to more accurately describe the rows in the listview.
<b>Activities</b>	
CC-55763	Fixed an issue where PolicyCenter created a blank note when you requested or approved an underwriting issue. This occurred if you completed the activity without editing the note.
<b>Administration</b>	
CC-56854	Fixed an issue where an external user could not create a new group because they could not enter a security zone. (The security zone field is required, but not editable.) Now when an external user creates a new group, the security zone field is set by default to the security zone of the root group of the organization.
CC-57041	Added a Holiday link in left sidebar of the Administration tab. The Add Holiday screen allows you to specify holidays by name, date, and type.
<b>Command line tools</b>	
CC-56035	Fixed a number of problems in the administrative tools, including: <ul style="list-style-type: none"> <li>• problems running <code>maintenance_tools.bat</code> to start, stop, or get the status of a batch job.</li> <li>• references to resyncing claims in <code>messageing_tools.bat</code>.</li> <li>• <code>template_tools.bat</code> did not list its options from the command line.</li> </ul>
<b>Configuration of the product</b>	
CC-52522	Fixed capitalization in QuickJump classes (the <i>J</i> should be capitalized). The following instances of Quickjump were changed to QuickJump: <ul style="list-style-type: none"> <li>• Quickjump</li> <li>• Quickjump.AccountContextual</li> <li>• Quickjump.PolicyContextual</li> <li>• Quickjump.Static</li> <li>• Quickjump.AutoCompleteHandler</li> </ul>
<b>Contacts</b>	
CC-50278	Fixed an issue with contacts. Previously, you could add policy additional interests with the same type and contract number. Now, PolicyCenter displays a validation error message if the additional interest contact has the same interest type and contract number.
CC-53229	Fixed an issue that occurred when two driver contacts were merged. The driver contact was duplicated in policies using them.
CC-53399	Fixed issues with producer permissions from the Search Contact screen. The new behavior when accessing specific accounts or policies associated with a contact is: <ul style="list-style-type: none"> <li>• Producers who do not have permissions to access specific accounts can see them but cannot click on them when viewing a specific contact from the Search Contact screen.</li> <li>• Producers who do not have permissions to access specific policies cannot see them when viewing a specific contact from the Search Contact screen.</li> <li>• Based on producer code permission, the account or policy link is available.</li> </ul>
CC-54140	Validation at quote and bind now prevents a job from quoting or issuing if there are inactive contacts on the policy.
CC-55731	Previously, the Remove Contact button was available whether or not the contact was in use. Now this button is available only when the contact is not in use.
CC-56199	Fixed an issue in the default configuration where searching for contacts of type Person always returned the contact <i>Stone Cold</i> . Similarly searching for Company always returned <i>BBB Cars &amp; Trucks</i> . These contacts were external contacts returned by <code>DemoContactSystemPlugin</code> . <code>IContactSystemPlugin.xml</code> now uses <code>StandAloneContactSystemPlugin</code> instead of <code>DemoContactSystemPlugin</code> .

ID	Description
CC-56305	<p>Fixed an issue where the number of AccountSyncable rows proliferated with each completed job (for example, each promoted policy period slice). In the default configuration, the AccountSyncable entities are PolicyContactRole, PolicyLocation, and PolicyAddress.</p> <p>The fix replaced the AccountSyncable.SyncedToAccount column with a Gosu property that calls through to the IAccountSyncable.isSyncedToAccount(AccountSyncable) method to determine if the entity is synced to the account. The default implementation of IAccountSyncable considers all AccountSyncable entities to be synced if their PolicyPeriod is in the Temporary, New, or Draft state. It also is considered synced if the status is null (in the moments when the period is being constructed). PolicyBillingContact, additionally, remains synced when the period is Quoting or Quoted.</p>
CC-56644	<p>Fixed an issue where the AdditionalNamedInsureds DV panel on PolicyInfo tests for the PolicyLine but should look at the product. Previously, on the submission wizard Policy Info screen, the Additional Insured DV panel was visible when the Personal Auto line did not exist. Now this panel is visible when the Product is not Personal Auto.</p>
CC-56694	<p>Previously, when a new job started off of a bound period where policy contact roles could not be merged, the roles were not automatically merged in the new job. The roles are now automatically merged.</p>
PC-7799	<p>Fixed an issue on preempted policy change jobs where PolicyCenter added duplicate policy contact roles when a contact was added in the same role in both the preempting and preempted policy change jobs. Although you clicked Handle Preemptions, and Apply all changes, the duplicate policy contact role still existed. This resulted in a validation error and blocked quoting.</p> <p>In some cases, the screen displayed both contact roles, and you could delete one of them. In other cases, the screen displayed only one contact role, but you received the validation error when you clicked Quote.</p>
<b>Core</b>	
PL-8199	Guidewire has modified the behavior of SystemPermissionsType.getArrayType() so that it now returns a new DefaultArrayType. Code with an unclosed array bracket (for example, perm.System[]) will no longer cause a NullPointerException.
PL-8324	<p>Guidewire has added the ability to specify system properties by passing them as Ant system properties. For example, to set http.proxyUser (if you are starting a Jetty server), enter:</p> <pre>-Dhttp.proxyUser=myuser</pre> <p>However, do not pass Xms, Xmx, or -XX:MaxPermSize this way as the values specified in memory.properties overrides these values.</p>
PL-8332	<p>There is a new way to expose arbitrary types to Java external (entity) libraries. Add a line for that type in the new configuration file: PolicyCenter/modules/configuration/config/plugin/ExternalTypes.1st. The format of the file is simply one fully-qualified type (with the package as it exists in the Gosu type system) on each line. For more information, see the "Entity Libraries" chapter in the <i>Integration Guide</i>. This works for all Gosu types, as well as Java types in the gw.* namespace (such as gw.api.util.CurrencyUtil).</p>
PL-8457	Guidewire has added a Download button to the Server Tools → Archive Graph Info page. This enables you download a ZIP file containing DOT-formatted files for the admin and domain graphs.
PL-8490	Corrected an issue involving archiving claims with related contacts.
PL-8607	Archiving now skips claims that have active work items.
PL-8761	Corrected an issue with query generation that caused Oracle <i>SQLException: identifier is too long</i> . The exception occurred if the query contained an alias longer than the 30 characters allowed by the Oracle database.
PL-8944	Guidewire now checks the ArchiveEnabled configuration parameter at server start up to determine whether to trigger an upgrade. A change from false to true triggers an upgrade. If you attempt to change the value of this parameter from true to false, the server refuses to start.
PL-9457	Updated the joda-time library from version 1.4 to 1.6 to correct a problem with the IBM JDK with WebSphere that returned incorrect results for class DateMidnight.
PL-9689	Corrected an issue that occurred when running maintenance_tools -markclaimsforpurge against a claim with a large number of checks (4000). This caused an SQLServerException.
PL-9821	Corrected an issue in which the localized field validators throw exceptions for null bean object

ID	Description
PL-9857	Guidewire now provides four modes for debugging when starting the development server during configuration. You can change the debug port to use by configuring the <code>port.properties</code> file. The modes are: <ul style="list-style-type: none"> <li>• <code>dev-debug-shmem</code> (Debug dev-mode webapp using shmem)</li> <li>• <code>dev-debug-socket</code> (Debug dev-mode webapp using socket)</li> <li>• <code>dev-suspend-shmem</code> (Debug dev-mode webapp using shmem, start suspended)</li> <li>• <code>dev-suspend-socket</code> (Debug dev-mode webapp using socket, start suspended)</li> </ul>
<b>Database upgrader</b>	
CC-57185	The <code>Partition</code> column on the <code>Extractable</code> delegate was renamed <code>ArchivePartition</code> .
<b>Diffs in the user interface and integration</b>	
CC-46043	Fixed an issue in policy change jobs where changes to official ids were not reflected in the <b>Differences</b> tab.
CC-53733	Fixed an issue where window mode changes did not appear in the <b>Policy Review</b> screen as differences.
CC-54927	When configuring differences for Bean nodes, the <code>parentexpression</code> , you can now reference BEAN instead of VALUE. BEAN represents the owning bean of the properties being included by the diff element.
CC-54960	In the differences display, the section sort order (the <code>sortOrder</code> field) is now respected.
CC-55006	Improved display of differences between policies after a policy rewrite. For example, there are fewer unnecessary add and remove lines for unchanged attributes.
CC-55238	Change default display for dates in the difference user interface to the short date format. You can override the format using the <code>Label</code> property in the difference tree configuration.
CC-56220	For difference display, some logic was moved from <code>DiffHelper</code> and <code>LOBDiffHelper</code> subclasses to the <code>DiffTree</code> configuration files in the <code>&lt;ExcludeProperty&gt;</code> element.
CC-56292	On the <b>Differences</b> page, the label for comparing transactions changed from <b>Differences Between Policy Periods</b> to <b>Differences Between Policy Transactions</b> .
CC-56529	Improved labels related to differences in <code>PolicySearchDV.pcf</code> .
CC-56932	PolicyCenter 4.0.0 included a new system for displaying differences using a tree structure, and XML configuration of what to display in the tree. The <code>DiffItemDisplay</code> class and its subclasses did a similar task in PolicyCenter 3.0. The <code>DiffItemDisplay</code> class remained for some products in PolicyCenter 4.0.0 that did not yet use the new tree difference system. In PolicyCenter 4.0.1, all products now use the difference trees. Also, differences shown for preempted jobs and out-of-sequence conflicts display differences using the difference trees. Consequently, the <code>DiffItemDisplay</code> class and its subclasses are removed from the product.
<b>Document management subsystem</b>	
CC-45018	Fixed an issue with document that were unhidden not appearing. Documents that are just unhidden now appear when the user looks for only unhidden documents.
CC-49813	Fixed an issue where a localized document template descriptor was not being used properly.
<b>Documentation</b>	
CC-55370	Corrected an issue in which Guidewire Standard Reporting displayed the local client currency (meaning the default currency of the operating system) instead of the actual transaction and reporting currency. You now set the default currency symbol for InetSoft through the <code>sree.properties</code> file. You use the following properties to set the default currency: <code>language</code> and <code>country</code> . In the base application configuration, Guidewire sets the default values for these two properties to <code>en</code> and <code>US</code> (English, US), which displays the \$ sign for the currency symbol. To show the British Pound, set these properties to <code>en GB</code> (English, Great Britain). For the Euro, and country that uses the Euro, you can use the following values, for example: <code>fr FR</code> (French, France) or <code>de DE</code> (German, Germany).
CC-56868	Added the following recommendations to the <i>Configuration Guide</i> : <p><b>Maximum Typelist Size:</b> Guidewire strongly recommends that you limit the maximum number of typecodes in a typelist to 250 items. Any number larger than that can cause performance issues. If you need more typecodes than the 250 limit, then use a lookup (reference) table and a query to generate the typelist. In any case, Guidewire does not support the use of more than 8000 typecodes on a typelist.</p> <p><b>WARNING:</b> Guidewire imposes an upper limit of 8000 typecodes on a typelist. If you attempt to create a typelist with a larger number of typecodes, then Guidewire PolicyCenter refuses to compile.</p>
PL-5264	Guidewire now exposes the <i>PCF Format Reference</i> on the Studio Help menu.
<b>General application bug</b>	

ID	Description
CC-46660	Fixed an issue with the left sidebar in the <b>New Submission</b> screen. In some cases, the left sidebar contained links from the <b>Desktop</b> tab or displayed links to submission screens which did nothing. The left sidebar on this screen is now empty.
CC-46662	Fixed an issue with the left sidebar when going from the <b>New Submission</b> screen to the <b>Enter Account Information</b> screen. In some cases, the left sidebar contained links. The left sidebar on this screen is now empty.
CC-50611	Fixed an issue that occurred when two users attempt to change the same data at the same time. Previously, the user who <i>lost</i> would cancel the change by clicking on the <b>Desktop</b> , and cleaning the <b>Unsaved Work</b> queue. Now, the user who <i>loses</i> can click a <b>Discard Changes</b> link.
CC-51235	Removed unused display keys.
CC-52259	In some cases, the wizard step displayed a different label than the corresponding screen in the Policy File. For example, Inland Marine displayed <b>Buildings and Locations</b> in the wizard step, but just <b>Buildings</b> in the Policy File. Now the wizard steps match the Policy File labels.
CC-56298	On the account version of the <b>Participants</b> screen, the role column was labeled <b>Participant</b> . This has been renamed <b>Role</b> to be consistent with the policy version of the <b>Participants</b> screen.
<b>General policy</b>	
CC-56273	Fixed an issue where <b>Copy Submission</b> copied the total cost from the original submission. The <b>Submission</b> enhancement now nulls or zeroes individual columns on the policy period, including bill method items and denormalized total fields.
<b>Gosu</b>	
PL-9275	Studio now generates deprecation reason messages for <b>MetaType</b> properties exposed from <b>TypeLiteral</b> expressions.
PL-9692	Corrected an issue that caused <b>AuthenticationSourceCreatorPlugin</b> implemented in Gosu to fail authentication if used with WebSphere 7.
<b>Integration</b>	
PL-4337	Guidewire has modified the behavior of <b>OrganizationSearchCriteria</b> . It now supports search by the contact's country and county, as well as the contact's city, state and postal code.
PL-6982	Guidewire has removed <b>ClaimWorkflow</b> as a choice on the <b>Workflow</b> type drop-down in the <b>Workflow Statistics</b> screen available to administrators. <b>ClaimWorkflow</b> is actually the abstract supertype for <b>MetroWorkflow</b> .
PL-7714	Guidewire has modified the behavior of the <b>GuidewireDocumentAssistant</b> ActiveX control to provide better security.
PL-7715	Guidewire has modified the behavior of the <b>GuidewireDocumentAssistant</b> ActiveX control to provide better security.
PL-7716	Guidewire has modified the behavior of the <b>GuidewireDocumentAssistant</b> ActiveX control to provide better security.
<b>Integration (between applications)</b>	

ID	Description
CC-55149	<p>PolicyCenter ExitPoints that go to ClaimCenter and BillingCenter now use configuration parameters for URLs. The configuration parameter, <code>appSystemURL</code>, provides the base URL of the other application. For example, this configuration parameter can be <code>ClaimSystemURL</code>, <code>PolicySystemURL</code>, or <code>BillingSystemURL</code>.</p> <p>Using this configuration parameter, the exit points between the applications might be defined as follows:</p> <pre data-bbox="437 361 1454 1189">&lt;ExitPoint     id="ViewPolicy"     popup="true"     url="{policySystemURL}/PolicyEntryPoint.do"&gt;     &lt;LocationEntryPoint         signature="ViewPolicy(policySystemURL : String, urlParam1: String, urlParam2: DateTime)"/&gt;     &lt;ExitPointParameter         locationParam="policySystemURL"         type="String"         urlParam="policySystemURL"/&gt;     &lt;ExitPointParameter         locationParam="urlParam1"         type="String"         urlParam="urlParam1"/&gt;     &lt;ExitPointParameter         locationParam="urlParam2"         type="DateTime" urlParam="urlParam2"/&gt;</pre> <p>The button to go to the exit point might be defined as follows:</p> <pre data-bbox="437 910 1323 1121">&lt;ToolbarButton     action="ViewPolicy.push(gw.api.system.CCConfigParameters.PolicySystemURL.Value,     Claim.Policy.PolicyNumber, Claim.LossDate)"     icon="icon_newWindow.gif"     id="ViewInPC"     label="displaykey.Button.ViewInPC"     visible="gw.api.system.CCConfigParameters.PolicySystemURL.Value.HasContent and     perm.System.viewpolicysystem"/&gt;</pre> <p>The parameter in config.xml is defined as follows:</p> <pre data-bbox="437 1153 894 1178">&lt;param name="PolicySystemURL" value="" /&gt;</pre>

**Integration with BillingCenter**

CC-53512	The <code>initBillingInformation</code> method has been moved out of the payment page PCF file. It is now a method in the <code>JobProcess</code> class and can be called from other code.
CC-55007	Billing instruction messages have been made retryable.
CC-55992	Fixed an issue with the BillingCenter integration where a submission required that BillingCenter also be started.

**Integration with ClaimCenter**

CC-53465	Fixed an issue in the integration between PolicyCenter and ClaimCenter. The <code>View Claim in claim system</code> link is now disabled if the claim status is <code>Archived</code> .
CC-56501	Fixed an issue where an incorrect period effective date was returned to ClaimCenter on policy search and retrieval.

**Integration with contact management system**

CC-55228	Fixed an issue where searching for contacts invokes <code>IContactSystemPlugin.searchContact(ContactSearchCriteria)</code> regardless of whether or not the plugin is enabled.
----------	--

**Internationalization**

CC-55718	Changed a number of hard-coded strings in Gosu to typekeys for easier internationalization.
----------	---

**Job flow**

ID	Description
CC-55911	<p>Removed duplication in JobProcess code by making the subclasses as similar as possible, and by moving shared code into the parent JobProcess class. There are two sets of changes which work together.</p> <p>The first set gives us a generic way of checking system permissions for different job processes. For example, we can check to see if the current user has permission to request that job process do a bind by checking the property: <code>jobProcess.Permissions.Bind</code>.</p> <p>The second set of changes collects lists of low-level checks that are commonly performed at well-known points of the JobProcess workflow and makes them available as methods on JobProcess. Specific JobProcess subclasses call these methods directly at the appropriate points in their processing. If a particular type of JobProcess needs to make a specialized set of checks at some point in its workflow, it overrides the appropriate method from JobProcess and can then either extend or replace the set of checks normally made at that point. So, JobProcess tells you what needs to be checked and when, for processes to proceed normally through a workflow. The subclasses tell you what special rules and exceptions apply, and suggest the kinds of low-level checks that any new process type should make as it moves a PolicyPeriod through the workflow.</p>
CC-55952	Added missing keyboard shortcuts, including Back and Next, for wizard buttons.
<b>Job: cancellation process</b>	
CC-55782	Provided a more accurate warning message when scheduling two cancellations. Although the user could schedule the second cancellation, the user received a message that they must rescind the cancellation before proceeding. The message now advises them to rescind before proceeding.
CC-56293	When a policy is scheduled for cancellation, PolicyCenter now displays a warning in the policy file. This is useful because there are no other indicators on the policy file that it is scheduled for cancellation (the policy status is <b>In Force</b> , as it should be).
<b>Job: premium reporting</b>	
CC-56631	Fixed an issue where cost dates outside of premium report start and end dates created negative transaction amounts. Now PolicyCenter prevents costs in premium reporting branches from extending outside of the period of time covered by the report. Since PolicyCenter also prevents overlapping reporting periods, this prevents costs from overlapping in the way that caused the issue.
<b>Job: rewrite process</b>	
CC-55658	Fixed an issue where full term rewrite was not considered out-of-sequence to a policy change. This could cause duplicate coverages.
<b>Job: renewal process</b>	
CC-55303	Fixed an issue where the user received an exception on a preempted renewal job if an earlier revision had a split on basis amount based on anniversary date.
CC-55416	Fixed an issue where the user received an exception when quoting a renewal job after removing jurisdictions from an earlier period.
CC-55612	Fixed an issue where users without <code>editlockoverride</code> permission could edit and quote locked renewals.
CC-56927	Fixed an issue where PolicyCenter was not calling the plugin that determines what the policy number should be for the renewal term of a policy.
<b>Job: submission process</b>	
CC-52408	Fixed an issue where a missing required field validation error message blocked the closing options <b>Not Taken</b> and <b>Decline</b> .
CC-56336	Removed the grayed out <b>Bind Options</b> button from the <b>Quick Quote</b> screen.
CC-56491	Fixed an issue with formatting money fields on billing screens for installment plans and <b>Preview Payments</b> . The money fields are now right-justified with two decimal places.
<b>LOB: business auto</b>	
CC-52146	In Business Auto, PolicyCenter now checks to see if the vehicle class codes match the fleet type. PolicyCenter displays a validation error if they do not match.
CC-53819	Fixed an issue where the user could not complete a quote when making out-of-sequence changes to coverage symbol patterns that had also been changed in a future dated change
CC-55675	Fixed an issue in business auto where missing required coverages were added when quoting. At quote, missing required coverages are now reported as errors, and not automatically added.
CC-56646	Fixed an issue where the user received a validation error message about auto coverage symbol patterns in an out-of-sequence policy change even though no auto symbols were modified.

<b>ID</b>	<b>Description</b>
CC-57190	Fixed an issue where the Covered Auto Symbols for the policy were not set unless you visited the Covered Vehicles screen.
<b>LOB: businessowners</b>	
CC-53206	Wizard steps indentation for Businessowners items is now consistent with other lines of business where wizard steps (wizard steps are not indented).
CC-55417	In businessowners policy, removed the following confirmation message which does not apply to this type of policy: <i>Removing this location will cause the expiration date for all risks at this location to be set to this date. Do you want to continue?</i>
CC-55734	Fixed an issue in general liability where the location appeared as <b>null</b> in conflicting changes worksheet after applying all changes from preempting job. The class code validation for businessowners and commercial property lines used the same logic, but implemented this in separate Gosu classes. To avoid duplication of code, these lines of business now use the abstract <code>BuildingClassCode</code> delegate to do class code validation. The class code logic was moved from <code>BOPBuildingValidation.gs</code> and <code>CPBuildingValidation.gs</code> .
CC-55898	Fixed an issue where a user could bind a policy without setting <b>Small Business Type</b> , which is required.
CC-56218	Fixed an issue where flat Costs had a null reference to Coverage in a policy change job where the corresponding coverage is removed.
CC-56486	Fixed an issue where the location questions card was not available for new locations until the user returned to the Locations page.
<b>LOB: commercial package policy</b>	
CC-54207	Fixed an issue in which two Signs links appeared in the links for Inland Marine on a CPP policy when the signs coverage part was added in both an OOS change and a future dated change.
<b>LOB: commercial property</b>	
	Additional commercial property issues: <ul style="list-style-type: none"><li>• CC-55734</li></ul>
<b>LOB: general liability</b>	
CC-54265	Fixed an issue in general liability. In some cases, the <i>from</i> and <i>to</i> dates for the jurisdiction were incorrect, and the estimated basis on the payroll was also incorrect.
CC-55678	Fixed an issue where the location appears as <b>null</b> in the Conflicting Changes worksheet after applying all changes from a preempting job.
<b>LOB: inland marine</b>	
CC-55736	Fixed an issue where IMBuilding autonumbering was incorrect after handling preemption.
CC-56801	Fixed an issue on the inland marine Policy Review and Quote pages where the label repeated the word <i>coverage</i> . This occurred in the labels for signs and scheduled equipment coverages.
<b>LOB: personal auto</b>	
CC-53533	In the PA Coverages screen in personal auto, check boxes in vehicle list view were removed. These check boxes had no functionality.
CC-55250	Fixed an issue in personal auto where the StandAloneBillingSummaryPlugin plug-in returned delinquency data. PolicyCenter displayed a <i>This policy is in delinquency</i> warning. The plugin no longer returns delinquency data in this case.
CC-55743	You can now view cost differences for personal auto when comparing two versions of a policy.
CC-55953	Fixed an issue where the PA Coverages screen did not display covterms as read-only when there was only one option.
CC-56507	Fixed an issue in the PALiability enhancement. The enhancement checks the limits on the UM/UIM covterm patterns against the PA Liability limits. If the UM/UIM limits exceed the corresponding Liability limit, then it returns false, meaning that the UM or UIM covterm is not available. The original version of the enhancement assumed that the Liability Package terms would be in the correct sequence, but that is not necessarily true. So the enhancement now finds the correct Liability package term based on the Aggregation Model and Restriction Model.

ID	Description
CC-57081	Fixed an issue in PALineValidation.gs. Previously the validation assumed that the package values were in sequence. The validation now compares the package values using the aggregation and restriction models on each package value.
CC-57377	Updated some values in the personal auto offerings to be set to values different from the base policy line value, rather than the same value. Updated the following:  In Basic Program: <ul style="list-style-type: none"><li>• Rental Reimbursement set to 30/15</li><li>• Uninsured Motorist - Property Damage Limit set to 10,000</li></ul> In Premium Program: <ul style="list-style-type: none"><li>• Rental Reimbursement set to 60/20</li><li>• Uninsured Motorist - Property Damage Limit set to 1,000,000</li></ul>
<b>LOB: workers compensation</b>	
CC-45829	Fixed an issue where the Exp Mod difference for the second period did not appear in the Policy Versions differences.
CC-45834	Fixed an issue with displaying differences for anniversary date and Exp Mod changes.
CC-54937	Fixed an issue where the user received an incorrect premium if they added the same jurisdiction to a preempting policy change as to the preempted policy change.
CC-56013	For Unsavd Work, added countsAsWork="false" to the SplitPeriodPopup.pcf called by the submission wizard.
CC-56208	Fixed an issue in workers' compensation where exposures were removed in the future without any warning.
CC-56286	In workers' compensation, a <b>Manuscript Option</b> has been added to the WC Options screen.
CC-56487	Fixed an issue in workers' compensation line of business. The <b>Location Questions</b> card was missing from the <b>Locations</b> screen.
CC-56621	Fixed an issue where the <b>WC Coverages</b> screen and <b>WC Options</b> screen covterms did not display as read-only when there was only one option or package available. Now, if there is only one option available, that value displays as a read-only value.
CC-57325	The following columns were removed from WorkersCompLine.eti and replaced with covterms on the WCOtherStatesInsurance coverage: <ul style="list-style-type: none"><li>• CustomStates</li><li>• IncludedMonopolyStates</li><li>• OtherStatesOpt</li><li>• StopGapOpt</li></ul>
<b>Location</b>	
CC-51392	Improved display of address in <b>Account File Locations</b> and <b>Policy Locations</b> screens. PolicyCenter displays the address in an address field for both account and locations.
CC-55651	Fixed an issue where the user received a validation error message to set the territory code for a location which should have appeared in the location list view but did not.
CC-56742	Removed usage of AddressOwner in PCF files. This has been replaced with Address.  Additional location issues: <ul style="list-style-type: none"><li>• CC-56305</li><li>• CC-55651</li></ul>
<b>Performance Internals</b>	
CC-55488	To improve search performance, an internal policy search class was removed and replaced by a more efficient mechanism.
CC-55657	Improved performance of submission job by removing repeated calls to the letters table.
CC-55672	Improved performance of menu containing AccountContact names so that contacts are now loaded only when necessary.
CC-56051	Improved performance by using subMenuOnDemand in the MenuItemIterator when possible.
CC-56602	You can no longer do an account search entering only the zip code field.
CC-56611	Improved performance of account search indexing.
CC-56624	Improved performance of <i>jobwizard_PolicyInfoDV</i> PCF files by reducing the number of queries for UWCompany.

ID	Description
CC-56767	Improved the performance of query on jobs with no locked periods on an account.
CC-56787	Improved the performance of account search when searching for the work phone number.
CC-56928	Improved performance of the AccountContactRoleSubtypes property query.
CC-57021	Improved performance of note search query by policy and security type, sorted by authoring date and create time.
CC-57131	Fixed a performance issue by reducing the number of queries to get future periods. To get the benefit of this fix, you must cache the results of PolicyPeriod.FutureRenewals. See StartRewriteMenuItemSet.pcf for an example.
CC-57149	Fixed a performance issue by reducing the number of queries on Job.Periods. To maximize this benefit, cache results of Job.Periods, Job.ActivePeriods, and Job.PolicyPeriod wherever possible in your code and PCF configuration.
CC-57367	Improved performance of the AccountFile_Contacts screen by caching the account contact role DisplayName for all account contacts.
<b>Policy file</b>	
CC-52806	<p>The Issue Date field in the user interface pointed to the same underlying field as Close Date.</p> <p>The following PCF files were changed:</p> <ul style="list-style-type: none"> <li>• Policy_Summary_AssocJobDV.default</li> <li>• Policy_Summary_AssocJobDV.Rewrite</li> </ul> <p>On the .default and .Rewrite ones, the Period Issued label and associated field were removed.</p> <ul style="list-style-type: none"> <li>• Policy_Summary_AssocJobDV.Cancellation</li> <li>• Policy_JobInfoDV</li> </ul> <p>On the .Cancellation PCF and Policy_JobInfoDV, the Issue Date label was changed to Closed.</p>
<b>Preemption</b>	
CC-53716	Fixed a class of problems where removing an object in a future dated change, then making changes in an out-of-sequence change that involved that object, resulted in errors. This also fixed a similar class of problems when a preempted change referenced an object removed in the preempting change.
<b>Product model configuration</b>	
CC-56022	<p>Added additional product model synchronization methods that allow you to synchronize at more granular levels. The following methods were added to ProductModelSyncIssuesHandler.gs:</p> <ul style="list-style-type: none"> <li>• syncCoverages(coverables : Coverable[], categories : CoverageCategory[], helper : JobWizardHelper)</li> <li>• syncExclusions(coverables : Coverable[], categories : CoverageCategory[], helper : JobWizardHelper)</li> <li>• syncConditions(coverables : Coverable[], categories : CoverageCategory[], helper : JobWizardHelper)</li> <li>• syncCoverages(coverages: Coverage[], helper : JobWizardHelper)</li> <li>• syncExclusions(exclusions: Exclusion[], helper : JobWizardHelper)</li> <li>• syncConditions(conditions: PolicyCondition[], helper : JobWizardHelper)</li> </ul>
<b>Rating engine</b>	
CC-53394	The AbstractRatingEngine provided the wrong rate for a mid-term policy change if there was only one EditEffectiveDate found. (This can occur in a policy change job with no actual changes.)
CC-55688	Improved Gosu code for handling taxes in personal auto. It is now easier to configure personal auto for handling multiple taxes. Changes were made in PersonalAutoTaxCostData.gs.
CC-56135	Fixed an issue that occurred when adding an override amount. The cost of the covered item was split into two slices. However, the item did not have the date and proration information.
CC-57159	Fixed a round-off error which occurred when the Cost entity had pennies and the Transaction entity was rounded to dollars.
<b>Reporting subsystem</b>	

ID	Description
CC-54316	There is a new property Job.SelectedVersion, which indicates for multi-revision jobs which branch is the selected (active) branch. This is mainly relevant where there is more than one non-withdrawn branch in the job. This new property is an edge foreign key from Job to PolicyPeriod. In the database, this creates a join table called pc_jobpolicyperiod where the OwnerID column points to the Job, and the ForeignEntityID points to the PolicyPeriod. On upgrade, the upgrade trigger tries to set Job.SelectedVersion to the non-withdrawn branch that has the highest BranchNumber (the auto-numbered field). If all branches of a job are withdrawn, then SelectedVersion will be set to the branch with the highest BranchNumber. In the user interface, the button and labels changed from Make Active to Make Selected, and now affect both the wizard user interface and set this new field.
CC-55223	Reporting views now use the ActivePeriod field rather than sub-selects. this should improve performance.
CC-55224	PolicyCenter to ClaimCenter reporting join now uses PeriodID. Joining on PeriodID is safer than using PolicyNumber and EffDate.  This fix has a dependency on a change in ClaimCenter 6.0.1.
CC-55305	The calculation of claim financials totals now uses incremental amounts for a given period and as of an end date.  This fix has a dependency upon a change in ClaimCenter 6.0.1.
CC-55334	Added some stored procedures in support of financial reconciliation between PolicyCenter and BillingCenter.
CC-55666	Add purging procedures for the following reporting tables: <ul style="list-style-type: none"><li>• Denorm table of transactions</li><li>• Accruals table</li></ul>
CC-56086	Converted reporting views and stored procedures to Oracle syntax.
CC-56768	Created Oracle version of all reporting SQL Server files. These can be used by PolicyCenter reports.
PL-8086	Guidewire has modified the way user permissions work with reports so it is now possible to <i>hide</i> a report in the user interface, but grant access permission to specific users.
<b>Searching</b>	
CC-46504	Improved the Search screen by adding a Primary Insured Information heading for the First Name, Last Name, Company Name, and Official ID fields.
CC-46736	Fixed an issue in the Administration tab. Previously, if an organization did not have a preferred underwriter, the Preferred Underwriter column in the search results contained a blank link to a user called Null.
<b>SOAP APIs</b>	
CC-56710	The IDataExtractionAPI web service is now deprecated, as described in the chapter describing integration changes in “ <i>What’s New and Changed</i> ”. In addition, note that the web service no longer has the @WebService annotation. If you need to use this temporarily to transition your integration code to custom services you can do so by removing the comment marker before the annotation. However, Guidewire strongly recommends against using the methods that have the @DoNotPublish annotation. Those methods allow code injections, so you must <i>not</i> use them in a production system.
<b>Studio</b>	
CC-54156	The following improvements were made to the Offerings tab for Coverage, Covterms, Options, Packages, and Modifiers. <ol style="list-style-type: none"><li>1. The words <b>Exclude</b> and <b>Include</b> have been removed from the <b>Exclude-&gt;</b> and <b>&lt;-Include</b> buttons.</li><li>2. The label on the left-hand panel is <b>Included or Implied</b>. This is now <b>Included or implied</b> in these Offerings.</li><li>3. The label on the right-hand panel is <b>Line-Item Excluded</b>. This is now <b>Disabled</b> in these Offerings.</li></ol>
CC-54866	Reformatted typelist (.tti, .tix, and .ttix) and entity (.eti, .eix, and .etx) files as follows: <ul style="list-style-type: none"><li>• Limit line length to 120 char</li><li>• Order the properties so that name is first, type is second</li><li>• Order the lines so that columns are first, then typekeys, foreign keys, arrays. The implementsEntity and implementsInterface elements appear last.</li></ul>
CC-55507	Fixed an issue where PolicyFile_PriorHistoryDV did not display fully in Studio.
CC-56023	Fixed an issue where Studio verification did not catch missing required code fields in system tables.
CC-56068	Fixed an issue where Studio incorrectly reported errors about product model public identifiers that did not conform to Java identifier syntax.

ID	Description
PL-437	Guidewire has changed Studio behavior so that it now flags an empty Rule Condition as an error and displays a red error marker in the right-hand margin of the Condition editor. If you verify the rule, Studio generates a <i>Missing return value</i> error.
PL-7239	Corrected an issue that caused a <code>NullPointerException</code> if you disabled a PCF element, then attempted to save the file.
PL-7595	Guidewire has modified Studio behavior. If you open a resource under <b>Data Model Extensions</b> , <b>Web Resources</b> , and <b>Other Resources</b> , then stop Studio, Studio correctly opens the previously open views upon Studio restart.
PL-7823	Corrected an issue in which adding a locale for a type code for a subtype typelist did not work correctly.
PL-7930	Corrected an issue in which Studio indicated a row iterator value field was invalid, but Studio verification did not show any errors.
PL-7974	The <code>createhistogram</code> column attribute is now available on the <code>column-override</code> element in <code>extensions.xml</code> . This means the customer can override this attribute from <code>false</code> to <code>true</code> or <code>true</code> to <code>false</code> on a column that is part of the base configuration.
PL-7996	Corrected an issue in which using the keyboard shortcut <code>CTRL-E</code> to return to a previously visited <b>Display Key</b> tab did not work correctly.
PL-8101	Guidewire has modified Studio behavior so that it now opens files with <code>.gsm</code> extensions in a text editor view.
PL-8132	Corrected an issue in which opening a previously unedited PCF file and attempting to perform a <b>Change mode...</b> did not correctly copy the file to the configuration module, resulting in checksum errors on server restart.
PL-8179	Guidewire has added an <code>oracleMarkColumnsUnused</code> attribute to the <code>&lt;database&gt;&lt;upgrade&gt;</code> element in <code>config.xml</code> . The database upgrade removes some columns. For Oracle, you can configure whether the removed columns are dropped immediately or are marked as unused through the use of the <code>&lt;upgrade&gt;</code> attribute. (The default is to mark the columns as unused.) The advantage of marking the columns as unused is the potential reduction of the upgrade time, as it would eliminate the scanning of the table and potential updating of multiple database pages. The DBA could then remove the unused columns during a maintenance window at a later time. Note that it is not easy to determine the potential space that could be reclaimed by dropping the unused columns or the potential cost of dropping the unused columns.
PL-8307	Guidewire has modified the behavior of JavaDoc (both Gosu and Java) display on overridden methods. If the overridden method does not have its own JavaDoc, Studio now displays the JavaDoc of the superclass method.
PL-8610	Guidewire has modified the default for running tests from Studio from <code>PermSize=64m</code> to <code>permSize=128m</code> . These values now appear in the default settings if running tests in Studio. If you already have sets defaults on your machine, this does not override them. Instead, you need to change the defaults manually (if you run into problems) by changing <code>-XX:PermSize=64m</code> to <code>-XX:MaxPermSize=128m</code> in the JVM command line options.
PL-8997	Corrected an issue in which Studio failed to correctly create a new Rule Set category in the Resources tree if no rules directory existed in the configuration module.
PL-9214	It is possible to configure Studio to open XML files directly in an XML editor that is external to Guidewire Studio. To facilitate this process, Guidewire now provides the following optional XML attribute: <code>xmlns="http://guidewire.com/datamodel"</code> This attribute defines the namespace for the XML elements in the file. For information on how to use this feature, see “Working with an External XML Editor” in the <i>Configuration Guide</i> .
PL-9666	Corrected an issue in which implementing the <code>OldEncryption</code> plugin caused Studio to incorrectly show the plugin under <code>Plugins → gw → plugin → util → OldEncryption</code> as well as <code>Plugins → gw → plugin → util → IEncryption → OldEncryption</code> .
<b>System administration</b>	
CC-54959	In <code>logging.properties</code> located in <b>Other Resources → logging</b> , changed <code>Log4j.additivity.Server.Database=false</code> . If true, all database logging goes to <code>stdout</code> .
PL-8053	Guidewire has added a new <b>Server Tools → Runtime Environment Info</b> page that details, among other information, the specific application version, and Java Runtime information and other useful information.

ID	Description
PL-8055	Guidewire has added a new download to the <b>System Tools → Data Distribution Info</b> page that reports row counts and growth rates of tables within a specified span.
PL-8108	Guidewire has implemented database searching and sorting appropriate to non-English locales.
PL-8207	Guidewire has modified the colors used on the charts in the <b>System Tools → Cache Info</b> page to make the information more clear.
PL-8405	Guidewire has added new Oracle-only parameters to the <b>System Tools → Database Parameters Info</b> page. They are: <ul style="list-style-type: none"> <li>• Linguistic Search Options</li> <li>• Linguistic search strength</li> <li>• Linguistic search function name</li> </ul>
<b>Underwriting authority</b>	
CC-51530	Added new <b>UnlimitedApproval</b> permission which is assigned only to su by default. In the <b>Risk Analysis UW Issue</b> summary screen, if the underwriting issue meets the conditions to have an <b>Approve</b> button and: <ul style="list-style-type: none"> <li>• The issue is not within the user's authority to approve (making the normal <b>Approve</b> button inactive), and</li> <li>• The user has the <b>UnlimitedApproval</b> permission</li> </ul> then PolicyCenter displays a <b>Special Approve</b> button to the right of the other buttons.  If the <b>Special Approve</b> button is clicked, the user is brought to the same screen as a single-approval request, with the addition of the following warning text: <i>Approval will be through UnlimitedApproval permission, not normal Authority Profiles.</i>
CC-53410	Fixed an issue where <b>Release Lock</b> did not release the lock when a submission had two different versions.
CC-53651	Fixed an issue which generated an <b>EvaluationException</b> if you requested approval for an evaluation issue but did not properly assign the activity.
CC-53909	Converted uses of <b>PolicyEvalIssue</b> to <b>UWIssue</b> . These include question sets which hold metadata related to <b>PolicyEvalIssue</b> in the description of the dataset. Changes include the following: <ul style="list-style-type: none"> <li>• Removed <b>PolicyEvalIssue</b>.</li> <li>• Removed methods related to <b>PolicyEvalIssue</b> on <b>PolicyPeriod</b>, as well as the associated array.</li> <li>• Removed the <b>PolicyEvalIssue</b> evaluation and UW authority-related rule sets.</li> </ul>
CC-54944	The <b>UWIssueApproval</b> entity (available in PolicyCenter 4.0.0) has been removed, and the fields previously on this issue have been moved to <b>UWIssue</b> . For compatibility with existing code, a Gosu version of <b>UWIssueApproval</b> is provided so that most Gosu code which manipulates issue and approvals can remain unchanged.
CC-55314	The <b>UWIssueValueType</b> class is no longer generic.
CC-55501	Fixed an issue where revealing the slice selector for underwriting issues could lead to edits which invalidated the out-of-sequence integrity.
CC-55760	Fixed an issue where the user received an unhandled exception when trying to view the slice date in an out-of-sequence scenario.
CC-55770	Fixed an issue to release the quote when the producer approves last issue blocking quote release.
CC-55967	For <b>Unsaved Work</b> , added <b>countsAsWork="false"</b> to the <b>NewManualUWIssuePopup.pcf</b> called by the submission wizard.
CC-56012	For <b>Unsaved Work</b> , added <b>countsAsWork="false"</b> to the <b>RiskApprovalDetailsPopup.pcf</b> called by the submission wizard.
CC-56017	Changed the text of the <b>Status</b> message on the <b>Risk Approval Details</b> screen. The message <b>Removed</b> is replaced by <b>Issue no longer applies</b> .
CC-56090	Added underwriting authority for underwriters and underwriting managers to approve a <b>Policy Risk Segmentation</b> issue.
CC-56149	Fixed an issue with auto-approvable underwriting issues not appearing on the <b>Risk Analysis</b> screen. Previously, an auto-approved underwriting issue that blocked progress did not appear as blocking on the <b>Risk Analysis</b> screen.
CC-56616	Fixed an issue where automatic selection of the underwriting company selected the best match based on a possibly invalid licensing state.
CC-56707	Fixed an issue where a producer could <b>Save Draft</b> in a locked rewrite job.

ID	Description
CC-57156	Fixed an issue with configuring underwriting issues. The code to specify blocking point/job-type combinations for new checking sets was in read-only job files ( <code>SubmissionProcess.gs</code> , for example). This code was moved to <code>JobProcessUWIssueEvaluator.gs</code> , which is editable.
<b>Validation</b>	
CC-56688	Fixed typographical error in validation message: <i>Good drive discount not applied to these eligible...</i> The word <i>drive</i> should be <i>driver</i> .
<b>Web</b>	
PL-8749	Guidewire has added a new configuration parameter: <code>WizardPrevNextButtonsVisible</code> . Setting this parameter to <code>true</code> always renders the <code>Back</code> and <code>Next</code> buttons on a wizard step even if it is not available (rendered as grayed-out), to maintain consistent positions of other buttons in the wizard toolbar.
PL-9307	Guidewire discovered a security vulnerability in a previous ClaimCenter release in which a user who carefully manipulated a server request was able to view more data than that to which the user had legitimate access. This vulnerability was viewed on the Common Vulnerability Scoring System (CVSS) as Medium, with a low likelihood of discovery.  Guidewire has modified the application to detect this kind of malformed request and to return permitted data only. Guidewire has incorporated the fix into this release and into all future releases.
PL-9328	Fixed an issue in which a CSV export to Microsoft Excel using an English (UK) locale created an invalid character.
PL-9396	Corrected an issue that caused the menu button label color to be incorrect if user selected a list row.
<b>Workflow</b>	
CC-52405	The <code>WorkflowLV</code> on the <code>Admin → Workflows</code> search screen now displays the name of main contact instead of the role name.

## 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 if they 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.

### WebSphere timeout during initial startup (CC-4904)

- **Issue:** The initial creation of the PolicyCenter database may take longer than the default WebSphere startup timeouts will allow, and WebSphere may abort the startup before the database upgrade has completed. In this situation, the `Default_Server_stderr.log` file will show an interruption in the `Applying upgrade step #### of #####` log messages.
- **Workaround:** Increase the Ping Timeout and Ping Initial Timeout values for WebSphere. These can be found in the WSADMIN Console; click `Nodes`, drill down to your application server, click the `Advanced` tab, and increase the values.

### On AIX without an X11 server, PolicyCenter may report that an X11 resource is missing (CC-5889)

- **Issue:** PolicyCenter does not require an X11 server; however other components of the PolicyCenter environment (such as Java) may inadvertently have this requirement. If so, PolicyCenter may report that an X11 resource is missing when it is starting up.
- **Workaround:** Install an X11 server, or contact Guidewire support to determine if the dependency can be removed.

### Port conflict when starting PolicyCenter with JMX RMI adapter enabled and Microsoft Outlook running (CC-6964)

- **Issue:** When Microsoft Outlook starts, it chooses several ports to use for its connection to an Exchange server. In some instances, Outlook may use port 1099, which is a port that the PolicyCenter JMX RMI

adapter can use, if enabled. The PolicyCenter server will then report a port conflict during startup when it tries to use that port for the RMI adapter.

- **Workaround:** Close Outlook to free the port, and then start the PolicyCenter server again. You can also change the port that the RMI adapter uses by editing the JMXManagementPlugin through Guidewire Studio.

#### **Commercial Package Policy: Exception when quoting a preempted policy change job (CC-54333)**

- **Issue:** In a commercial package policy, quoting a preempted policy change job results in an exception. In a submission job, create a policy that contains all lines of business. Quote and issue the submission. Start a policy change job. Remove a line then add a location and use it on the GL line. Quote the policy change. Start another policy change job with the same effective date. Do not remove the line but add the same location and use it on the GL line. Quote the policy change. Bind the submission job (which has removed commercial property line) In the second policy change job, handle preemptions and apply all changes. When you try to quote, PolicyCenter throws an exception.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

#### **PolicyCenter displays many warnings when regenerating the toolkit for regenerating Javadoc (CC-55564)**

- **Issue:** This issue occurs when running the regen-toolkit command.
- **Workaround:** In general, warnings do not indicate loss of function. Guidewire is aware of this issue and will address it in a future release.

#### **Duplicate contact when merging contacts through API (CC-56177)**

- **Issue:** When you merge two instances of a contact through the API, you may get duplicate contacts for the following contact roles:
  - Labor Contact
  - Named Insured
  - Additional Interest
  - Additional Insured
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

#### **Studio: “Verify Changed Resources” is not implemented in the product model (PC-6928)**

- **Issue:** The Studio Tools → Verify → Verify Changed Resources command does not work with the PolicyCenter product model.
- **Workaround:** Use one of the other verification commands instead. See “Validating Studio Resources” in the *PolicyCenter Configuration Guide* for details.

#### **Zone data with missing values cannot be loaded (PL-9123)**

- **Issue:** Zone data, which is used for associating postal codes with cities and regions, does not load if it is incomplete.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

#### **Pre-update rules for extension entities (PL-8469)**

- **Issue:** In configurations where the parameter UseOldStylePreUpdate is set to TRUE, pre-update rules cannot be applied to entities that cannot be validated.
- **Workaround:** Add `<implementsEntity name="Validatable"/>` to the definitions of all entities that the pre-update rules manipulate.

#### **Need a summary of underwriting issues across all policies in a job group (CC-54177)**

- **Issue:** The JobGroupRiskAnalysisLV needs to show a summary of underwriting issues across all policies in the job group.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Provide a way to filter out added and removed subtentities in a policy differences tree (CC-54957)**

- **Issue:** Need a way to filter out the children of an added or removed entity.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Load Sample data results in stack traces when H2 is the database (CC-55397)**

- **Issue:** Loading the Small sample data from the Internal Tools → PC Sample Data menu results in a stack trace when H2 is the database. PolicyCenter reports errors related to geocoding while loading the small accounts. The first Policy loads and quotes, and the errors and stack traces appear. The messaging plugin displays a suspending message, then the remaining policies load.
- **Workaround:** Sample data loads without the stack trace if the data is loaded through the config.xml file:  
`<param name="InitialSampleDataSet" value="small"/>`

**In an out-of-sequence policy change, the effective date does not appear in the description. (CC-56001)**

- **Issue:** In an out-of-sequence policy change, if a change is made at a date other than the edit effective date of the policy, that effective date does not appear in the description in the Differences tab of the Policy Review screen. A change can be made at a date other than the edit effective date in one of two ways:
  - An out-of-sequence conflict is overridden, so a value changes as of the date of the future conflict.
  - A validation error occurs in a future slice and the user makes a change in the future slice.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Internal user incorrectly allowed to see producer codes outside their security zones (CC-56955)**

- **Issue:** An internal user who does not have the View All Users permission, can find and view producer codes outside of their security zone. The internal user should not be able to do this.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Wrong payment plan preview for installment plans when using BillingCenter integration (CC-57089)**

- **Issue:** When using the BillingCenter integration, PolicyCenter displays the wrong payment plan preview for installment plans when Payment Method is switched from Reporting Plan to Installments. The steps to reproduce this problem are:
  1. Start a new workers' compensation submission.
  2. Advance to the WC Coverages screen.
  3. Click Add Class, and fill in the required fields.
  4. Click Quote.
  5. On the Payments screen, select Reporting Plan for Payment Method.
  6. Click Save Draft.
  7. Click Edit Submission.
  8. Click Quote.
  9. Change the Payment Method to Installments.
  10. Click Preview Payments.The preview displays incorrect payment information.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Improve the loading time of question sets by adding new synchronization method (CC-57134)**

- **Issue:** To improve the load time of question set screens, provide a method that synchronizes a specified question set on a specified answer container for a particular screen.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**"An invalid quote was generated" message is displayed after policy changes in a Commercial Package Policy (CC-57154)**

- **Issue:** The user receives “An invalid quote was generated” validation message when an out-of-sequence policy change is made on a commercial package policy under the following conditions:
  - After two in-sequence policy changes, the second of which removed something
  - Dated in-between the two in sequence changesThe user cannot quote the policy change.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**Eliminate unnecessary checking for question availability (CC-57211)**

- **Issue:** In some cases when checking the availability of a question set, PolicyCenter checks for the availability of each question in the question set twice. This extra checking should be eliminated.
- **Workaround:** Guidewire is aware of this issue and will address it in a future release.

**The regen-toolkit target has been removed (PL-10301)**

- **Issue:** The gwpc tool regen-toolkit target no longer exists.
- **Workaround:** Call regen-java-api for Java plugin libraries and documentation, or regen-soap-api for SOAP libraries and documentation.



# Guidewire PolicyCenter 4.0.2 Release Notes

## Release 4.0.2.9

Copyright © 2001-2010 Guidewire Software, Inc. All rights reserved. Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Insurance Suite, Guidewire ContactCenter, Guidewire Studio, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries. All other product names are trademarks of their respective companies.

**This documentation is published as Guidewire Confidential.** The contents of this documentation, including product architecture details and APIs, are considered confidential and are fully protected by customer licensing confidentiality agreements and signed Non-Disclosure Agreements (NDAs).

This product includes software developed by the Apache Software Foundation (<http://www.apache.org>).

## Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Changes in This Release
- Improvements and General Issues
- Known Issues and Limitations

## Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

## Version Number

- This release of Guidewire PolicyCenter is 4.0.2.9.
- If you are using the integration with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 3.0.1.
- For Standard Reporting, this release of PolicyCenter supports InetSoft StyleReport Enterprise Edition 10.1, 20100113.

## Installation Requirements

For installation requirements, see the *PolicyCenter Installation Guide*.

## Support

This document includes important information about the current release of Guidewire PolicyCenter. For assistance with this software release, contact Guidewire Customer Support. You can contact support on the Web at <http://portal.guidewire.com>, by email at [support@guidewire.com](mailto:support@guidewire.com), or by phone at +1-650-356-4955.

## Installing This Release

For installation information, see the *PolicyCenter Installation Guide*.

This section describes the following changes to PolicyCenter:

- InetSoft Evaluation

## InetSoft Evaluation

You can evaluate InetSoft Style Report Enterprise Edition Version 10.1 using the following license key:

L000-798-ERX-0000BF00100001F-F2AFC9DD34ED

If you want to use InetSoft and the PolicyCenter reporting module in a production environment, contact Guidewire Customer Support to request a full license key. For more information, see the *PolicyCenter Reporting Guide*.

---

**WARNING** The build number of the reporting install files must match the build number of the Guidewire PolicyCenter installation. If the build numbers do not match, you will have problems with your reporting installation.

---

## Changes in This Release

This section describes the product changes in this release.

- Base PCF File Changes
- Base Resources Changes

## Base PCF File Changes

All links below require the `readme_files` directory on your local disk.

### PolicyCenter release 4.0.1 to 4.0.2

- To view a report of the changes in the base PCF files in the `modules/pc` directory, [click here](#).

- To view a report of the changes in the base PCF files in the `modules/p1` directory, [click here](#).

## Base Resources Changes

All links below require the `readme_files` directory on your local disk.

### PolicyCenter release 4.0.1 to 4.0.2

- To view a report of the changes in the base resources in the `modules/bc` directory, [click here](#).

## Improvements and General Issues

This topic describes the primary improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This is not meant to be a comprehensive list. Many performance related issues were addressed in this release but are not listed here.

ID	Description
<b>Account</b>	
PC-11039	Made a change to the search behavior in the Organization Search screen on the Administration tab. When searching for an Account #, you must provide the full account number. Previously you could find accounts by providing the first few numbers or letters of the account number. This improves performance when searching for organizations.
PC-11119	Fixed an issue on the Work Orders in Progress listview in the Account File Summary screen for jobs that had multiple versions. If a job was open, the listview did not display that job as open if any version was withdrawn.
PC-8411	Fixed an issue where the FEIN field on the Create account and Policy Info screens of a job had different field validators. This problem occurred for Company accounts with FEIN formatted as xxx-xx-xxxx.
<b>Administration</b>	
PC-10728	Fixed an issue in Administration → Event Messages → Message Search where search by product did not work.
PC-7395	Organizations in the Group hierarchy are now validated. New diagnostic messages can be generated, so there are new display keys that you may need to localize. As part of this change, any non-root Group now requires a parent. The Javadoc documentation for the <code>ProducerAPI.synchronizeBranch()</code> method was modified to accommodate these changes.
PC-7930	Fixed an error that occurred when trying to create an organization when PolicyCenter had no administrative data (such as users or organizations). Previously PolicyCenter displayed an <i>Error Occurred</i> message even though it created the Organization.
PC-7943	Some additional error checking was added to administrative data. An Organization which is a Producer type (Agency, Broker, Insurer, or Managing General Agent) is required to have a ProducerCode. PolicyCenter generates an exception when you attempt to add a User to a ProducerCode when there are no Roles associated with the ProducerCode.
<b>Contacts</b>	
PC-11306	Policy search results with last name now return policies where the last name matches at any point in time, rather than just policies that currently match.
PC-11307	To improve search performance, policy search results with last name now searches only for primary named insured contacts with a matching name, rather than all contacts.
PC-11388	Fixed an issue with policy search on last name when there is a name change. The search results show the current name on the account rather than the name when the policy was bound.
PC-7258	Fixed an issue where the code for removing an <code>AdditionalInterestDetail</code> could remove an <code>AccountContact</code> when that <code>AccountContact</code> had other roles. This method is in the <code>AddlInterestDetailsMethodsDefaultImpl.gs</code> in <code>Classes.gw.api.contact</code> package.
PC-7338	If you change the primary named insured on a policy, PolicyCenter now blanks out the relationship fields of existing additional named insureds. This is because the additional named insureds may have different relationships to the new primary named insured.

<b>ID</b>	<b>Description</b>
PC-7421	<p>Fixed an issue that resulted in duplicate contacts when merging contacts through API. Previously, when you merged two instances of a contact through the API, you may have gotten duplicate contacts for the following contact roles:</p> <ul style="list-style-type: none"> <li>• Labor Contact</li> <li>• Named Insured</li> <li>• Additional Interest</li> <li>• Additional Insured</li> </ul> <p>This was previously logged as known issue CC-56177.</p>
<b>Core</b>	
PL-8740	<p>Guidewire has added two new methods to the <code>IDatabaseUpgrade</code> plugin:</p> <pre>function preUpgrade(context : IUpgradeContext) function postUpgrade(context : IUpgradeContext)</pre> <p>Use these methods as hooks to perform custom SQL before and after an upgrade. If you have already implemented <code>IDatabaseUpgrade</code> in order to upgrade foreign key entities to <code>edgeForeignKey</code> entities, then you still need to implement these new methods. However, you can make them non-operational.</p>
PL-8746	Guidewire has corrected an issue with <code>regen-datamapping-together</code> and <code>regen-datamapping-split</code> that did not generate mappings for custom entities and extensions.
PL-9046	Guidewire has modified the <code>localization.xml</code> file to use an XML namespace, rather than a hard-coded path to a schema. The namespace is <code>http://guidewire.com/localization</code> . The schema is located at <code>modules/p1/xsd/localization.xsd</code>
PL-9163	Guidewire has modified the <code>config.xml</code> file to use an XML namespace, rather than a hard-coded path to a schema. The namespace is <code>http://guidewire.com/config</code> . The schema is located at <code>modules/p1/xsd/config.xsd</code> .
PL-9223	Guidewire now provides the ability to generate the Data Dictionary and Security Dictionary as you build the application .war file. Use the following command: <code>gwcc build-war -Dconfig.war.dictionary=true</code> .
PL-9793	Guidewire has added the ability to trigger a data model upgrade without modifying any of the data model entity extension files. A new file in Data Model extensions, called <code>extensions.properties</code> , contains a single (numeric) property: <code>version</code> . Guidewire stores the value of this property after an upgrade. As the server starts, the application forces an upgrade if the version in the database is less than the version in this file. The upgrade fails if the version in the database is greater than the version in this file.
PL-9914	Guidewire has corrected a localization issue in which the entity type display name did not use the subtype key's localized name. This led to some user interface labels not being localized correctly.
PL-9987	Guidewire has corrected an issue in which running the <code>regen-pcfmapping</code> tool did not correctly pick up custom PCF pages.
PL-10002	Guidewire has provided the ability to add events to an entity via extension. So, if a base entity does not declare events, a customer can add events to the entity by extending the entity and adding the <code>&lt;events&gt;</code> tag.
PL-10095	Corrected an issue with the <code>MessageTransport</code> plugin that caused it to not shut down if you shut down the application server.
PL-10141	In calling a Java plugin from Gosu code, the application did not automatically convert the domain entity class to an external entity class. For the application to handle this correctly, you need to copy the external entity jar ( <code>gw-entity-XX.jar</code> ) from:  <code>java-api/lib/ to modules/configuration/plugins/shared/lib/</code>

ID	Description
PL-10183	<p>Guidewire modified the behavior of the encryption plugin (<code>IEncryption</code>). It is now possible to register any number of <code>IEncryption</code> plugins, just as with messaging plugins. Only one encryption plugin is the current encryption plugin. There is a new <code>config.xml</code> configuration parameter called <code>CurrentEncryptionPlugin</code>. It specifies which encryption plugin (among potentially multiple implementations) is the current encryption algorithm. Set the parameter to the plugin name (not the class name). If this configuration parameter is missing, the application uses the <code>IEncryption</code> plugin with name <code>IEncryption</code>. Any legacy existing <code>IEncryption</code> plugins automatically have the plugin name <code>IEncryption</code>. When you use Studio in the Plugins editor to add an implementation of <code>IEncryption</code>, Studio prompts you for a text value to use as the plugin name for this implementation. Guidewire strongly recommends you set the plugin name for encryption plugins to names that describe the algorithm. For example, <code>encryptDES3</code>.</p> <p>Every <code>IEncryption</code> plugin implementation returns the plugin's unique encryption ID as a <code>String</code> value. This is exposed to Gosu as the <code>propertyEncryptionIdentifier</code>. This API did not change in this release. However, starting in this release this encryption ID is very important. The application decides whether to upgrade the encrypted data with a new algorithm by comparing the encryption ID of the current encryption plugin to the encryption ID associated with the database last time the server ran. If the encryption ID is different, the upgrader decrypts the data with the old algorithm and re-encrypts it with the new algorithm. For new plugins, set the encryption ID to describe the algorithm. Do not confuse the encryption ID with the plugin name. The encryption ID is saved in the database. As in previous releases, if the upgrader detects added or removed encryption for any data model fields, the upgrader encrypts the field in the database using the current encryption plugin.</p> <p>The upgrade-specific plugin interface, <code>OldEncryption</code>, is now unsupported and removed from the product. Its functionality is replaced by comparing the encryption ID of data in the database with the encryption ID associated with the current encryption plugin. This triggers decryption and then re-encryption of the data using the newer encryption plugin. The upgrader checks the encryption ID of the plugin to determine whether to decrypt and re-encrypt the encrypted fields in the database.</p>
PL-10402	Guidewire has corrected an issue that if you started the tools from <code>Ant/Program.gs</code> , it produced a <code>NoClassDefFoundError: Files</code> on some Windows machines.
PL-10601	Guidewire has corrected an issue that caused the <code>regen-toolkit</code> command to successfully build due to an issue with how the application handled third-party JAR files.
PL-11677	PolicyCenter fixed a bug in how messaging destinations read new messages from the messaging queue and dispatch them to messaging plugins. In a rare edge case, a non-account-specific message could enter the send queue before the related account-specific messages because of the separate database queries for these two categories of messages. Starting in this release, in each round of sending, each messaging destination always reads account-specific messages (safe-ordered messages) from the queue before non-account-specific messages. (Although the read order changed, the dispatch order to messaging plugins did not change. As before, messaging destination threads send non-account-specific messages before account-specific messages.)
<b>Data Model</b>	
PC-10120	Reviewed and updated usage of <code>cascadeDelete="true"</code> on entities. In some cases, this was incorrectly set to <code>false</code> . This problem was found in Commercial Package Policy (CPP). Previously, if you added the Business Auto line in CPP then removed it, you received a DB referential constraint violation exception.
PC-11000	<p>In versions of PolicyCenter prior to 4.0.2, all Modifiers were stored in one table with subtypes for different lines and modifiable associations. This could cause a performance issue due to the time required to maintain numerous indexes on <code>pc_modifier</code>.</p> <p>In PolicyCenter 4.0.2, the <code>Modifier</code> and <code>RateFactor</code> entities are modeled using delegates, similar to <code>Coverage</code> entities. For more information, see the <i>Upgrade Guide</i>.</p>
PC-11092	The following delegates are now extendable.
PC-11094	<ul style="list-style-type: none"> <li>• <code>Auditable.eti</code></li> <li>• <code>Cost.eti</code></li> </ul>
PC-11095	<ul style="list-style-type: none"> <li>• <code>Coverable.eti</code></li> <li>• <code>Coverage.eti</code></li> <li>• <code>Exclusion.eti</code></li> <li>• <code>Modifiable.eti</code></li> <li>• <code>Modifier.eti</code></li> <li>• <code>PolicyCondition.eti</code></li> <li>• <code>RateFactor.eti</code></li> <li>• <code>Transaction.eti</code></li> <li>• <code>PCAssignable.eti</code></li> <li>• <code>UWIssueDelegate.eti</code></li> </ul>

ID	Description
PC-11252	<p>In Studio, you can now view and edit the following adapter classes for conditions, exclusion, coverages, and coverables:</p> <ul style="list-style-type: none"> <li>• BAStateCondConditionAdapter.gs</li> <li>• BAStateExclExclusionAdapter.gs</li> <li>• BusinessOwnersCondConditionAdapter.gs</li> <li>• BusinessOwnersExclExclusionAdapter.gs</li> <li>• CommercialPropertyCondConditionAdapter.gs</li> <li>• CommercialPropertyExclExclusionAdapter.gs</li> <li>• ContractorsEquipPartCondConditionAdapter.gs</li> <li>• ContractorsEquipPartExclExclusionAdapter.gs</li> <li>• ContractorsEquipmentExclExclusionAdapter.gs</li> <li>• GeneralLiabilityCondConditionAdapter.gs</li> <li>• GeneralLiabilityExclExclusionAdapter.gs</li> <li>• IMAccountsRecPartCondConditionAdapter.gs</li> <li>• IMAccountsRecPartExclExclusionAdapter.gs</li> <li>• PersonalAutoCondConditionAdapter.gs</li> <li>• PersonalAutoExclExclusionAdapter.gs</li> <li>• WorkersCompensationCondConditionAdapter.gs</li> <li>• WorkersCompensationExclExclusionAdapter.gs</li> <li>• BALineCoverageSymbolGroupAdapter.gs</li> <li>• ContractorsEquipmentPartCoverableAdapter.gs</li> <li>• ContractorsEquipmentPartCoverageAdapter.gs</li> <li>• IMARCovCoverageAdapter.gs</li> <li>• IMAccountsRecPartCoverableAdapter.gs</li> <li>• IMAccountsRecPartCoverageAdapter.gs</li> <li>• PolicyAddressSyncableAdapter.gs</li> <li>• PolicyPeriodTransactionCalculatorAdapter.gs</li> <li>• ProductModifiableAdapter.gs</li> </ul>
PC-11299	Added a <code>MostRecentTerm</code> boolean on <code>PolicyTerm</code> . This flags the future-most term for a policy.
<b>Documentation</b>	
PC-99019	Added a topic on how to create a custom line of business.
	Added a topic on how to add premium audit to a line of business.
	Added a "General Liability Screens" topic to the <i>Application Guide</i> .
<b>Infrastructure</b>	
PC-7297	Fixed an issue where PolicyCenter displayed many warnings when regenerating the toolkit for regenerating Javadoc. This issue occurs when running the <code>regen-toolkit</code> command. This was previously known issue CC-55564.
<b>Integration</b>	
PC-7268	Improved PolicyCenter error logging when quoting ( <code>QuoteProcess.gs</code> ). Log messages now contain the error message from the exception.
PC-7495	Improved the web service interface <code>CCPolicySearchIntegrationV2</code> to properly use the time portion of the effective date parameter (the <code>AsOfDate</code> parameter) when searching. In previous releases, this API stripped out the time portion before searching for policies by effective date. If you use this web service with any third-party integrations, be sure to set the time appropriately, either midnight (the beginning of the day) or a specific time.
PC-8128	The <code>insertAccount</code> method in <code>AccountAPI</code> is now deprecated. Use the <code>addAccount</code> instead. This method returns the <code>AccountNumber</code> of the created account.
PC-10277	The <code>Message.canSkip</code> method is deprecated due to changes in the messaging system that guarantee that all such messages can be skipped. Therefore, this method is no longer needed.
PC-10730	The <code>IMessagingToolsAPI</code> web service method <code>resyncPolicy</code> is now deprecated. Instead, use <code>resyncAccount</code> to resynchronize all messages associated with an account.
PL-9353	Part of the Guidewire/InetSoft reporting integration involves a security plugin. Guidewire has modified the plugin implementation to now cache up to one user's information to reduce the number of soap calls made between a Guidewire application and the integrated InetSoft server. The cache has a default timeout of 30 seconds. You can modify this value by altering the property <code>gw.usercache.timeoutinseconds</code> in <code>sree.properties</code> . The value is in terms of seconds.
PL-9569	Guidewire has corrected an issue with reporting JAR file ( <code>gw-sree.jar</code> ). Previous builds did not automatically contain the Security Provider classes.
PL-9659	Guidewire has modified the base configuration <code>address-config.xml</code> entry for Canada so that autofill works correctly for both city and postal codes.

ID	Description
PL-9714	Guidewire has deprecated the following method: <code>gw.api.webservice.zone.ZoneImportHelper.importDefaultDataToProduction</code> which deprecated the support for the <code>dataFile</code> attribute in the <code>zone-config.xml</code> file.
PL-9721	Guidewire corrected an issue in which disabling the ActiveX document control caused an attempt to create a document from a template to fail. You can now perform this task even if you disable this ActiveX control. This is similar to CC-57112.
PL-9900	Guidewire has corrected an issue with incorrect formatting and parsing of the values for XSD <code>gMonth</code> , <code>gDay</code> and <code>gMonthDay</code> .
PL-10599	<p>It is possible that you can experience namespace collisions in the names of argument types and return types on published web services. (If this occurs, Studio notifies you with errors starting in PolicyCenter 4.0.0.)</p> <p>To make debugging during upgrade easier, PolicyCenter now has a new configuration parameter called <code>AllowSoapWebServiceReferenceNamespaceCollisions</code>. If set to true, these error messages become warnings. Use this for development and debugging until you have time to rename your classes to fix the namespace collision. This setting is <code>false</code> by default. Guidewire does not support setting this value to true for production servers. (It is unsafe to do so.)</p> <p>Contact Guidewire Customer Support for complete details.</p>
PL-10629	<p>Guidewire has modified the way it supports InetSoft VPM. This work has tightened the Guidewire SOAP security for <code>ISREEAAuthenticationAPI</code> and added three more properties to the <code>sree.properties</code> file.</p> <ul style="list-style-type: none"> <li>• <code>gw.soapuser.username</code>: The user name used by InetSoft to retrieve user/group/role information used by VPM and authentication</li> <li>• <code>gw.soapuser.userpassword</code>: The password associated with the user name.</li> <li>• <code>gw.usercache.size</code>: The cache size for user cache, group cache, and role cache.</li> </ul>

#### Integration with BillingCenter or a billing system

PC-10	PolicyCenter no longer calls the billing system to recalculate the payment plan values after a reinstatement, policy change, or issuance. Previously in these jobs, PolicyCenter called the billing system to recalculate payment plan values after quoting. This is not necessary because the selected payment plan cannot be changed.
PC-11227	<p>The PolicyCenter integration with BillingCenter changed how the applications uniquely refer to a policy period in their integration. In previous PolicyCenter releases, the PolicyCenter integration with BillingCenter used the policy number and periodID as the two IDs that collectively identify a policy term. Starting with BillingCenter 3.0.1 and PolicyCenter 4.0.2, the applications use the policy number and the term number (<code>PolicyPeriod.TermNumber</code>) as the two IDs that identify a policy term. The term number starts with 1 and increments for each renewal and for each rewrite. BillingCenter 3.0.1 web services changed such that the policy period information includes the term number and uses the policy number and term number to identify the policy term in PolicyCenter. In PolicyCenter, the PolicyCenter 4.0.2 calls the updated BillingCenter web services.</p> <p><b>IMPORTANT:</b> This change requires updating your BillingCenter installation to version 3.0.1.</p> <p>Note that the data model differs between the two applications, in part due to how PolicyCenter implements policy revisioning. The BillingCenter version of a policy term is unique using policy number and the term number. Within PolicyCenter, more than one <code>PolicyPeriod</code> entity in the database can have the same policy number and term number.</p>
PC-12	Fixed an issue where PolicyCenter retrieved payment plans from the billing system at the beginning of the job process. To improve performance, PolicyCenter now calls the billing system after a valid quote.
PC-7239	Fixed an issue where the Premium Reporting Plan drop-down menu on the Payment screen displayed plans available in the billing system but not available in PolicyCenter. The drop-down menu now displays plans where the Payment Plan Code matches the ID in the billing system. You specify the Payment Plan Code on the Audit Schedule in Studio.
PC-7255	Fixed a race condition that occurred when two policies were bound at the same time using the same contact. The contact was created twice in BillingCenter.
PC-7347	Fixed an issue where PolicyCenter displayed the wrong payment plan preview for installment plans when using BillingCenter integration. Fixed an issue where PolicyCenter previewed the wrong payment plan. This occurred for installment plans when reporting plan was selected. This was previously logged as known issue CC-57089.

#### Integration with ClaimCenter or a claim system

ID	Description
PC-10928	<p>The following Gosu class related to the ClaimCenter integration is now available in Studio:</p> <ul style="list-style-type: none"> <li>• Classes/gw/webservice/pc/ccintegration/v2/CCContactGenerator.gs</li> </ul> <p>The following files in Classes.gw.webservice.pc.ccintegration.v2ccentities are also available in Studio:</p> <ul style="list-style-type: none"> <li>• CCAddress.gs</li> <li>• CCAdjudicator.gs</li> <li>• CCCCompany.gs</li> <li>• CCCCompanyVendor.gs</li> <li>• CCContact.gs</li> <li>• CCContactAddress.gs</li> <li>• CCLegalVenue.gs</li> <li>• CCOfficialID.gs</li> <li>• CCPCSearchCriteria.gs</li> <li>• CCPerson.gs</li> <li>• CCPersonVendor.gs</li> <li>• CCPPlace.gs</li> </ul>
<b>Job infrastructure</b>	
PC-10781	Fixed an issue where buildings were not correctly renumbered after merging out-of-sequence changes.
PC-10799	Fixed a problem where fixing an out-of-sequence change in a future slice could make it impossible to fix errors in the first slice. This required a change to the file OOSEPanelSet.pcf.
PC-10849	Made a change to the System Tables → uw_issue_types.xml in Studio. In the PAPPrimaryDriverUnder25 issue, removed NOT USED from the description. The PolicyEvalNew ruleset uses this issue.
PC-10899	The policy foreign key for PolicyPeriod always points to the policy. Previously, it returned null in certain circumstances.
PC-11082	Fixed an issue where the Policy Review → Differences tab did not reflect changes made on the Change Conflicts tab.
PC-11405	Removed many usages of PolicyPeriod.openForEdit() to improve performance.
PC-7211	The PolicyAddlInsuredEnhancement.additionalInsuredTypeExistsOnRole() method is defined to return a Boolean but sometimes returned null. The method now returns only true or false.
PC-7215	Fixed an issue where PolicyPeriodImpl.getOpenPeriodStatusSet() only included typekeys that were specified as one of the Include elements on TypeFilter. It ignored the Categories, Excludes, and Include All flag on TypeFilter.
PC-7288	<p>Fixed an issue in an out-of-sequence policy change where the effective date did not appear in the description. This was previously known issue CC-56001.</p> <p>A new display key has been created.</p> <pre>Java.Diff.DiffEffectiveAsOf = {0} [eff. {1}]</pre> <p>If a diff node has an effective date that does not match the edit effective date for the period, it will appear in the label. This display key defines how it will appear.</p> <p>These methods were modified to include a Date as an argument. If this date is non-null, it will appear as part of the label on the node.</p> <pre>gw.api.diff.node.generator.IAddNodeGenerator:createAddNode(...) gw.api.diff.node.generator.IRemoveNodeGenerator:createRemoveNode(...) gw.api.diff.node.generator.IPropertyNodeGenerator:createChangeNode(...)</pre> <p>Also the constructor for gw.diff.tree.DiffTree now includes the typelist DiffReason. If this reason is null, or MultiVersion job, then all effective dates in labels will be suppressed. Otherwise, these dates will appear when they do not match the edit effective date of the branch.</p>
PC-7419	Added DemoPreUpdateImpl class to replace the former Pre-update ruleset. The Pre-update plugin (PreUpdateHandlerImpl) now delegates to this class, rather than using PreUpdateUtil to execute the rule-set.
PC-7422	Fixed an issue in the policy change job where the address of the insured changes. On the Policy Review screen, the Differences tab now shows that the Policy Address changed if this address is used as the policy address. Previously, this tab showed that the Primary Named Insured Address changed.

ID	Description
PC-7530	<p>Previously, only submission jobs expired. Now all job types can expire. In the default configuration, only submissions expire, but you can enable expiration for other job types.</p> <p>To enable expiration for other job types:</p> <ol style="list-style-type: none"> <li>1. In Studio, open Other Resources → config.xml.</li> <li>2. Set JobExpireCheck from false to true:  <code>&lt;param name="JobExpireCheck[jobtype]" value="false"/&gt;</code></li> <li>3. (Audit only) In AuditProcess.gv in the Classes.gw.job.uw package, modify the canExpireJob method to return true instead of false.</li> </ol>
PC-7638	<p>The following permissions have been marked as Retired in the SystemPermissionType typelist:</p> <ul style="list-style-type: none"> <li>• reviewcancellation</li> <li>• reviewpolchange</li> <li>• reviewreinstate</li> <li>• reviewrenewal</li> <li>• reviewrewrite</li> <li>• reviewsubmission</li> </ul>
PC-7653	<p>When you click Diff to view the differences between two policies or work orders, the top-level section elements are now sorted in all diff trees provided in the configuration. You can configure how the nodes are sorted by making changes to the files in Other Resources → diff in Studio.</p>
PC-7899	<p>Fixed an issue on the Differences screen. Previously in a policy change job, the added entity and the children of the added entity appeared separately. PolicyCenter now displays these together. The same problem was fixed for removed entities.</p>
PC-8009	<p>Fixed several issues that created duplicate contact roles or locations in a preempted job. In some cases, this caused an exception.</p>
PC-8072	<p>Fixed an issue where the PolicyChangeProcess did not validate future out-of-sequence slices at binding.</p>
PC-8967	<p>The JobProcess and its subclasses have been modified. In particular, the JobProcess and its subclasses are now marked as @Export, and can be edited directly.</p> <p>Other changes included improved code-level documentation (GosuDoc and inline comments) and code readability improvements. These changes should have no impact on the behavior of the jobs.</p>
<b>Job: cancellation process</b>	
PC-7186	<p>Fixed an issue where already scheduled cancellations could be withdrawn but could not be rescinded. Scheduled cancellations can now be rescinded but cannot be withdrawn.</p>
<b>Job: policy change</b>	
PC-10765	<p>Fixed stack traces that occurred in preempted policy changes. Also fixed issue where the Bind button appeared in preempted policy changes.</p>
<b>Job: submission process</b>	
PC-8727	<p>The Copy Submission choice is no longer available if the user does not have access to the current producer code of service.</p>
PC-10999	<p>Fixed an issue where the status of a submission was not updated when it changed to draft mode. Previously, the status did not update to draft when you edited a quoted submission.</p>
<b>LOB: commercial package policy</b>	
PC-7619	<p>Fixed an issue where PolicyCenter displayed "An invalid quote was generated" message after policy changes in a Commercial Package Policy. This was previously logged as known issue CC-57154.</p>
PC-7631	<p>Fixed an issue where the base state changed after adding the General Liability Line to Commercial Package. This problem happened if the policy address and the base state were located in different states.</p>
PC-7897	<p>Fixed an exception when quoting a preempted policy change job. This was previously logged as known issue CC-54333.</p>
<b>LOB: general liability</b>	

<b>ID</b>	<b>Description</b>
PC-7858	<p>Added warning messages in general liability with out-of-sequence policy changes to an exposure. The messages appear in the following circumstances:</p> <ol style="list-style-type: none"> <li>1. Create a GL Submission, with one exposure on GL. Quote and issue the policy.</li> <li>2. Start a Policy change, Add a new location, add GL exposure at the new location.</li> <li>3. Start an out-of-sequence policy change, go to GL Exposure page, you will see the new exposure. Select the new exposure row, and click Split.</li> <li>Nothing happens. To alert the user, PolicyCenter now displays a warning message that the exposure will not be split.</li> <li>4. Select the exposure and click End.</li> </ol> <p>The row disappears. To alert the user, PolicyCenter now displays a warning message that the exposure will be removed.</p> <p><b>Note:</b> This issue also applies to general liability in a commercial package policy.</p>
<b>LOB: inland marine</b>	
PC-7276	
PC-7276	Fixed an issue with locations on the Signs screen. Previously, if the user did not have the location they want on the Signs screen, they could not return to the Locations screen without adding a location to the sign. PolicyCenter now generates a warning which becomes an error at quote.
<b>LOB: personal auto</b>	
PC-7184	Fixed an issue with costs in personal auto. Previously, if a cancellation was preempted, PolicyCenter showed cost changes which could not be applied to the preempted cancellation. These cost changes no longer appear in the user interface.
PC-11239	Fixed an issue where the garaging location address on a bound policy changed if the account address was updated. (The location address on a bound policy must not change.)
<b>LOB: workers compensation</b>	
PC-7350	Fixed an issue where PolicyCenter generated a stack trace when you tried to rewrite a policy and could not, then attempt reinstate the canceled policy. Prior to the stack trace, PolicyCenter displayed an error message about slice date out of range.
PC-11296	Fixed an issue with rate factors in workers' compensation. Previously, when there was a split around an anniversary date, PolicyCenter split the Schedule Credits modifier, but did not split the rate factors on that modifier.
<b>Localization</b>	
PC-7656	Fixed an issue with class codes in general liability and commercial package policy products. Previously, you could not select a class code in a localized version of PolicyCenter.
<b>Locations</b>	
PC-7677	Fixed an issue where the MappointGeocodePlugin returned an incorrect route.
PC-11145	Fixed an issue where removing a primary location reported the EffectiveDatedFields.PrimaryLocation as null when it contained an non-null value.
<b>Manageability</b>	
PL-9385	Guidewire has corrected an issue that generated a Null Pointer Exception if you clicked the Download button in the (Server Tools) Workflow Info page. The exception occurred if there were any work items that had a status of <i>Error of Failure</i> .
PL-9697	Guidewire has modified the (Server Tools) Data Distribution page so that you can generate data on the minimum and maximum array sizes that shows the row count changes across distributions. This is for all arrays in the data distribution download.
PL-9812	Guidewire has modified the (Server Tools) Database Storage page so you can choose specific tables and mode.
PL-9895	Guidewire has added a (Server Tools) Server DMV Snapshot screen that displays information on the aggregate usage for all queries in the DMV, along with the percentage of resource usage by the top N queries.
PL-9911	Guidewire has modified the (Server Tools) Database Storage page to include the following information: <ul style="list-style-type: none"> <li>• Queries Executed to Build Download</li> <li>• Summary of Queries Executed to Build Download</li> </ul>
PL-10000	Guidewire has added two additional columns to the Index Physical Statistics tab of the (Server Tools) Database Storage Information page. The two new columns are <i>fill_factor</i> and <i>is_padded</i> .
PL-10110	Guidewire has modified the (Server Tools) Work Queue Info page to show additional worker summary information, including throughput information.

ID	Description
<b>Performance</b>	
PC-17	Improved performance of the consistency check to ensure that no overlapping coverages exist.
PC-7553	Improved performance when checking the availability of a question set.
PC-10905	Reduced the number of queries to the database by using a single query to return a result size and the first result. In some instances, this was done using two queries.
<b>Product model</b>	
PC-10731	Fixed an issue in Studio where removing a coverage symbol group did not remove it from the underlying XML file. The user received the message: <i>Error: Coverage symbol group with code XXX does not exist.</i>
PC-10757	Improved performance of product model synchronization when synchronizing a large number of questions.
PC-11072	Fixed an issue with covterm availability. PolicyCenter displayed a message: <i>&lt;covterm&gt; is no longer available and has been removed.</i> Although the covterm was no longer available, the Coverages tab displayed a drop-down menu without a label for the covterm. This occurred for covterms that used a custom covterm input set. (Custom covterm input sets are used when a coverage or covterm has display special requirements that aren't handled by the standard input and iterators.)
PC-7553	Fixed an issue to eliminate unnecessary checking for question availability. This was previously logged as known issue CC-57211.
PC-7640	When writing an availability script for a question, you can now directly access PolicyPeriod, PolicyLocation, or PolicyLine, depending upon the answer container for the question set.
PC-7960	Fixed an issue in the product definition in Product Model → Products in Studio. On the Offerings → Selections tab, the tree nodes now display the pattern code in addition to the pattern name. There are often patterns with similar or identical names, and the code helps to differentiate them.
PC-8272	Fixed an issue with availability rows within Studio not duplicating properly.
PC-9017	Fixed an issue to improve the loading time of question sets by adding new synchronization method. This was previously logged as known issue CC-57134. To improve PolicyCenter performance when synchronizing question sets, the following changes were made: <ul style="list-style-type: none"><li>• Added a method that allows you to synchronize answers for specified question sets. This gives you more granular control over what is being synchronized. This method has the following signature: <code>syncQuestions(answerContainers : AnswerContainer[], questionSets : QuestionSet[], helper : JobWizardHelper)</code></li><li>• Opened access to the <code>getQuestionSets</code> method on <code>AnswerContainer</code>.</li></ul>
<b>Reporting</b>	
PC-10841	In the <code>sree.properties</code> file, the Oracle and SQLServer JAR files were missing from the scheduler classpath. Both <code>ojdbc6.jar</code> and <code>sqljdbc4.jar</code> have been added to <code>schedule.classpath</code> in <code>sree.properties</code> .
PL-11010	Guidewire has modified the default behavior of <code>sree.properties</code> configuration parameter: <code>dependency.checker.enabled</code> The default is now <code>false</code> . This disables the InetSoft dependency finder. If you set this parameter to <code>true</code> , it can significantly impact system performance.
<b>Security</b>	
PC-7373	Fixed an issue where an internal user was incorrectly allowed to see producer codes outside their security zones. This was previously logged as known issue CC-56955.
PC-8117	Added permissions to the <code>Tools View</code> role. These permissions start with <code>tools</code> and end with <code>view</code> .
PC-10797	Added two users, <code>Renewal Daemon</code> and <code>PolicyChange Daemon</code> , to the system bootstrap data. The <code>renewal_daemon</code> and <code>policychange_daemon</code> users are now in the <code>bootstrap.xml</code> file.
<b>Searching</b>	
PC-10854	Fixed an issue in <code>Search → Search Policies</code> where the <code>Search Results</code> displayed multiple instances of the same job.
PC-11038	Fixed an issue in the default configuration where <code>Search → Account</code> only found accounts which had the primary language set to <code>en_US</code> . This fix modified <code>search-config.xml</code> by removing the line that included primary language as a search criteria.
<b>Studio</b>	

<b>ID</b>	<b>Description</b>
PL-9774	Guidewire has added the ability to change the root entity of a rule. To do so, select a rule set in Studio, right-click, and select Change Root Entity from the drop-down menu.
PL-10020	Guidewire has corrected an issue in which the Studio Rules editor did not properly escape double quote marks in the rule description. This caused rules to not behave as expected and caused Studio to not start upon attempting to restart it.
PL-10185	Guidewire has corrected a Studio issue with searching that threw an error if you were searching within a rule set containing CVS or Subversion files. This issue only occurred if you searched for text or names within the rule resources, not with any other Studio resources.
PL-10293	Guidewire has corrected an issue that caused a Null Pointer Error if you double-clicked a view tab to maximize it.
PL-10329	Corrected a Studio search option that did not work correctly if searching from the lowest node in the rule set hierarchy. You can no longer search on Rule Set View in the global replace dialog.
PL-10333	Guidewire has corrected an issue with the Revert to Base right-click command on PCF folders deleting the base configuration folder.
PL-10357	Corrected an issue that prevented Studio SmartHelp from providing a selection list for Group or User ID if creating assignment rules.
PL-10363	Corrected an issue in which the Studio global replace functionality did not work correctly within rule set categories, generating an error message.
PL-10369	Corrected an issue in which Studio did not properly recognize the VALUE keyword conversionExpression attribute (in a PCF page) causing Studio to display an error message.
PC-7550	Removed the following XML files from Studio because PolicyCenter does not use them: <ul style="list-style-type: none"> <li>• Currencies.xml</li> <li>• Contact-config.xml</li> <li>• contact-sync-config.xml</li> </ul>
PC-11247	After the value was removed from the referenced policy line, PolicyCenter displayed <none selected> from the drop-down menu for the default value of an OptionCovTermPattern or PackageCovTermPattern in an offering. PolicyCenter now displays the removed value in red. The user can select another value. If you do not select another value, PolicyCenter displays an error on verification.
PC-11279	The difference tree XML configuration files now appear in the Resources pane in Studio. Expand Resources → configuration → Other Resources → difftree to view these XML files.
<b>Team support</b>	
PC-7353	Fixed an issue where you could not reassign an activity that did not have an owner. The superuser (su) can now reassign an activity.
PC-11062	Fixed an issue on the Team tab. The columns indicate that the display has the following time constraints: <ul style="list-style-type: none"> <li>• Activities (past day)</li> <li>• Submissions by User (this week)</li> <li>• Renewals by User (this month)</li> <li>• Other Work Orders by User (this week)</li> </ul> Previously, the queries behind the tabs did not include the time constraints, and returned all items.
<b>Tools</b>	
PC-11431	Fixed an issue where CONCAT, LOC, and ORDER BY SQLs in custom data distributions failed with exceptions in SQLServer.
<b>UI configuration</b>	
PC-10896	Fixed an issue where setting Job.CloseDate did not set CloseDate on all its assignables.
PC-11035	Modified TabBar.pcf located in Page Configuration (PCF) in Studio. Removed the explicit child MenuItems on the Desktop, Search, and Admin tabs because the menu should be automatically generated. The tab menu is now automatically generated from the LocationGroup items in the Desktop, Search, and Admin PCF files.
PC-11133	Improved the performance of displaying the Producer Code Search screen. This required a change to ProducerCodeSearchDV.pcf located in Page Configuration (PCF) → search.
PC-11248	Fixed a performance issue with the Underwriting Company drop-down menu in the Search Policies screen.
<b>Underwriting authority</b>	
PC-10894	Fixed a problem with underwriting issues that block bind. If the underwriting issue was approved as valid with no editing allowed, the issue was not reopened upon editing a renewal or a policy change.

ID	Description
PC-10983	Fixed an issue where links to jobs in the issue history were broken if a job was preempted.
PC-10984	Modified how PolicyCenter handles underwriting issues in preempted jobs. For more information, see "Handling Underwriting Issues in Preempted Jobs" in the <i>Application Guide</i> .
PC-11378	The TinySampleUWAuthorityData Gosu class is now available. It is an editable class.
PC-20	The <b>Through</b> column for Issue histories selects a display key for each blocking point, expressed as <i>through</i> the previous blocking point. When an issue is created with a blocking point of <b>BlocksQuote</b> , the through column is empty because an issue blocking quote is not yet approved <i>through</i> anything.
PC-7829	When displaying authority grants on the Admin screens, currency amounts are now formatted as currency. Previously, amounts were displayed as 1000000.0000 instead of \$1,000,000.
<b>Upgrade</b>	
PC-7183	Improved performance of internal database consistency checks.
<b>User tools</b>	
PC-9167	Fixed an issue where list view printing could not be configured within job wizards.
<b>Web</b>	
PL-10130	Guidewire has corrected an issue in which a multi-select input did not correctly display the selected items in read-only mode.

## 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 if they 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.

### WebSphere timeout during initial startup (CC-4904)

- **Issue:** The initial creation of the PolicyCenter database may take longer than the default WebSphere startup timeouts will allow, and WebSphere may abort the startup before the database upgrade has completed. In this situation, the `Default_Server_stderr.log` file will show an interruption in the `Applying upgrade step #### of #####` log messages.
- **Workaround:** Increase the Ping Timeout and Ping Initial Timeout values for WebSphere. These can be found in the WSADMIN Console; click **Nodes**, drill down to your application server, click the **Advanced** tab, and increase the values.

### On AIX without an X11 server, PolicyCenter may report that an X11 resource is missing (CC-5889)

- **Issue:** PolicyCenter does not require an X11 server; however other components of the PolicyCenter environment (such as Java) may inadvertently have this requirement. If so, PolicyCenter may report that an X11 resource is missing when it is starting up.
- **Workaround:** Install an X11 server, or contact Guidewire support to determine if the dependency can be removed.

### Port conflict when starting PolicyCenter with JMX RMI adapter enabled and Microsoft Outlook running (CC-6964)

- **Issue:** When Microsoft Outlook starts, it chooses several ports to use for its connection to an Exchange server. In some instances, Outlook may use port 1099, which is a port that the PolicyCenter JMX RMI adapter can use, *if enabled*. The PolicyCenter server will then report a port conflict during startup when it tries to use that port for the RMI adapter.
- **Workaround:** Close Outlook to free the port, and then start the PolicyCenter server again. You can also change the port that the RMI adapter uses by editing the `JMXManagementPlugin` through Guidewire Studio.

**Duplicate contact when merging contacts through API (PC-7421)**

- **Issue:** When you merge two instances of a contact through the API, you may get duplicate contacts for the following contact roles:
  - PolicySecNamedInsured
  - PolicyPriNamedInsured
  - PolicyBillingContact

This was previously logged as CC-56177.

- **Status:** Guidewire is aware of this issue.

**Studio: “Verify Changed Resources” is not implemented in the product model (PC-6928)**

- **Issue:** The Studio Tools → Verify → Verify Changed Resources command does not work with the PolicyCenter product model.
- **Workaround:** Use one of the other verification commands instead. See “Validating Studio Resources” in the *PolicyCenter Configuration Guide* for details.

**Zone data with missing values cannot be loaded (PL-9123)**

- **Issue:** Zone data, which is used for associating postal codes with cities and regions, does not load if it is incomplete.
- **Workaround:** As a temporary workaround, you can enter a character into that field, for example, a hyphen (-).

**Pre-update rules for extension entities (PL-8469)**

- **Issue:** In configurations where the parameter UseOldStylePreUpdate is set to TRUE, pre-update rules cannot be applied to entities that cannot be validated.
- **Workaround:** Add `<implementsEntity name="Validatable"/>` to the definitions of all entities that the pre-update rules manipulate.

**Need a summary of underwriting issues across all policies in a job group (PC-9102)**

- **Issue:** The JobGroupRiskAnalysisLV needs to show a summary of underwriting issues across all policies in the job group. This was previously logged as CC-54177.
- **Status:** Guidewire is aware of this issue.

**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. This was previously logged as CC-54957.
- **Status:** Guidewire is aware of this issue.

**DFS exception (PL-11044)**

PolicyCenter generates a DFS exception if the server resources change while something is being processed. There are basically two ways to get this to happen.

- **Issue 1:** If you are running in a shared environment, and someone changes a configuration resource, other people using the server get the DFS exception.
- **Workaround 1:** To avoid this problem, set the `ResourcesMutable` flag to `false` in `config.xml` located in **Other Resources** in Studio.
- **Issue 2:** If you are developing on the same system where the server is running, the DFS exception occurs if you have an editable page open in PolicyCenter when you make a resource change. In PolicyCenter, essentially every wizard step is an editable page with an uncommitted bundle, so it is not unusual to be on a wizard step while you are changing resources. When you try to move to another wizard step (or click **Save Draft**), PolicyCenter attempts to commit the bundle, and you get the DFS exception. You can get the DFS exception even without reloading PCF files using **ALT-SHIFT-L** in PolicyCenter. You can get the DFS exception if you change some Gosu in Studio, and click **Next** in the wizard.

- **Workaround 2:** Navigate outside the wizard (to the Desktop tab, for example). Clear the unsaved work cache by clicking the trash icon in the **Unsaved Work** list. Then, go back into the wizard, and resume your work. Clearing out the unsaved work cache is important because if you don't, then it will just try to commit that invalid bundle again the next time you click **Next**. But you do not have to log out and you do not have to restart the server.

#### EvaluationException: [251861] in Account tab (PC-7861)

- **Issue:** Receive the following exception when creating a new submission from a newly created company account:  

```
EvaluationException: [251861] Cannot load a temporary id from the backing entity source: Account
```

- **Workaround:** In Studio, navigate to **Page Configuration (PCF) → account → new**. Click to open **CreateAccount.pcf**, and select the **Code** tab. Move the first line of the **beforeCommitNewAccount()** method (which calls **setLastProducerSelection**) to be the first line of the **afterCommitNewAccount()** method.

#### Limitation in reporting when a typecode name contains a comma (CLM-8019)

- **Issue:** Due to the limitations of the InetSoft reporting software, issues occur when passing a typelist as a parameter to a drill-down report if one of its typecode names contains a comma. (For example, typelist *Loss Cause* contains a typecode with the name *Fall, slip, or trip injury*.) If a typecode name does contain a comma, InetSoft treats it as an array and sends multiple values. Guidewire has implemented a solution for the *Loss Cause* typelist in PolicyCenter. However, if you are passing any other typelist as a parameter to a drill-down report and it contains a typecode with a comma, you will experience this issue.
- **Workaround:** Guidewire is aware of this issue. If you encounter this issue with any other report parameters, contact Guidewire Support for assistance.

#### Duplicate descriptions for image/JPG mime types in the configuration file (CLM-14188)

- **Issue:** In the **config.xml** file, there are two entries under MIME types for jpeg: one for pjpeg and one for normal jpgs. This causes a problem as the descriptions of both are identical. If you upload a normal jpeg and chooses the wrong jpeg MIME type, an error occurs.
- **Workaround:** Change the description of the pjpeg so that duplicate entries do not appear in the MIME type drop down menu in Guidewire Studio.

#### PolicyCenter login remains active in new browser window (CC-4152)

- **Issue:** When you log into PolicyCenter in a browser window, then you close that window and open a new one, in some instances you can access PolicyCenter pages without needing to log in again.

This is an issue with some versions of Microsoft Internet Explorer. When multiple browser windows are running in the same process, they can share session information, and this allows them to share a single login with PolicyCenter. Early versions of Internet Explorer provided a **Launch browser windows in a separate process** option, which, if enabled, would avoid this situation. Some versions of Internet Explorer do not allow you to set this option, and enable or disable it automatically. You can review the relevant Microsoft Knowledge Base Article 240928. However, you may experience a situation in which Internet Explorer does not correctly enable this option, especially if you manually disabled it in an earlier version and then later upgraded. In this case, the multiple browser windows are run in the same process, exhibiting the undesired behavior.

- **Workaround:** The simplest solution is to always click **Logout** at the top of the PolicyCenter window before closing your browser.

Optionally, you can take steps to manually enable the Internet Explorer option. This requires you to edit the Windows registry on the affected computer.

---

**WARNING** Editing the registry is a risky procedure, and any improper changes may result in permanent data loss and require Microsoft Windows to be completely reinstalled on that computer. To make this change, create or edit the registry entry:

---

[HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\BrowseNewProcess]

Set the data type to REC\_SZ, and then set the data value to yes.

**Guidewire Studio PCF Editor treats commented-out property values as errors (PL-4582)**

- **Issue:** If you attempt to comment out a property value in the Studio PCF editor, Studio treats this as an error.
- **Status:** None. Guidewire is aware of this issue.

**Server exception during startup (PL-8167)**

- **Issue:** During server startup, there can be a Tomcat server exception. However, after the exception, the server does start up. This is caused by the Tomcat application server attempting to either save or restore sessions using serialization. The session restore causes an exception because PolicyCenter sessions contain objects which rely on the metadata being started up.
- **Workaround:** None. The Tomcat application server does start up after the error.

**H2 development database creates LONGTEXT fields as VARCHAR (65000), instead of as CLOB as in Oracle and SQL Server (PL-9314)**

- **Issue:** Guidewire creates schema fields defined as LONGTEXT as LONGVARCHAR columns, instead as CLOB as in the Oracle or SQL Server databases. In addition to the inconsistency involved, this mandates a field length of 65,000 characters or less in LONGTEXT columns in the H2 database. Note that the H2 database is *only* used for testing and should never be used in production.
- **Workaround:** Define a LONGTEXT field as a VARCHAR( . . . ) with the required size, for example, as VARCHAR(120000).

**PolicyCenter does not display empty report folders (PL-1281)**

- **Issue:** PolicyCenter does not display a report folder in the Administration → Report Admin page if that report folder is empty.
- **Workaround:** Within InetSoft Enterprise Manager, insert a *replet* (report template) into the empty report folder. You can then mark this *replet* as not visible (using the InetSoft Enterprise Manager), which causes the *replet* to not show in PolicyCenter. However, PolicyCenter does display the parent folder.

**Tab widgets do not work with Microsoft Windows Server 2003 enhanced security enabled (PL-9813)**

- **Issue:** If you have the Windows Server 2003 component called IE Enhanced Security Feature installed, tabs may be unresponsive in the user interface.
- **Workaround:** Uninstall this component from the Windows 2003 Server.

**Country specific field validation does not run if the fields are not modified when a country is changed (PL-9828)**

- **Issue:** There may be an error if you change a contact field in the user interface that is validated with country specific field validators. Those specific validators do not get validated again if the country changes and the field is not modified. This issue only occurs if you use country specific field validators.

For example, suppose that a carrier operates in two countries, the United States and Canada and wants to implement different field validation for certain data types such as Tax ID. If an adjuster changes the country on an existing contact record from the United States to Canada, without modifying the Tax ID to reflect the new format, the system allows the US formatted Tax ID to be saved. If the adjuster edits the Tax ID, then the system validates the format again.

- **Status:** Guidewire is aware of this issue.

**Deleting a report folder from the Report Admin causes an exception (PL-10348)**

- **Issue:** If you delete a report folder from the Report Admin, users will not be able to access the Report Admin.
- **Status:** Guidewire is aware of this issue.

**Not possible to hot-deploy on all supported application servers (PL-9955)**

- **Issue:** It is not possible to hot-deploy on all supported application servers.
- **Workaround:** If you use Tomcat, you are able to see the changes without requiring a restart of the application server. If you use an other application server, you need to perform a restart to see any configuration changes reflected.

**Values in MIME-type drop-down menus are not localized (PL-10921)**

- **Issue:** PolicyCenter does not localize the values in the MIME type drop-down list properly.
- **Workaround:** If you want to localize MIME types, you can do so by modifying the definition in `config.xml`. Currently, PolicyCenter does not support MIME type definitions in multiple languages.

**Cancel button does not work properly (PL-10926)**

- **Issue:** The Cancel button does not work correctly if you start in edit mode by clicking on a button such as `Add` from read-only mode.
- **Workaround:** Click `Edit` to move the page into edit mode. In this case, all behavior is as expected.

**Finder in annotation causes throws error (PL-10812)**

- **Issue:** Studio throws error and fails to do syntax error highlight, code completion, and similar functions, if there is a finder in an annotation. This occurs even if you are connected to the server.
- **Workaround:** Remove the finder from the annotation.

**Navigating between pages in wizard loses data (PL-10920)**

- **Issue:** In wizard pop-ups, it is possible to lose data if you navigate to another page in the wizard.
- **Workaround:** Navigate out of the wizard and then back into it rather than changing steps directly in the wizard.

**Studio does not properly report verification error (PL-10822)**

- **Issue:** Studio does not properly report a verification error if a typecode contains a trailing space.
- **Workaround:** Remove trailing spaces from all type codes.

**Double quotes in PATH environment variable causes problems (PL-10981)**

- **Issue:** PolicyCenter does not run if there are double quotes in the PATH environment variable.
- **Workaround:** Check the PATH variable and remove any double quotes.

**Studio Debugger does not handle rule breakpoints properly (PL-10553)**

- **Issue:** If you set a breakpoint in a Gosu rule and attempt to step through the code using the Studio debugger, the debugger does not correctly display the code on the breakpoint step.
- **Status:** Guidewire is aware of this issue.

**Inability to access typecodes that contain a dash (PL-10532)**

- **Issue:** It is not possible to access a typekey code (typecode) that contains a dash using Gosu.
- **Workaround:** Remove any dashes from the typekey codes.

**Method `assignByUserAttributes` does not distinguish between attributes (PL-10607)**

- **Issue:** Method `assignByUserAttributes` does not distinguish between different attributes.
- **Workaround:** Ensure that you thoroughly test all assignment scenarios to ensure that assignment happens correctly.

**Cannot start a Spanish translation pack with the Oracle database (PL-11230)**

- **Issue:** If you set the application locale to a locale for which you did not configure database collations (in the `collations.xml` file), then the application fails on restart. If this is the case, then you see one of the following error messages (depending on your database):
  - `java.sql.SQLException`
  - `ORA-12705: Cannot access NLS data files or invalid environment specified`
- **Workaround:** Verify that you have properly configured database collations for a given locale before setting the application to that locale.

**Cache summary statistics rely on server ID being unique (PL-11239)**

- **Issue:** If you use the same server ID for multiple servers in a cluster, it can create the following problems:
  - Different servers in a cluster deleting or changing each others results
  - Deadlocks at the database level
- **Status:** Guidewire is aware of this issue.

# Guidewire PolicyCenter 4.0.3 Release Notes

## Release 4.0.3.11

Copyright © 2001-2010 Guidewire Software, Inc. All rights reserved. Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Insurance Suite, Guidewire ContactCenter, Guidewire Studio, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries. All other product names are trademarks of their respective companies.

**This documentation is published as Guidewire Confidential.** The contents of this documentation, including product architecture details and APIs, are considered confidential and are fully protected by customer licensing confidentiality agreements and signed Non-Disclosure Agreements (NDAs).

This product includes software developed by the Apache Software Foundation (<http://www.apache.org>).

## Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Changes in This Release
- Improvements and General Issues
- Known Issues and Limitations

## Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

## Version Number

- This release of Guidewire PolicyCenter is 4.0.3.11.
- If you are using the integration with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 3.0.1 or later.
- If you are using the integration with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 6.0.0 or later.
- For Standard Reporting, this release of PolicyCenter supports InetSoft StyleReport Enterprise Edition v10.1 20100113.

## Installation Requirements

For installation requirements, see the *PolicyCenter Installation Guide*.

## Support

This document includes important information about the current release of Guidewire PolicyCenter. For assistance with this software release, contact Guidewire Customer Support. You can contact support on the Web at <http://portal.guidewire.com>, by email at [support@guidewire.com](mailto:support@guidewire.com), or by phone at +1-650-356-4955.

## Installing This Release

For installation information, see the *PolicyCenter Installation Guide*.

This section describes the following changes to PolicyCenter:

- InetSoft Evaluation

## InetSoft Evaluation

You can evaluate InetSoft Style Report Enterprise Edition Version 10.1 using the following license key:

L000-798-ERX-0000BF00100001F-F2AFC9DD34ED

If you want to use InetSoft and the PolicyCenter reporting module in a production environment, contact Guidewire Customer Support to request a full license key. For more information, see the *PolicyCenter Reporting Guide*.

---

**WARNING** The build number of the reporting install files must match the build number of the Guidewire PolicyCenter installation. If the build numbers do not match, you will have problems with your reporting installation.

---

## Issues and Major Changes

This section contains issues or changes that may affect your installation.

### Clustered Indexes

In this release, Guidewire has implemented clustered indexes on the SQL Server database. The database upgrade automatically recreates non-clustered backing indexes for primary keys as clustered indexes. This change improves the performance of certain types of queries, and improves backup performance. Before recreating the indexes, the upgrade automatically drops (and later rebuilds) any referencing foreign keys and drops any clustered indexes on tables with a primary key.

Recreating the indexes will cause the database upgrade to take a substantial amount of time – many hours for a large database. Guidewire has tried to contact large SQL Server customer before making this change to ensure it will not cause them problems. But please contact customer support if you have not already been contacted and have a large SQL Server database you plan to upgrade to 6.0.2.

### Linguistic Search Performance

PolicyCenter uses an Oracle Java Stored Function to perform linguistic searches. Such search queries can perform poorly.

See PL-12143 for details.

## Changes in This Release

This section describes the product changes in this release.

- Base PCF File Changes
- Base Resources Changes

### Base PCF File Changes

All links below require the `readme_files` directory on your local disk.

#### PolicyCenter release 4.0.2 to 4.0.3

- To view a report of the changes in the base PCF files in the `modules/pc` directory, [click here](#).
- To view a report of the changes in the base PCF files in the `modules/p1` directory, [click here](#).

### Base Resources Changes

All links below require the `readme_files` directory on your local disk.

#### PolicyCenter release 4.0.2 to 4.0.3

- To view a report of the changes in the base resources in the `modules/bc` directory, [click here](#).

## Improvements and General Issues

This topic describes the primary improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This is not meant to be a comprehensive list. Many performance related issues were addressed in this release but are not listed here.

ID	Description
<b>Account</b>	
PC-11393	Improved performance in the <code>asViews</code> function in <code>gw.account.AccountContactArrayEnhancement</code> .
<b>Administration</b>	

ID	Description
PC-10524	<p>In <code>com.guidewire.pc.system.dependency.PCDependencies</code>, the <code>getUWCompanyFinder().findUWCompaniesByStatesAndProductAndValidOnDate()</code> method has been updated to improve performance. As a result of this performance improvement, the following changes were made:</p> <ul style="list-style-type: none"> <li><code>UWCompany.finder.findUWCompaniesByStatesAndProductAndValidOnDate()</code> assumes that there are not multiple <code>LicensedStates</code> for a <code>UWCompany</code> in a given State and ProductCode that have overlapping effective date ranges. In <code>gw.plugin.systable.verifier.LicenseStateVerifier.gs</code> enforces this.</li> <li>Segment is not a unique identifier for a <code>LicensedState</code>. If you wish to segment your underwriting company, create multiple <code>UWCompany</code> entities with their own <code>LicensedState</code> entries. The unique identifiers are <code>UWCompany</code>, <code>State</code>, and <code>ProductCode</code>.</li> <li>Editing <code>LicensedStateVerifier</code> or other system table verifiers by hand is not recommended, as these classes are expected to become <code>@ReadOnly</code> in a future release. If you need to make changes to a verifier, simply discard it and write your own. In <code>systables.xml</code>, replace the <code>Verifier</code> with your new class.</li> </ul>
<b>Contacts</b>	
PC-11373	Improved the performance of policy search on last name when last name has changed.
<b>Platform - Core</b>	
PL-10533	Fixed an issue with the <code>regen-pcfmapping</code> tool that caused it to not generate some display labels.
PL-10601	Corrected an issue that caused the <code>regen-toolkit</code> command to fail due to an issue with how the application handled third-party JAR files.
PL-10749	Provided the ability for extensions to support linguistic search on a base column by providing a <code>supportsLinguisticSearch</code> attribute on the <code>&lt;column-override&gt;</code> element.
PL-10761	Corrected a GosuDoc issue that generated a <code>NoClassDefFoundError</code> exception. Even if a <code>gosudoc.properties</code> file existed in the configuration module, the application did not read it.
PL-10896	Disabled the running of validation rules for importing <code>SystemTables.xml</code> and <code>bootstrap.xml</code> . Guidewire still runs validation rules on files in <code>config/import/gen</code> , which are modifiable.
PL-11001	Corrected an issue in which the <code>gw.api.util.Logger</code> utility did not consistently log the server name to the console and in the log file.
PL-11015	Created a new server mode <code>test</code> , that behaves exactly like a production server, except for the ability to modify the system time through an implementation of <code>ITestingClock</code> .
PL-11053	Dates with trailing characters were previously allowed and parsed with potentially the wrong year. Date strings with trailing characters are now rejected.
PL-11062	Moved <code>gw.plugin.util.RemotableSearchResultSpec</code> and <code>gw.plugin.util.RemotableSortColumn</code> out of <code>util</code> . This change means that they are no longer accessible in Java. Instead, for Java, use <code>com.guidewire.cc.plugin.util.RemotableSearchResultSpec</code> and <code>com.guidewire.cc.plugin.util.RemotableSearchColumn</code> .
PL-11068	In calling a Java plugin from Gosu code, the application did not automatically convert the domain entity class to an external entity class. For the application to handle this correctly, you need to copy the external entity jar ( <code>gw-entity-cc.jar</code> ) from: <code>java-api/lib/</code> to: <code>modules/configuration/plugins/shared/lib/</code>
PL-11077	Corrected an issue that caused a Runtime Exception if you did the following sequence of events: <ul style="list-style-type: none"> <li>Created a user validation rule that references a custom system permission.</li> <li>Dropped the application database.</li> <li>Started the application, which triggered a database upgrade.</li> </ul>
PL-11078	Guidewire officially deprecates library package <code>gw.api.domain.ServiceInfoSource</code> . In its place, use package library <code>gw.api.system.server.ServerUtil</code> .
PL-11136	Corrected an issue with the configuration upgrader that caused it to throw an exception and quit. This happened specifically if you were running an upgrade within a major release to a maintenance release.
PL-11162	Corrected an issue that caused the application to fail to resize an encrypted column properly if you changed the encryption plugin.
PL-11278	Modified the Data Dictionary so that Yes/No fields reflect the actual column type, which is <code>bit</code> (not Boolean).

ID	Description
PL-11283	Modified the Data Dictionary so that the display name for a foreign key is the <code>columnName</code> attribute for that column.
PL-11284	Changed the Data Dictionary to display the <code>columnName</code> attribute (not the name attribute), if a <code>&lt;column&gt;</code> element contained both. The dictionary now displays the name attribute if both exist.
PL-11285	Removed the set of links that existed underneath the Subtypes lists (if a data entity had subtypes). The links referenced Fields, Arrays, Subtypes, and Unique Keys. The list of Unique Keys was, indeed, not unique.
PL-11286	Corrected an issue that caused the Data Dictionary to display derived fields and arrays in inappropriate sections.
PL-11312	Corrected an issue that caused the Data Dictionary to reference custom entity database tables incorrectly.
PL-11549	Corrected an issue with the (Server Tools) Guidewire Profiler so that it now correctly displays the proper number of milliseconds. Previously, there were cases in which this value was incorrect.
PL-11634	Corrected an issue in which the application ignored the reject methods on validatable extension entities.
PL-11709	Added a histogram to the column <code>EditEffectiveDate</code> in <code>EffDatedBranch.eti</code> . PolicyCenter uses this specifically for the Job Expire ( <code>jobexpire</code> ) batch process.
PL-11739	Introduced a way to suppress conversion of containers (arrays, lists, maps) when calling Java code. This only is available if the classes do not use Guidewire entity instances (or containers that contain entity instances). It is also unavailable if the class implements a PolicyCenter plugin interface. To use this feature, underneath the <code>PolicyCenter/modules/configuration/plugins/shared</code> directory, create a new directory called <code>basic</code> . Put your libraries and classes in subdirectories <code>classes</code> and <code>lib</code> , just like other plugins directories.
PL-11819	Corrected an issue with the Studio debugger that caused an exception to occur during the logging of the SQL statements made by Gosu find expressions.
PL-11835	Modified the data model to allow extending a non-extendable entity provided that the only changes are to <code>validator</code> or <code>logicalSize</code> parameters using the <code>&lt;column-override&gt;</code> element. This usage replace the <code>&lt;FieldValidatorOverride&gt;</code> and <code>&lt;FieldLengthOverride&gt;</code> elements, which have been removed from <code>fieldvalidators.xml</code> .
PL-9041	Modified how the currency amount data type works. You no longer specify <code>secondaryAmountProperty</code> and <code>exchangeRateProperty</code> on columns of type <code>CurrencyAmount</code> . Instead, use the new <code>CurrencyAmountComponent</code> if you want a quantity that is editable in the application interface.
<b>Data Model</b>	
PC-11784	Fixed an issue which prevented you from allowing more than one user to have the same participant role on an entity. The base configuration does not allow more than one user to have the same participant role on an entity. However, you can now configure this in <code>AssignmentUtil.filterAssignableRoles</code> .
PC-11920	Fixed an issue where using <code>PolicyPeriodVersionList</code> caused data corruption on arrays accessed from it.
<b>Desktop</b>	
PC-11382	Fixed an issue where PolicyCenter displayed errors in the <code>Search → Search Policies</code> screen. This occurred if <code>Search Results</code> were displayed, and you changed the selection in the <code>Search For</code> pull-down menu.
<b>Financials</b>	
PC-12169	Fixed an issue where the transaction calculator created full offsets in cases where a partial offset should be allowed. This caused a problem where extra onset and offset transactions were generated.
PC-12198	In addition to <b>Quote Rounding Level</b> , you can now set <b>Quote Rounding Mode</b> in the product model editor in Studio. In the base application, PolicyCenter uses this rounding mode in the rating engine and transaction calculator. If you have integrated your own rating engine, set this field in the product model to match the mode of your rating engine. Then the transaction calculator will use the same rounding mode as your rating engine.  The values for <b>Quote Round Mode</b> are: <ul style="list-style-type: none"><li>• <b>UP</b>—Round up.</li><li>• <b>DOWN</b>—Round down.</li><li>• <b>CEILING</b>—Round up.</li><li>• <b>FLOOR</b>—Round down.</li><li>• <b>HALF_UP</b>—Round to the nearest. If exactly halfway, round up (for example, round 2.5000 to 3.0000).</li><li>• <b>HALF_DOWN</b>—Round to the nearest. If exact halfway, round down (for example, round 2.5000 to 2.0000).</li><li>• <b>HALF_EVEN</b>—Round to the nearest. If exactly halfway, round toward the nearest even number (for example, round 2.5000 to 2.0000, but round 3.5000 to 4.0000).</li><li>• <b>UNNECESSARY</b>—No rounding is necessary because the value returned is exact.</li></ul>

ID	Description
<b>Gosu</b>	
PL-10778, PL-12585	This change fixes a bug that was introduced in Gosu in PolicyCenter 4.0.2 that caused the arguments of a method call to be evaluated before the object target expression of the method call. In nearly all cases this causes no change in behavior. In extremely rare cases, if there were side effects of the argument expressions that change the object target expression value, the behavior is now consistent with the original semantics.
PL-11044	Added a new server configuration option, called <code>RetainDebugInfo</code> , to facilitate debugging from Studio without a type system refresh. If set to true, <code>GosuClasses</code> does not clear debug information after compilation and the server is always considered to be debuggable.
PC-11793	Modified a number of Gosu classes that override the <code>equals</code> function. These now have a corresponding override of the <code>hashCode</code> function.  In Java and Gosu programming, if you override the <code>equals</code> function, you should also make a corresponding override to the <code>hashCode</code> function. The code has to be written to guarantee that if <code>a.equals(b)</code> then <code>a.hashCode() == b.hashCode()</code> . Otherwise the object cannot be used as a key in <code>Hashtable</code> or <code>HashMap</code> or as a member of a <code>HashSet</code> .
<b>Platform - Integration</b>	
PL-10503	Updated the <code>Archive</code> batch process name and description to reflect the new archiving framework.
PL-10599	It is possible that you can experience namespace collisions in the names of argument types and return types on published web services. (If this occurs, Studio notifies you with errors starting in PolicyCenter 4.0.0.) To make debugging during upgrade easier, PolicyCenter 4.0.3 has a new configuration parameter called <code>AllowSoapWebServiceReferenceNamespaceCollisions</code> . If set to true, these error messages become warnings. Use this for development and debugging until you have time to rename your classes to fix the namespace collision. This setting is false by default.  Guidewire does not support setting this value to true for production servers. (It is unsafe to do so.)  Please contact Guidewire Customer Support for complete details.
PL-11193	Guidewire has added a <code>UseMessageCreatorAsUser</code> property to the <code>emailMessageTransport</code> plugin. By default, the <code>emailMessageTransport</code> plugin will use system user to retrieve a document. Set this value to true to retrieve the document on behalf of the user who generated the email message.
PL-11221	Corrected an issue with the <code>IMessagingToolsAPI</code> API that occurred while attempting to acknowledge messages using web services. This issue occurred if the number of field changes was bigger than the smallest number of field values in each field change.
PL-11548	Corrected an issue with adding more than one document to a <code>ReserveSet</code> if using the <code>IDocumentMetadata</code> plugin. For example, any attempt to add multiple documents to a claim on the <code>New Reserve</code> screen would fail.
PL-11677	Fixed a subtle bug in how messaging destinations reads new messages from the messaging queue and dispatches them to messaging plugins. In a rare edge case, a non-claim-specific message could enter the send queue before the related claim-specific messages because of the separate database queries for these two categories of messages. Starting in this release, in each round of sending, each messaging destination always reads claim-specific messages (safe-ordered messages) from the queue before non-claim-specific messages. (Although the read order changed, the dispatch order to messaging plugins did not change. As before, messaging destination threads send non-claim-specific messages before claim-specific messages.)

ID	Description
PL-12689	<p>Modified the list of file types that PolicyCenter opens automatically. Support has been added for the following file types:</p> <ul style="list-style-type: none"> <li>• .docx application/vnd.openxmlformats-officedocument.wordprocessingml.document</li> <li>• .xlsx application/vnd.openxmlformats-officedocument.spreadsheetml.sheet</li> <li>• .pptx application/vnd.openxmlformats-officedocument.presentationml.presentation</li> <li>• .ppt additional extension for application/vnd.ms-powerpoint</li> <li>• .pps application/vnd.ms-powerpoint</li> <li>• .ps application/postscript</li> <li>• .rtx text/richtext</li> <li>• .wav audio/wav</li> <li>• .wma audio/x-ms-wma</li> <li>• .mdi image/vnd.ms-modi</li> <li>• .gif image/gif</li> <li>• .png image/x-png</li> <li>• .mov video/quicktime</li> <li>• .mpeg video/mpeg</li> <li>• .mpg video/mpeg (not returned)</li> <li>• .avi video/x-msvideo</li> </ul> <p>Since .avi and .wma files can contain security vulnerabilities, Guidewire strongly recommends that you update all client machines with video-playing software that contains the latest security patches. For specific information, see Microsoft web pages <a href="http://www.microsoft.com/technet/security/Bulletin/MS09-038.mspx">.avi files vulnerability</a> (<a href="http://www.microsoft.com/technet/security/Bulletin/MS09-038.mspx">http://www.microsoft.com/technet/security/Bulletin/MS09-038.mspx</a>) or <a href="http://www.microsoft.com/technet/security/bulletin/ms09-051.mspx">.wma file vulnerability</a> (<a href="http://www.microsoft.com/technet/security/bulletin/ms09-051.mspx">http://www.microsoft.com/technet/security/bulletin/ms09-051.mspx</a>).</p>
<b>Integration with BillingCenter or a billing system</b>	
PC-11729	Fixed an issue where PolicyCenter did not send a message to BillingCenter to open a locked period when PolicyCenter removed the final audit at flat cancellation. During a flat cancellation, PolicyCenter removes the scheduled final audit. Now PolicyCenter sends a message to BillingCenter to reset the Open Locked closure status to Open and the status of Require Final Audit is also set to No.
<b>Integration with a contact management system</b>	
PC-11987	Fixed an issue where IAccountPlugin.transferPolicies() failed to create AccountContacts to match PolicyContacts.
<b>Job infrastructure</b>	
PC-11444	Fixed an issue with EffDatedBeans in difference trees. The EffDatedBeans are sometimes returned in window mode and not slice mode. These EffDatedBeans are now returned in slice mode.
PC-11698	Fixed an issue where an edited submission had a status of withdrawn instead of draft. This occurred in a Quick Quote submission with multiple versions.
PC-8057	Fixed an issue where diffs did not display window changes in Policy Review.
<b>Jobs</b>	
PC-11441	Fixed an issue where diffs with a <RootEntity> and <SubSection> did not appear in the Policy Review screen.
PC-11799	Fixed an issue with rewrite jobs after quoting. Previously, you were able to edit the policy information after quoting the job. Now you can no longer edit the policy information after quoting.
<b>Platform - Manageability</b>	
PL-10035	Modified the Server Tools → Data Distribution page and added several new non-Boolean types to track the size of a group of values and the number of groups with that size.
PL-10984	Modified the Server Tools → Info Pages → Data Distribution screen to ensure that the user selects at least two data distributions to include in the download.
PL-11038	Corrected a caching issue with the Sequence Generator that caused it to work incorrectly.
PL-11127	Modified the Server Tools → Data Distribution page to disable the Download Zip File button if no background data distribution batch process is running.
PL-11153	Added a new download to the System Tools → Work Queue page. This download provides a zipped CSV file with the history of a given work queue (timing, completed items count, and throughput).
PL-11244	Added a new Server Tools → Database Statistics screen that you can use to view previous database statistics based on a given date.

ID	Description
PL-11427	Added a new <code>sqlserverCreateIndexSortInTempDB</code> attribute to the <code>&lt;upgrade&gt;</code> element of the <code>&lt;database&gt;</code> element in the <code>config.xml</code> file. The default is <code>false</code> . If true, the attribute enables the ability to use <code>tempdb</code> for sorting if building indexes in SQL Server. Refer to <a href="http://msdn.microsoft.com/en-us/library/ms188281.aspx">http://msdn.microsoft.com/en-us/library/ms188281.aspx</a> for details on the requirements to use <code>tempdb</code> for sort results.
<b>Performance</b>	
PL-10874	Modified how the application handles caching of zone data. Previously, large amounts of data would create a severe performance bottleneck and had the potential to crash the application server. This performance fix greatly mitigates these issues.
PL-11431	Converted all backing indexes for all primary keys to clustered indexes in SQL Server. The change improves performance of archiving and purge operations. During a database upgrade, the upgrader automatically converts the backing indexes for all primary keys to clustered indexes and drops any clustered indexes on tables with a primary key.
PL-12257	Guidewire has modified how the application handles caching of zone data. Previously, large amounts of data would create a severe performance bottleneck and had the potential to crash the application server. This performance fix greatly mitigates these issues.
<b>Policy file</b>	
PC-11844	Fixed an issue in <b>Search Policies</b> where the <b>As of Date</b> returned incorrect results. Previously, if the policy became effective on the <b>As of Date</b> , that policy was not in the search results. If the policy expired on the <b>As of Date</b> , it was returned in the search results.
<b>Reporting</b>	
PL-10629	Guidewire has modified the way it supports InetSoft VPM. This work has tightened the Guidewire SOAP security for <code>ISREEAAuthenticationAPI</code> and added three more properties to <code>sree.properties</code> . <ul style="list-style-type: none"> <li>• <code>gw.soapuser.username</code>: The user name used by InetSoft to retrieve user/group/role information used by VPM and authentication</li> <li>• <code>gw.soapuser.userpassword</code>: The password associated with the user name.</li> <li>• <code>gw.usercache.size</code>: The cache size for user cache, group cache, and role cache.</li> </ul>
PL-10629	Guidewire has modified the way it supports InetSoft VPM. This work has tightened the Guidewire SOAP security for <code>ISREEAAuthenticationAPI</code> and added three more properties to <code>sree.properties</code> . <ul style="list-style-type: none"> <li>• <code>gw.soapuser.username</code>: The user name used by InetSoft to retrieve user/group/role information used by VPM and authentication</li> <li>• <code>gw.soapuser.userpassword</code>: The password associated with the user name.</li> <li>• <code>gw.usercache.size</code>: The cache size for user cache, group cache, and role cache.</li> </ul>
PL-10972	The application server now logs an error whenever its connection status to the InetSoft report server changes.
PL-10972	The application server now logs an error whenever its connection status to the InetSoft report server changes.
PL-11010	Modified the default behavior of <code>sree.properties</code> configuration parameter <code>dependency.checker.enabled</code> . The default is now <code>false</code> . This disables the InetSoft dependency finder. If you set this parameter to <code>true</code> , it can significantly impact system performance.
PL-11375	Corrected a security issue with InetSoft reports. The fix included using username credentials instead of client information for principal creation.
<b>Revisioning</b>	
PC-11362	Fixed an issue where out-of-sequence policy changes caused duplicate modifiers to be created. The matchers that prevent these duplicates are now implemented for all standard modifiers and can be used as examples for additional modifiers added by customers.
PC-11372	Fixed an issue where PolicyCenter did not always generate a unique <code>TermNumber</code> for each <code>PolicyTerm</code> in a <code>Policy</code> . Now PolicyCenter always generates a unique <code>TermNumber</code> .
<b>Security</b>	
PC-11334	Fixed an issue in the Security Dictionary where permissions on objects (such as Account view) were labelled <code>static</code> . Permissions on objects are now labelled <code>object-based</code> .
<b>Searching</b>	
PC-11417	Improved performance on <code>PolicySearch</code> for <code>First</code> , <code>Last</code> , and <code>Company</code> names in bound <code>PolicyPeriods</code> .

ID	Description
PC-11696	Added a PolicySearchMaxResult parameter that sets the maximum number of results that a policy search returns.
<b>Studio</b>	
PL-10380	Modified the behavior of the Messaging editor so that the Remove button is unavailable if there is only a single defined messaging environment.
PL-10442	Corrected an issue that prevented Studio from moving a PCF file between folders if its name was not also changed.
PL-10447	Corrected an issue that caused a ClassNotFoundException (on AbstractFireTest) upon trying to start the GUnit Debug utility.
PL-10455	Corrected an issue in which using CTRL+left/right arrow to move the cursor from one word to another treated characters accents, cedillas, umlauts, and other modifiers as word breaks (in the same way it treats spaces). This happened with é, ç, and è, for example.
PL-10530	Added the ability to manage certain CSV (comma separated value) files within Studio. (These files are in the <i>Other Resources</i> folder and include authority-limits.csv, activity-patterns.csv, reportgroups.csv, and similar files.) Studio manages the file (adding an editable copy to the configuration folder, for example). You must still, however, edit these files in an external editor.
PL-10570	Corrected an issue that caused an Illegal ordering found for element InputFooterSection error. This occurred if you added an InputColumn widget to the far right of an existing InputColumn widget in a PCF containing an InputFooterSection and tried to save the change.
PL-10598	Modified the behavior of the PCF TypeKeyInput/RangeInput widget's filter so that it now recognizes VALUES as a valid value.
PL-10605	Modified the behavior of the Studio Entity Names editor so that it now properly displays entities with displayName columns, and makes the symbols used in the Entity Name Gosu editor into string types for displayName=true columns.
PL-10698	Modified the behavior of the Create Display Key dialog so that it now wraps text automatically as needed.
PL-10728	Corrected an issue that caused a ClassNotFoundException error if you attempted to revert a newly created PCF file within source control using the (Studio) File → Source Control → Revert menu.
PL-10734	Modified the Rule Conditions editor to display an error if the condition is not a boolean expression. (For example, an assignment statement such as account.AccountType = "payer" is not a boolean expression and should display an error.)
PL-10806	Modified the behavior of the Go-To (CTRL-N) pop-up dialog so that you can now search on workflow names, entity names, and plugin names.
PL-10835	Corrected an issue in which Studio did not display a warning on the use of improper case in Gosu templates. This occurred, for example, if you defined a variable with an initial upper-case character and then you later used that variable with all lower-case characters.
PL-10919	Modified the (Studio) Messaging editor to disable the Add button if the Env field is empty (meaning no name exists in the Env field).
PL-11023	Corrected an issue with typing text in the (Studio) New Workflow Subtype Metadata dialog.
PL-11024	Corrected an issue that permitted recursive screen references on a PCF page, causing a stack overflow.
PL-11137	Modified the extension file that Studio automatically generates when you select an entity .eti file, right-click, and select the Create extension file option. Studio now creates separate <extension> and </extension> tags, making the file easier to modify.
PL-11195	Corrected an issue that caused Studio to truncate the display key text used to generate widget titles in PCF files.
PL-11225	Updated Studio so that the (File) Import and Export submenus are correctly localized, if you install a localization module.
PL-11339	Corrected a timing issue that required you click OK twice to close the Display Key Editor dialog that opened from the PCF editor.
PL-11345	Corrected an issue in which the GUnit tester failed (and generated an error) if you attempted to reinvoke a test from the Result window.

**Studio - Product Model**

ID	Description
PC-11639	Fixed an issue in the Forms editor for policy lines. Previously, when adding a form by clicking <b>Add or Duplicate</b> , the editor required that the <b>Form Number</b> be unique. This was incorrect since many form patterns have the same form number with a different <b>Edition</b> . Now only the form <b>Code</b> must be unique.
<b>Team support</b>	
PC-11634	Fixed an issue on the <b>Team</b> tab where automatically generated activities did not appear in the list of activities.
<b>Tools</b>	
PC-11862	Fixed an issue where the <code>maintenance_tools -startprocess</code> command only picked up batch processes from the default configuration. It did not pick up custom batch processes that you added.
<b>Underwriting authority</b>	
PC-11834	Fixed an issue where underwriting issues were created for slices on and after the cancellation date. Now, no underwriting issues are created on these slices.
PC-12236	Fixed an issue where PolicyCenter reported the following slice error after multiple policy changes with UWIssues: <code>IllegalArgumentException: Slice date...."</code>
<b>Platform - Web</b>	
PL-10438	Corrected an exception if you revisited an uncommon search page whose search result is not <code>QueryResult</code> type.
PL-10476	Added the ability to generate a new row in <code>ListView</code> edit mode by pressing <b>ENTER</b> in the last editable cell.
PL-10706	Corrected an issue that generated a <code>NullPointerException</code> if you called an <code>addFieldError</code> method from a validate method invoked from certain PCF page.
PL-11227	Corrected an issue that the application did not correctly apply the <code>useArchivedStyle</code> property to a <code>TextAreaCell</code> widget. (The cell still shows the standard font color.)

## 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 if they 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.

### On AIX without an X11 server, PolicyCenter may report that an X11 resource is missing (CC-5889)

- **Issue:** PolicyCenter does not require an X11 server; however other components of the PolicyCenter environment (such as Java) may inadvertently have this requirement. If so, PolicyCenter may report that an X11 resource is missing when it is starting up.
- **Workaround:** Install an X11 server, or contact Guidewire support to determine if the dependency can be removed.

### Port conflict when starting PolicyCenter with JMX RMI adapter enabled and Microsoft Outlook running (CC-6964)

- **Issue:** When Microsoft Outlook starts, it chooses several ports to use for its connection to an Exchange server. In some instances, Outlook may use port 1099, which is a port that the PolicyCenter JMX RMI adapter can use, *if enabled*. The PolicyCenter server will then report a port conflict during startup when it tries to use that port for the RMI adapter.
- **Workaround:** Close Outlook to free the port, and then start the PolicyCenter server again. You can also change the port that the RMI adapter uses by editing the `JMXManagementPlugin` through Guidewire Studio.

### Studio: “Verify Changed Resources” is not implemented in the product model (PC-6928)

- **Issue:** The Studio Tools → Verify → Verify Changed Resources command does not work with the PolicyCenter product model.

- **Workaround:** Use one of the other verification commands instead. See “Validating Studio Resources” in the *PolicyCenter Configuration Guide* for details.

#### Duplicate contact when merging contacts through API (PC-7421)

- **Issue:** When you merge two instances of a contact through the API, you may get duplicate contacts for the following contact roles:
  - PolicySecNamedInsured
  - PolicyPriNamedInsured
  - PolicyBillingContactThis was previously logged as CC-56177.
- **Status:** Guidewire is aware of this issue.

#### EvaluationException: [251861] in Account tab (PC-7861)

- **Issue:** Receive the following exception when creating a new submission from an newly created company account:  
EvaluationException: [251861] Cannot load a temporary id from the backing entity source: Account
- **Workaround:** In Studio, navigate to Page Configuration (PCF) → account → new. Click to open CreateAccount.pcf, and select the Code tab. Move the first line of the beforeCommitNewAccount() method (which calls setLastProducerSelection) to be the first line of the afterCommitNewAccount() method.

#### 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. This was previously logged as CC-54957.
- **Status:** Guidewire is aware of this issue.

#### Pre-update rules for extension entities (PL-8469)

- **Issue:** In configurations where the parameter UseOldStylePreUpdate is set to TRUE, pre-update rules cannot be applied to entities that cannot be validated.
- **Workaround:** Add `<implementsEntity name="Validatable"/>` to the definitions of all entities that the pre-update rules manipulate.

#### Need a summary of underwriting issues across all policies in a job group (PC-9102)

- **Issue:** The JobGroupRiskAnalysisLV needs to show a summary of underwriting issues across all policies in the job group. This was previously logged as CC-54177.
- **Status:** Guidewire is aware of this issue.

#### Answers persist on filtered out questions (PC-10645)

- **Issue:** All answers to questions in questionsets are normally persisted, even if the question itself is no longer relevant. 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 original answer is still saved along with the other answers.

- **Workaround:** The answers to these *irrelevant* questions 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)
        }
    }
}
```

#### Duplicate descriptions for image/JPG mime types in the configuration file (PC-12320)

- **Issue:** In the `config.xml` file, there are two entries under MIME types for jpeg: one for `pjpeg` and one for normal `jpeg`. This causes a problem as the descriptions of both are identical. If you upload a normal `jpeg` and chooses the wrong `jpeg` MIME type, an error occurs.
- **Workaround:** Change the description of the `pjpeg` so that duplicate entries do not appear in the MIME type drop down menu in Guidewire Studio.

#### 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 screen of the agent
- 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.

#### PolicyCenter does not display empty report folders (PL-1281)

- **Issue:** PolicyCenter does not display a report folder in the **Administration → Report Admin** page if that report folder is empty.
- **Workaround:** Within InetSoft Enterprise Manager, insert a *replet* (report template) into the empty report folder. You can then mark this *replet* as not visible (using the InetSoft Enterprise Manager), which causes the *replet* to not show in PolicyCenter. However, PolicyCenter does display the parent folder.

#### SampleAcrobat document template does not allow extra fields (PL-1416)

- **Issue:** Currently, the example Acrobat document template uses two fields, and this requires the same number of fields in `SampleAcrobat.pdf.descriptor` and `SampleAcrobat.pdf` files.
- **Workaround:** To allow more fields in the template and documents created from it, you must update its descriptor file.

#### Guidewire Studio PCF Editor treats commented-out property values as errors (PL-4582)

- **Issue:** If you attempt to comment out a property value in the Studio PCF editor, Studio treats this as an error.
- **Status:** None. Guidewire is aware of this issue.

**PolicyCenter Not Generating *soap.local.entity* Correctly (PL-7560)**

- **Issue:** PolicyCenter is not generating the *soap.local.entity* for a class used by web services.
- **Workaround:** To ensure that PolicyCenter generates the *soap.local.entity* correctly for *FieldValue* classes (which are Java types), subtype *FieldValue* into *GosuFieldValue*, then further subtype that class.

**Tab widgets do not work with Microsoft Windows Server 2003 enhanced security enabled (PL-9813)**

- **Issue:** If you have the Windows Server 2003 component called IE Enhanced Security Feature installed, tabs may be unresponsive in the user interface.
- **Workaround:** Uninstall this component from the Windows 2003 Server.

**Server exception during startup (PL-8167)**

- **Issue:** During server startup, there can be a Tomcat server exception. However, after the exception, the server does start up. This is caused by the Tomcat application server attempting to either save or restore sessions using serialization. The session restore causes an exception because PolicyCenter sessions contain objects which rely on the metadata being started up.
- **Workaround:** None. The Tomcat application server does start up after the error.

**Zone data with missing values cannot be loaded (PL-9123)**

- **Issue:** Zone data, which is used for associating postal codes with cities and regions, does not load if it is incomplete.
- **Workaround:** As a temporary workaround, you can enter a character into that field, for example, a hyphen (-).

**H2 development database creates LONGTEXT fields as VARCHAR (65000), instead of as CLOB as in Oracle and SQL Server (PL-9314)**

- **Issue:** Guidewire creates schema fields defined as LONGTEXT as LONGVARCHAR columns, instead as CLOB as in the Oracle or SQL Server databases. In addition to the inconsistency involved, this mandates a field length of 65,000 characters or less in LONGTEXT columns in the H2 database. Note that the H2 database is *only* used for testing and should never be used in production.
- **Workaround:** Define a LONGTEXT field as a VARCHAR( . . . ) with the required size, for example, as VARCHAR(120000).

**Country specific field validation does not run if the fields are not modified when a country is changed (PL-9828)**

- **Issue:** There may be an error if you change a contact field in the user interface that is validated with country specific field validators. Those specific validators do not get validated again if the country changes and the field is not modified. This issue only occurs if you use country specific field validators.

For example, suppose that a carrier operates in two countries, the United States and Canada and wants to implement different field validation for certain data types such as Tax ID. If an adjuster changes the country on an existing contact record from the United States to Canada, without modifying the Tax ID to reflect the new format, the system allows the US formatted Tax ID to be saved. If the adjuster *edits* the Tax ID, then the system validates the format again.

- **Status:** Guidewire is aware of this issue.

**InetSoft Exception Failed Login Exception: Bad user name anonymous (PL-10148)**

- **Issue:** InetSoft generates a Failed Login Exception: Bad user name anonymous exception on server start-up. This occurs if you set the InetSoft logging level to *fine*.
- **Workaround:** This error does not cause any issues. You can safely ignore the exception.

**Deleting a report folder from the Report Admin causes an exception (PL-10348)**

- **Issue:** If you delete a report folder from the Report Admin, users will not be able to access the Report Admin.
- **Status:** Guidewire is aware of this issue.

**Report Still Visible After Permissions Disabled For Parent Folder (PL-10353)**

- **Issue:** An InetSoft report that is included in a permission set is still visible in the report tree even after its parent has been removed from the permission set.
- **Workaround:** You must remove individual reports from permission sets instead of maintaining the reports at the report folder level.

**ToolbarButton Widget Does Not Render Correctly (PL-10742)**

- **Issue:** PolicyCenter incorrectly renders a ToolbarButton widget if the associated AddMenuItemIterator is empty.
- **Workaround:** Modify your configuration to disable the menu button if the button contains no menu item.

**Not possible to hot-deploy on all supported application servers (PL-9955)**

- **Issue:** It is not possible to hot-deploy on all supported application servers.
- **Workaround:** If you use Tomcat, you are able to see the changes without requiring a restart of the application server. If you use an other application server, you need to perform a restart to see any configuration changes reflected.

**Inability to access typecodes that contain a dash (PL-10532)**

- **Issue:** It is not possible to access a typekey code (typecode) that contains a dash using Gosu.
- **Workaround:** Remove any dashes from the typekey codes.

**Studio Debugger does not handle rule breakpoints properly (PL-10553)**

- **Issue:** If you set a breakpoint in a Gosu rule and attempt to step through the code using the Studio debugger, the debugger does not correctly display the code on the breakpoint step.
- **Status:** Guidewire is aware of this issue.

**Method *assignByUserAttributes* does not distinguish between attributes (PL-10607)**

- **Issue:** Method *assignByUserAttributes* does not distinguish between different attributes.
- **Workaround:** Ensure that you thoroughly test all assignment scenarios to ensure that assignment happens correctly.

**Must Name Row Iterator Element with Capital Letter (PL-10795)**

- **Issue:** The Document search results page (DocumentLV.pcf) throws a *NullPointerException* if the row iterator element name is *document* (with a lower case d).
- **Workaround:** You must use *Document* (with a capital D) instead.

**Finder in annotation causes throws error (PL-10812)**

- **Issue:** Studio throws error and fails to do syntax error highlight, code completion, and similar functions, if there is a finder in an annotation. This occurs even if you are connected to the server.
- **Workaround:** Remove the finder from the annotation.

**Studio does not properly report verification error (PL-10822)**

- **Issue:** Studio does not properly report a verification error if a typecode contains a trailing space.
- **Workaround:** Remove trailing spaces from all type codes.

**Navigating between pages in wizard loses data (PL-10920)**

- **Issue:** In wizard pop-ups, it is possible to lose data if you navigate to another page in the wizard.
- **Workaround:** Navigate out of the wizard and then back into it rather than changing steps directly in the wizard.

**Values in MIME-type drop-down menus are not localized (PL-10921)**

- **Issue:** PolicyCenter does not localize the values in the MIME type drop-down list properly.
- **Workaround:** If you want to localize MIME types, you can do so by modifying the definition in config.xml. Currently, PolicyCenter does not support MIME type definitions in multiple languages.

**Cancel button does not work properly (PL-10926)**

- **Issue:** The Cancel button does not work correctly if you start in edit mode by clicking on a button such as Add from read-only mode.
- **Workaround:** Click Edit to move the page into edit mode. In this case, all behavior is as expected.

**Double quotes in PATH environment variable causes problems (PL-10981)**

- **Issue:** PolicyCenter does not run if there are double quotes in the PATH environment variable.
- **Workaround:** Check the PATH variable and remove any double quotes.

**Large Number of Classes in plugins\shared\lib Causes Studio to Fail to Start (PL-11026)**

- **Issue:** If you attempt to load multiple JAR files that contain many thousands of classes, Studio fails to start and throws an OutOfMemory error.
- **Workaround:** Increase MaxPermSize to something over 128m (the default) so that Studio has enough memory to build the type information database.

**DFS exception (PL-11044)**

PolicyCenter generates a DFS exception if the server resources change while something is being processed. There are basically two ways to get this to happen.

- **Issue 1:** If you are running in a shared environment, and someone changes a configuration resource, other people using the server get the DFS exception.
- **Workaround 1:** To avoid this problem, set the ResourcesMutable flag to false in config.xml located in Other Resources in Studio.
- **Issue 2:** If you are developing on the same system where the server is running, the DFS exception occurs if you have an editable page open in PolicyCenter when you make a resource change. In PolicyCenter, essentially every wizard step is an editable page with an uncommitted bundle, so it is not unusual to be on a wizard step while you are changing resources. When you try to move to another wizard step (or click Save Draft), PolicyCenter attempts to commit the bundle, and you get the DFS exception. You can get the DFS exception even without reloading PCF files using ALT-SHIFT-L in PolicyCenter. You can get the DFS exception if you change some Gosu in Studio, and click Next in the wizard.
- **Workaround 2:** Navigate outside the wizard (to the Desktop tab, for example). Clear the unsaved work cache by clicking the trash icon in the Unsaved Work list. Then, go back into the wizard, and resume your work. Clearing out the unsaved work cache is important because if you don't, then it will just try to commit that invalid bundle again the next time you click Next. But you do not have to log out and you do not have to restart the server.

**ANT Command build-war Hangs Indefinitely (PL-11045)**

- **Issue:** It is possible for the build-war ANT command to hang indefinitely. This issue is only reproducible in certain environments.
- **Workaround:** Set the -noinput parameter before calling the build-war ANT command. For example, instead of gwcc.bat build-war, do the following:  
`gwcc.bat -noinput build-war`

**Cannot start a Spanish translation pack with the Oracle database (PL-11230)**

- **Issue:** If you set the application locale to a locale for which you did not configure database collations (in the collations.xml file), then the application fails on restart. If this is the case, then you see one of the following error messages (depending on your database):

- `java.sql.SQLException`
- `ORA-12705: Cannot access NLS data files or invalid environment specified`
- **Workaround:** Verify that you have properly configured database collations for a given locale before setting the application to that locale.

#### **Cache summary statistics rely on server ID being unique (PL-11239)**

- **Issue:** If you use the same server ID for multiple servers in a cluster, it can create the following problems:
  - Different servers in a cluster deleting or changing each others results
  - Deadlocks at the database level
- **Status:** Guidewire is aware of this issue.

#### **Missing Display Keys Cause Server to Fail to Start (PL-11764)**

- **Issue:** Missing display keys defined in the application default locale can cause the server to fail to start.
- **Workaround:** Within Guidewire Studio, run the **Verify Resources** utility before attempting to start the application server. PolicyCenter bases the type system for display keys on those defined in the default application locale. If a display key in the default application locale is missing and a rule references the missing display key, then PolicyCenter cannot compile the rule. Running the resource verification utility can catch these types of missing display keys.

#### **Linguistic Search Performance on Oracle is Slow (PL-12143)**

- **Issue:** PolicyCenter uses an Oracle Java Stored Function to perform linguistic searches. Such search queries can perform poorly. PolicyCenter invokes the stored function on Oracle if either one of the following is true:
  - The primary locale language is German or Japanese
  - The `LinguisticSearchCollation` strength is set to `primary` in the default locale's `GWLocale` element in file `localization.xml`.

This is an issue for any linguistic search in which you set the `LinguisticSearchCollation` strength to `primary`. For example, if you are working in a language that commonly uses accents (such as French or Spanish), and you wish to perform accent-insensitive searches, then you typically set the `LinguisticSearchCollation` strength to `primary` for that language. If you do so, it is possible to experience slow performance on an Oracle database as you perform linguistic searches.

- **Workaround:** Guidewire recommends that you set the `LinguisticSearchCollation` strength to `secondary` if applicable. (This is not applicable for Japanese and German.)

#### **The CSV file is incorrectly configured in mimetype mapping (PL-12841)**

- **Issue:** In PolicyCenter, the `.csv` file is in a different format than what is specified by the file extension.
- **Workaround:** Update the mimetype in your `config.xml` file.

# Guidewire PolicyCenter 4.0.4 Release Notes

## Release 4.0.4.7

Copyright © 2001-2010 Guidewire Software, Inc. All rights reserved. Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Insurance Suite, Guidewire ContactCenter, Guidewire Studio, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries. All other product names are trademarks of their respective companies.

**This documentation is published as Guidewire Confidential.** The contents of this documentation, including product architecture details and APIs, are considered confidential and are fully protected by customer licensing confidentiality agreements and signed Non-Disclosure Agreements (NDAs).

This product includes software developed by the Apache Software Foundation (<http://www.apache.org>).

## Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Changes in This Release
- Improvements and General Issues
- Known Issues and Limitations

## Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

## Version Number

- This release of Guidewire PolicyCenter is 4.0.4.7.
- If you are using the integration with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 3.0.1 or later.
- If you are using the integration with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 6.0.0 or later.
- For Standard Reporting, this release of PolicyCenter supports InetSoft StyleReport Enterprise Edition v10.1 20100113.

## Support

This document includes important information about the current release of Guidewire PolicyCenter. For assistance with this software release, contact Guidewire Customer Support. You can contact support on the Web at <http://portal.guidewire.com>, by email at [support@guidewire.com](mailto:support@guidewire.com), or by phone at +1-650-356-4955.

## 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

The following issues pertain to installing this release.

## InetSoft Evaluation

You can evaluate InetSoft Style Report Enterprise Edition Version 10.1 using the following license key:

L000-798-ERX-0000BF00100001F-F2AFC9DD34ED

If you want to use InetSoft and the PolicyCenter reporting module in a production environment, contact Guidewire Customer Support to request a full license key. For more information, see the *PolicyCenter Reporting Guide*.

---

**WARNING** The build number of the reporting install files must match the build number of the Guidewire PolicyCenter installation. If the build numbers do not match, you will have problems with your reporting installation.

---

## Changes in This Release

This section describes the product changes in this release.

- Base PCF File Changes
- Improvements to Upgrade Diff Report

## Base PCF File Changes

All links below require the `readme_files` directory on your local disk.

### PolicyCenter release 4.0.3 to 4.0.4

- To view a report of the changes in the base PCF files in the `modules/pc` directory, [click here](#).
- To view a report of the changes in the base PCF files in the `modules/p1` directory, [click here](#).

## Improvements to Upgrade Diff Report

With previous releases, Guidewire provided a static report in the release notes detailing certain differences between the current release and the prior minor release. This report described changes in display keys, entities, typelists, and the Gosu API. Guidewire has improved this report to dynamically conform to each customer's particular upgrade path. In addition, the report is regularly updated as the tools for generating it are enhanced to provide more information.

Because the new report is tailored to your particular upgrade requirements, it is no longer included with the general release notes. To obtain your custom Upgrade Diff Report, contact your Guidewire representative. In the future, this report will be available on the Guidewire Resource Center web portal, allowing you to view the latest version right when you need it.

## Improvements and General Issues

This topic describes the primary improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This is not meant to be a comprehensive list. Many performance related issues were addressed in this release but are not listed here.

ID	Description
<b>Core</b>	
PL-11189	Guidewire has determined that an intermittent verification error on autoCompleteArgIds occurs due to a JVM bug. You can safely ignore this error. Or, if you choose, you can upgrade your JVM to 1.6.0_21 or later, which removes the cause of the error message.
PL-11310	Corrected an issue in which, if you were defining the sortOrder for an element related to a query-backed ListView, Studio did not report an error if you chose an element for sorting that could not be used for sorting the backing query.
PL-11572	Corrected an issue that caused a Null Pointer Exception if you cancelled out of a page that contained unsaved work and the page contained data that had been deleted in the database.
PL-11691	Added a tool that generates a report listing the Gosu rules defined in the Rules editor in Guidewire Studio. To generate the report, type gwpc regen-rulereport in the application bin directory. See the <i>System Administration Guide</i> for more information.
PL-12314	Modified the data model and added a new remove-index element to the data model elements. You can use this element to remove an index if it exists purely for performance. Such indexes are either a) non-unique, or b) unique but contain an ID column. (Guidewire performs metadata validation to ensure that the element removes only those indexes that fall into one of these categories.) To modify a database index, you can remove it using this element, then add a new index with a different name and with the desired characteristics.
PL-12402	Modified the VenueType typelist to make it non-final.
PL-12500	Corrected an issue in which the Data Dictionary did not display certain virtual properties on an entity if the virtual properties were virtual property fields on a delegate implemented by the entity.
PL-13076	Added a tool that generates a report listing the Gosu rules defined in the Rules editor in Guidewire Studio. To generate the report, type gwpc rulereport in the application bin directory. See the <i>System Administration Guide</i> for more information.
PL-13430	Corrected an issue associated with localizing a column on a custom entity that created a gw.xml.XMLParseException: Could not parse XML file temp.dti exception. The exception occurred if you added a <localization> element to <column> and used an underscore in the column or entity name.
<b>Database</b>	
PL-10943	Modified the Server Tools → Info Pages → Load Integrity Checks so that it no longer downloads duplicate entries. The process now respects the status of the Allow Non Admin References: drop-down option.
PL-11812	Added a new action attribute to the <tablestatistics> element in config.xml. By default, PolicyCenter on Oracle does not generate statistics on any table used for processing work items. PolicyCenter deletes any existing statistics on these tables whenever PolicyCenter updates statistics. You can override this behavior using the action attribute of the <tablestatistics> element. See the <i>System Administration Guide</i> , "Configuring Database Statistics", for details.

<b>ID</b>	<b>Description</b>
PL-12143	Corrected an issue that caused poor performance on an Oracle database if you performed a linguistic search in which the Search strength was set to primary or the default application locale was Japanese or German.
PL-12806	Modified the Administration → Internal Tools → Database Table Info page so that it now includes database partition information.
PL-12854	Modified the PolicyCenter upgrader so that it now ignores the status of database constraints created by Oracle supplemental logging.
PL-12889	Corrected an issue in which the Work Queue Info page did not include the Throughout column in a download to a CSV file. This occurred primarily if there was a big disparity in the dates or the worker completed immediately.
PL-13048	Modified the Server Tools → Info Pages -> Data Distribution page to include information on the number of updated rows and the total number of updates in HTML tables.
PL-13461	Modified the Server Tools → Info Pages -> Data Distribution Information download logic so that it no longer generates data on all assignables by default. Formerly, a data distribution download generated large amounts of data on all assignables, even if you did not select the Assignable by date option. Now, you must select that option to view this information.
PL-13549	Corrected an issue that caused an ArrayIndexOutOfBoundsException if you performed a Server Tools → Info Page -> Data Distribution query on a very large database.
PL-9368	Added a Download Comparison Zip File button to the Server Tools → Info Pages -> Data Distribution Information page if you select multiple data distributions. This provides the ability to merge multiple data distribution reports. This change also adds an additional column on the distribution reports index page that shows the origin distribution number.
<b>Desktop</b>	
PC-12554	Added methods to support filtering job queries by date range. Use the Job.restrictors.createdDuring or Job.restrictors.closedDuring methods.
PC-12564	Fixed a performance issue with a Canceled Activities alert bar that was not dismissed.
PC-12585	Provided a Job.restrictors utility class to improve filtering jobs queries. The following code shows examples of using the Job.restrictors methods. <pre>uses gw.api.web.desktop.DesktopJobFilters var filter1 = new DesktopJobFilters.CompletedInLastNDays(3) // deprecated var filter2 = Job.restrictors.complete() // equivalent .with(Job.restrictors.closedAsOf(DateTime.CurrentDate.addDays(-3))) .asFilter() var filter3 = Job.restrictors.complete() // can be more expressive .with(Job.restrictors.closedAsOf(DateTime.CurrentDate.addBusinessDays(-7))) .with(Job.restrictors.unbound()) .asFilter()</pre>
PC-12594	Improved performance when finding jobs created as of a date.
PC-12596	You can now specify the types of jobs that Desktop → My Other Work Orders filters. To specify the types of jobs, go to Typelists → Job in Studio. On the Filters tab, the OtherWorkOrders filter allows you to specify the job types.
PC-12620	Fixed an issue where a user with nothing but an open activity to a recently complete job did not see that job in their desktop.
PC-12724	On the Desktop tab, the My Submissions, My Renewals, and My Other Work Orders screens display jobs or work orders related to the current user. PolicyCenter uses the SearchActivityThresholdDays parameter to determine which jobs or work orders to display on these screens.  A user is related to a job or work order if one of the following is true: <ul style="list-style-type: none"> <li>• If the user has a UserRoleAssignment for the job.</li> <li>• If an activity on the job is assigned to the current user, and the activity has been modified within SearchActivityThresholdDays before the current date. The UpdateTime field contains a timestamp of when the activity was last modified.</li> </ul>
PC-12762	Improved performance by adding an index to the Account entity.
PC-12765	In Studio, added an Account example in Data Model Extension → extensions. This ETX file provides columns supporting AccountSummaries. Customize this file if you change the AccountSummary view.  There is now a new Account.restrictors utility for improved filtering of account queries.
<b>Financials</b>	

ID	Description
PC-12185	Improved performance of the quote process.
<b>Integration</b>	
PL-12964	<p>Added a second version of the Document Assistant ActiveX control that Guidewire provides to manage many of the functions related to document manipulation. The default version uses a whitelist of file types that the Document Assistant control will open and display. The blacklist version provides a list of known potentially dangerous file types that it prevents from opening under any circumstances.</p> <p>For information on how to select the appropriate version of the Document Assistant, and for information on how to install and configure the Guidewire Document Assistant ActiveX control, see the <i>System Administration Guide</i>, "Configuring Guidewire Document Assistant".</p> <p><b>WARNING</b> The Guidewire Document Assistant, like all ActiveX controls, has known security vulnerabilities.</p>
<b>Job infrastructure</b>	
PC-12268	Fixed an issue where assigning a User to a new UserRole on a closed Job did not set the CloseDate on the UserRoleAssignment.
PC-12315	The database consistency check now ensures that UserRoleAssignment relationships with Account, Policy, and Job are separate. Unless your deployment has explicitly changed this, this should have no effect on your existing installations.
PC-12593	There is a new Closed filter on the PolicyPeriodStatus typelist. The Closed filter, not the Open filter, is used in the Gosu code and other files provided in the default configuration. The Open filter remains in the PolicyPeriodStatus typelist but is not used in the default configuration.
PC-12692	Fixed performance issues with handling preemptions in commercial property.
<b>Jobs</b>	
PC-12327	Fixed an issue where quoting a submission with underwriting issues related to prior policies with data changes caused an exception.
PC-12828	Improved performance during conversion on renewal.
PC-12836	Fixed an issue with calculating the renewal process lead time. The date for comparison is now the expiration of the current term, not the start of the renewal term. This is especially important in the case of renewal with gap.
PC-12876	<p>To improve performance, there are changes to the conversion on renewal code in the ExternalPolicyPeriod and PolicyRenewalAPI Gosu classes.</p> <p>There are several changes in the gw.webservice.pc.policy.period.ExternalPolicyPeriod Gosu class.</p> <p>There is a new method called createRenewal. The createRenewal method provides code to populate the branch by calling the populateBranch method.</p> <p>The createBranch method has been replaced by a populateBranch method. The code is similar to the old createBranch method except for the following:</p> <ul style="list-style-type: none"> <li>• The check for required variables occurs in the createRenewal method.</li> <li>• Creating the account and branch are now explicit responsibilities of the account through the Account.createConversionRenewal method.</li> <li>• The populateBranch method is void.</li> </ul> <p>There is a change in the gw.webservices.pc.job.PolicyRenewalAPI Gosu class. Previously, the renewNewPersonalAutoPolicy method had code telling externalPeriod to createBranch, then create a Renewal and attached the branch to it using the startJob(branch) method. Now, the code tells externalPeriod to createRenewal() and uses that return value as the renewal.</p>
<b>LOB - Workers' Compensation</b>	
PC-12531	In the workers' compensation line of business, the entities WCRetrospectiveRatingPlan and WCStateMultiplier are now extendable.
PC-12533	The Gosu classes related to workers' compensation options are now available in Studio.
PC-12621	Fixed an issue where PolicyCenter threw an error when changing the anniversary rating date in a policy change job.
<b>Rating</b>	

<b>ID</b>	<b>Description</b>
PC-12536	<p>You can now override prorated calculations in financial transactions. For example, this allows you to override prorated calculations to use 365 days in all years, including leap years which actually have 366 days.</p> <p>In the default configuration, all financial date calculation go through a prorater class instance created in <code>ProrationPlugin</code>. (The default implementation is <code>gw.plugin.policyperiod.impl.PrationPlugin.gs</code>)</p> <p>To change how PolicyCenter prorates financial date calculations, you can either replace <code>ProrationPlugin</code> altogether, or modify the sample one with new implementations of the methods:</p> <ul style="list-style-type: none"> <li>• <code>function financialDaysBetween(startDate : Date, endDate : Date) : int</code></li> <li>• <code>function findEndOfRatedTerm(startDate : Date, daysInTerm : int) : Date</code></li> </ul>
<b>Revisioning</b>	
PC-12886	Fixed an issue in business auto where a policy change after a lapse in coverage caused a removed coverage to apply to the full term.
<b>Searching</b>	
PC-12242	Improved search in notes. Added search by author for accounts and policies. Added a starts with search on subject.
PC-12907	Improved performance of searching for policies.
<b>Studio</b>	
PL-11195	Corrected an issue that caused Studio to truncate the display key text used to generate widget titles in PCF files.
PC-12346	The code for the <code>AccountContactRolePlugin</code> is now marked as <code>@Export</code> and is accessible in Studio.
<b>Studio - Product Model</b>	
PC-12208	Improved performance of checking coverages against the product model.
PC-12745	Made changes to modifiers to prevent the creation of more than one modifiers with the same pattern.
PC-12747	Fixed a potential security issue by limiting access to certain internal java classes. There is no impact to customer configurations.
PL-13220	Improved Studio activation performance if using SCM systems that maintain a local repository (for example, SVN and CVS) and Studio is configured with proper repository mask in Tools   Options.
PL-13509	Modified Studio to provide ability for time-sensitive diagnostic logging in the Studio console. You can enable this functionality by creating a <code>studio.properties</code> file in Other Resources → logging and populating it with the following text:
	<pre># enable studio debug level logging log4j.category.Studio=DEBUG</pre>
<b>Team support</b>	
PC-12627	Fixed an issue where an activity assigned to a user who has no roles or activities on the job did not appear in the Team Screen detail view for submission and renewal jobs.
<b>UI Configuration</b>	
PC-12383	<p>On the Team tab, the <code>TeamScreenTabVisibilityRowCountCutoff</code> parameter sets the maximum number of rows that the Activities, Submissions, Renewals, and Other Work Orders screens display. Increasing this parameter may increase the amount of time that PolicyCenter takes to render the Team screens.</p> <p>The total number of results return by any one of the filters for Activities, Submissions, Renewals, or Other Work Orders must be less than this parameter. If the number of result is greater, then PolicyCenter displays a message on that screen and does not display the results. Using submissions as an example, view the Summary screen. If the total number of Open, New, or Bound submissions is greater than the cutoff parameter, then PolicyCenter displays a message on the Submissions screen. PolicyCenter does not display any submissions on the Submissions screen.</p> <p>If you select an individual user, PolicyCenter always displays the filtered results even if the number of results exceed the cutoff parameter.</p>
<b>Underwriting authority</b>	
PC-12329	The virtual array <code>PolicyPeriod.UWIssues</code> returns only those <code>UWIssues</code> which are currently active. There is a new array, <code>PolicyPeriod.UWIssuesIncludingSoftDeleted</code> , which returns the complete set of <code>UWIssues</code> .
PC-12644	The UW Issue summary and description are now localizable.
<b>Upgrade</b>	

ID	Description
PC-12297	Fixed an issue where PolicyCenter threw an error during quote on a policy change. This occurred after an upgrade from PolicyCenter 3.0.3 to 4.0.3.
<b>Web</b>	
PL-12262	Corrected an issue with Date input containing Japanese characters that sometimes caused the date text to not be fully visible in the text box. This caused the user to scroll the text to see the last part of the date.
PL-12842	Improved slow performance in Internet Explorer 7 that occurred if you moused over a page with a very long ListView and a Worksheet.
PL-13043	Modified the behavior of the Eras drop-down in the Japanese Imperial Calendar widget so that it displays only the most four recent eras (Meiji, Taicho, Showa, and Hisei).

## 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 if they 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.

### [Studio: “Verify Changed Resources” is not implemented in the product model \(PC-6928\)](#)

**Issue:** The Studio Tools → Verify → Verify Changed Resources command does not work with the PolicyCenter product model.

**Workaround:** Use one of the other verification commands instead. See “Validating Studio Resources” in the *PolicyCenter Configuration Guide* for details.

### [Duplicate contact when merging contacts through API \(PC-7421\)](#)

**Issue:** When you merge two instances of a contact through the API, you may get duplicate contacts for the following contact roles:

- PolicySecNamedInsured
- PolicyPriNamedInsured
- PolicyBillingContact

This was previously logged as CC-56177.

**Status:** Guidewire is aware of this issue.

### [EvaluationException: \[251861\] in Account tab \(PC-7861\)](#)

**Issue:** Receive the following exception when creating a new submission from a newly created company account:

EvaluationException: [251861] Cannot load a temporary id from the backing entity source: Account

**Workaround:** In Studio, navigate to Page Configuration (PCF) → account → new. Click to open CreateAccount.pcf, and select the Code tab. Move the first line of the beforeCommitNewAccount() method (which calls setLastProducerSelection) to be the first line of the afterCommitNewAccount() method.

### [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. This was previously logged as CC-54957.

**Status:** Guidewire is aware of this issue.

## Need a summary of underwriting issues across all policies in a job group (PC-9102)

**Issue:** The JobGroupRiskAnalysisLV needs to show a summary of underwriting issues across all policies in the job group. This was previously logged as CC-54177.

**Status:** Guidewire is aware of this issue.

## 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. 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 original answer is still saved along with the other answers.

**Workaround:** The answers to these *irrelevant* questions 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)
        }
    }
}
```

## Duplicate descriptions for image/JPG mime types in the configuration file (PC-12320)

**Issue:** In the `config.xml` file, there are two entries under MIME types for jpeg: one for `pjpeg` and one for normal `jpeg`. This causes a problem as the descriptions of both are identical. If you upload a normal jpeg and chooses the wrong jpeg MIME type, an error occurs.

**Workaround:** Change the description of the `pjpeg` so that duplicate entries do not appear in the MIME type drop down menu in Guidewire Studio.

## 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 screen of the agent
- 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.

### PolicyCenter does not display empty report folders (PL-1281)

**Issue:** PolicyCenter does not display a report folder in the Administration → Report Admin page if that report folder is empty.

**Workaround:** Within InetSoft Enterprise Manager, insert a *replet* (report template) into the empty report folder. You can then mark this *replet* as not visible (using the InetSoft Enterprise Manager), which causes the *replet* to not show in PolicyCenter. However, PolicyCenter does display the parent folder.

### SampleAcrobat document template does not allow extra fields (PL-1416)

**Issue:** Currently, the example Acrobat document template uses two fields, and this requires the same number of fields in SampleAcrobat.pdf.descriptor and SampleAcrobat.pdf files.

**Workaround:** To allow more fields in the template and documents created from it, you must update its descriptor file.

### Guidewire Studio PCF Editor treats commented-out property values as errors (PL-4582)

**Issue:** If you attempt to comment out a property value in the Studio PCF editor, Studio treats this as an error.

**Status:** Guidewire is aware of this issue.

### PolicyCenter Not Generating soap.local.entity Correctly (PL-7560)

**Issue:** PolicyCenter is not generating the soap.local.entity for a class used by web services.

**Workaround:** To ensure that PolicyCenter generates the soap.local.entity correctly for FieldValue classes (which are Java types), subtype FieldValue into GosuFieldValue, then further subtype that class.

### Server exception during startup (PL-8167)

**Issue:** During server startup, there can be a Tomcat server exception. However, after the exception, the server does start up. This is caused by the Tomcat application server attempting to either save or restore sessions using serialization. The session restore causes an exception because PolicyCenter sessions contain objects which rely on the metadata being started up.

**Workaround:** None. The Tomcat application server does start up after the error.

### Pre-update rules for extension entities (PL-8469)

**Issue:** In configurations where the parameter UseOldStylePreUpdate is set to TRUE, pre-update rules cannot be applied to entities that cannot be validated.

**Workaround:** Add <implementsEntity name="Validatable"/> to the definitions of all entities that the pre-update rules manipulate.

### Zone data with missing values cannot be loaded (PL-9123)

**Issue:** Zone data, which is used for associating postal codes with cities and regions, does not load if it is incomplete.

**Workaround:** As a temporary workaround, you can enter a character into that field, for example, a hyphen (-).

### H2 development database creates LONGTEXT fields as VARCHAR (65000), instead of as CLOB as in Oracle and SQL Server (PL-9314)

**Issue:** Guidewire creates schema fields defined as LONGTEXT as LONGVARCHAR columns, instead as CLOB as in the Oracle or SQL Server databases. In addition to the inconsistency involved, this mandates a field length of 65,000

characters or less in LONGTEXT columns in the H2 database. Note that the H2 database is *only* used for testing and should never be used in production.

**Workaround:** Define a LONGTEXT field as a VARCHAR(...) with the required size, for example, as VARCHAR(120000).

#### [Tab widgets do not work with Microsoft Windows Server 2003 enhanced security enabled \(PL-9813\)](#)

**Issue:** If you have the Windows Server 2003 component called IE Enhanced Security Feature installed, tabs may be unresponsive in the user interface.

**Workaround:** Uninstall this component from the Windows 2003 Server.

#### [Country specific field validation does not run if the fields are not modified when a country is changed \(PL-9828\)](#)

**Issue:** There may be an error if you change a contact field in the user interface that is validated with country specific field validators. Those specific validators do not get validated again if the country changes and the field is not modified. This issue only occurs if you use country specific field validators.

For example, suppose that a carrier operates in two countries, the United States and Canada and wants to implement different field validation for certain data types such as Tax ID. If an adjuster changes the country on an existing contact record from the United States to Canada, without modifying the Tax ID to reflect the new format, the system allows the US formatted Tax ID to be saved. If the adjuster *edits* the Tax ID, then the system validates the format again.

**Status:** Guidewire is aware of this issue.

#### [Not possible to hot-deploy on all supported application servers \(PL-9955\)](#)

**Issue:** It is not possible to hot-deploy on all supported application servers.

**Workaround:** If you use Tomcat, you are able to see the changes without requiring a restart of the application server. If you use an other application server, you need to perform a restart to see any configuration changes reflected.

#### [First time you click on the arrow of the typekey input, the drop-down menu will not open \(PL-10134\)](#)

**Issue:** The drop-down menu does not open on the first click of the arrow on a typekey input. Instead, the help text opens.

**Workaround:** Turn off help text on focus by setting InputHelpTextOnFocus to false in the config.xml file. By doing that, the help text shows only if you mouse over the input and will not interfere with opening a drop down menu.

#### [InetSoft Exception Failed Login Exception: Bad user name anonymous \(PL-10148\)](#)

**Issue:** InetSoft generates a Failed Login Exception: Bad user name anonymous exception on server start-up. This occurs if you set the InetSoft logging level to *fine*.

**Workaround:** This error does not cause any issues. You can safely ignore the exception.

#### [Report Still Visible After Permissions Disabled For Parent Folder \(PL-10353\)](#)

**Issue:** An InetSoft report that is included in a permission set is still visible in the report tree even after its parent has been removed from the permission set.

**Workaround:** You must remove individual reports from permission sets instead of maintaining the reports at the report folder level.

### Studio Debugger does not handle rule breakpoints properly (PL-10553)

**Issue:** If you set a breakpoint in a Gosu rule and attempt to step through the code using the Studio debugger, the debugger does not correctly display the code on the breakpoint step.

**Status:** Guidewire is aware of this issue.

### Method `assignByUserAttributes` does not distinguish between attributes (PL-10607)

**Issue:** Method `assignByUserAttributes` does not distinguish between different attributes.

**Workaround:** Ensure that you thoroughly test all assignment scenarios to ensure that assignment happens correctly.

### Studio Rules do not use correct capitalization for root object's name (PL-10740)

**Issue:** Rule set root objects are not named with first letter lower-cased.

**Workaround:** Rules engine will issue warnings when correct case for objects is not being used.

### ToolbarButton Widget Does Not Render Correctly (PL-10742)

**Issue:** PolicyCenter incorrectly renders a `ToolbarButton` widget if the associated `AddMenuItemIterator` is empty.

**Workaround:** Modify your configuration to disable the menu button if the button contains no menu item.

### Must Name Row Iterator Element with Capital Letter (PL-10795)

**Issue:** The Document search results page (`DocumentLV.pcf`) throws a `NullPointerException` if the row iterator element name is `document` (with a lower case d).

**Workaround:** You must use `Document` (with a capital D) instead.

### Finder in annotation causes throws error (PL-10812)

**Issue:** Studio throws error and fails to do syntax error highlight, code completion, and similar functions, if there is a finder in an annotation. This occurs even if you are connected to the server.

**Workaround:** Remove the finder from the annotation.

### Studio does not properly report verification error (PL-10822)

**Issue:** Studio does not properly report a verification error if a typecode contains a trailing space.

**Workaround:** Remove trailing spaces from all type codes.

### Navigating between pages in wizard loses data (PL-10920)

**Issue:** In wizard pop-ups, it is possible to lose data if you navigate to another page in the wizard.

**Workaround:** Navigate out of the wizard and then back into it rather than changing steps directly in the wizard.

### Values in MIME-type drop-down menus are not localized (PL-10921)

**Issue:** PolicyCenter does not localize the values in the `MIME type` drop-down list properly.

**Workaround:** If you want to localize MIME types, you can do so by modifying the definition in `config.xml`. Currently, PolicyCenter does not support MIME type definitions in multiple languages.

## Double quotes in PATH environment variable causes problems (PL-10981)

**Issue:** PolicyCenter does not run if there are double quotes in the PATH environment variable.

**Workaround:** Check the PATH variable and remove any double quotes.

## Large Number of Classes in plugins\shared\lib Causes Studio to Fail to Start (PL-11026)

**Issue:** If you attempt to load multiple JAR files that contain many thousands of classes, Studio fails to start and throws an OutOfMemory error.

**Workaround:** Increase MaxPermSize to something over 128m (the default) so that Studio has enough memory to build the type information database.

## ANT Command build-war Hangs Indefinitely (PL-11045)

**Issue:** It is possible for the build-war ANT command to hang indefinitely. This issue is only reproducible in certain environments.

**Workaround:** Set the -noinput parameter before calling the build-war ANT command. For example, instead of gwcc.bat build-war, do the following:

```
gwcc.bat -noinput build-war
```

## Intermittent verification error on autoCompleteArgIds (PL-11189)

**Issue:** An intermittent verification error on autoCompleteArgIds occurs due to a JVM bug.

**Workaround:** You can safely ignore this error. Or, if you choose, you can upgrade your JVM to 1.6.0\_21 or later, which removes the cause of the error message.

## Cache summary statistics rely on server ID being unique (PL-11239)

**Issue:** If you use the same server ID for multiple servers in a cluster, it can create the following problems:

- Different servers in a cluster deleting or changing each others results
- Deadlocks at the database level

**Status:** Guidewire is aware of this issue. Contact Guidewire customer support for more information.

## Missing Display Keys Cause Server to Fail to Start (PL-11764)

**Issue:** Missing display keys defined in the application default locale can cause the server to fail to start.

**Workaround:** Within Guidewire Studio, run the Verify Resources utility before attempting to start the application server. PolicyCenter bases the type system for display keys on those defined in the default application locale. If a display key in the default application locale is missing and a rule references the missing display key, then PolicyCenter cannot compile the rule. Running the resource verification utility can catch these types of missing display keys.

## Countries configured in zone-config.xml still generate a warning during regen-dictionary even when zone data is loaded for all of these countries (PL-11947)

**Issue:** Countries configured in zone-config.xml still generate a warning during regen-dictionary even when zone data is loaded for all of these countries.

**Workaround:** Warning message is created in error and can safely be ignored.

## The user interface cannot handle starting multiple instances of a batch process (PL-12372)

**Issue:** The user interface cannot handle multiple instances of a batch process.

**Workaround:** If multiple instances of a batch process need to be executed, they must be started from the command line. Also, you should ensure that the `BatchProcess.isExclusive()` method returns `false` to allow multiple instances to be run simultaneously.

#### [The CSV file is incorrectly configured in mimetype mapping \(PL-12841\)](#)

**Issue:** In PolicyCenter, the .csv file is in a different format than what is specified by the file extension.

**Workaround:** Update the mimetype in your `config.xml` file.

#### [Type system refresh after PCF page title change does not update corresponding menu label \(PL-13057\)](#)

**Issue:** The type system refresh after the PCF page title change does not update corresponding menu label.

**Workaround:** After updating a page title, the server restart must be done to refresh menu labels and avoid null pointer exceptions due to stale reference.

#### [There is a length limitation on entity localization table names \(PL-13360\)](#)

**Issue:** There is a length limitation on entity localization table names.

**Workaround:** Ensure that `localization.tableName` property specified in the entity extension file is less than 16 characters. The error message generated if the localization table name exceeds the maximum length indicates that 18 characters are allowed, but that does not account for two additional characters added by the application.

#### [US-Locations.txt file with the US geodata from GreatData has special characters that cause validation problems with United States Postal Service \(USPS\) data \(PL-13384\)](#)

**Issue:** The `US-Locations.txt` file contains information that does not conform to United States Postal Service (USPS) standards for bulk mailings.

**Workaround:** The provided `US-Locations.txt` file is intended only for use in geocoding to identify addresses for a location. You can process the `US-Locations.txt` file to conform to your particular address standards, and then import that version of the file instead.

#### [Guidewire Document Assistant ActiveX control, blacklist version, requires changes to DocumentControl.gs \(PL-13501\)](#)

**Issue:** The `CLASSID` and `CODEBASE` values for the blacklist version of the Guidewire Document Assistant are not included in the `DocumentControl.gs` file.

**Workaround:** If you want to use the blacklist version of Document Assistant, replace the `CLASSID` and `CODEBASE` values in `DocumentControl.gs` with the following values:

`CLASSID="CLSID:01A307B7-5CB5-4D91-A830-68BC53F12FD6"`

`CODEBASE="<% activexCodebase %>/GuidewireDocumentAssistantConfigurable.CAB#version=2,1,0,1"`

You can access `DocumentControl.gs` in Studio at: `configuration` → `Web Resources` → `web/templates` → `document` → `DocumentControl.gs`

For more information, see the Administration Guide topic: *Configuring Guidewire Document Assistant*.

#### [GX models that reference virtual fields and enhancements throw null pointers if null \(PL-13560\)](#)

**Issue:** The GX models that reference virtual fields and enhancements throw null pointers when they are null.

**Workaround:** Ensure that null checks and error handling is included so that if referenced virtual fields or enhancements are null, then there will not be a null pointer exception.

## Sending email with file attachment with unicode filename is not correctly handed over to the mail server (PL-13582)

**Issue:** Sending email with file attachment with unicode filename is not sent to the mail server correctly.

**Workaround:** Use Latin characters for file names on attached files.

## The GX model generated XSD cannot be parsed by JAXB (PL-13598)

**Issue:** The GX model generated XSD cannot be parsed by JAXB.

**Workaround:** Add JAXB annotation elements to the XSD to specify the necessary metadata (such as class names) for JAXB to generate the java class files. Contact Guidewire Support for an example XSD annotated in this way.

## JavaToolkit.gs has incorrectly hard coded memory which results in failed regen-java-api ant task (PL-13663)

**Issue:** The JavaToolkit.gs has hard coded memory which can result in failed regen-java-api ant tasks.

**Workaround:** Increase the size of the maximum heap setting on line 161 of JavaToolkit.gs in the Ant module. The default value is 512.

## Cannot make a field from a delegate a localized column (PL-13761)

**Issue:** You cannot make a field from a delegate a localized column.

**Workaround:** Move the column to be localized off the delegate and onto each of the implementing entities. Then, in order to make the column appear as though it exists on the delegate, define an enhancement property on the delegate, that *delegates* to the appropriate column, depending on the implementing entity.

## Copy and paste does not work from phone fields (PL-13792)

**Issue:** The copy and paste does not work for phone fields on a contact.

**Workaround:** Save the contact with the newly entered phone information before attempting to either copy or paste the value into another field.

## DiffUtils.findDuplicates method returns large number of queries (PL-14130)

**Issue:** It is possible for policies with a large number of data entities (such common commercial policies) to experience significant performance issues. Policies with root entities that have a large number of child entities (a location that contains a large number of buildings, for example) are especially vulnerable to this type of performance issue.

**Status:** Guidewire is aware of this issue.

# Guidewire PolicyCenter 4.0.5 Release Notes

## Release 4.0.5.13

Copyright © 2001-2011 Guidewire Software, Inc. All rights reserved. Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Insurance Suite, Guidewire ContactCenter, Guidewire Studio, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries. All other product names are trademarks of their respective companies.

**This documentation is published as Guidewire Confidential.** The contents of this documentation, including product architecture details and APIs, are considered confidential and are fully protected by customer licensing confidentiality agreements and signed Non-Disclosure Agreements (NDAs).

This product includes software developed by the Apache Software Foundation (<http://www.apache.org>).

## Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Changes in This Release
- Improvements and General Issues
- Known Issues and Limitations

## Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

## Version Number

- This release of Guidewire PolicyCenter is 4.0.5.13.
- If you are using the integration with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 3.0.1 or later.
- If you are using the integration with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 6.0.0 or later.
- For Standard Reporting, this release of PolicyCenter supports InetSoft StyleReport Enterprise Edition v10.1 20100113.

## Support

This document includes important information about the current release of Guidewire PolicyCenter. For assistance with this software release, contact Guidewire Customer Support:

- At the Guidewire Resource Center – <http://guidewire.custhelp.com>
- By email – [support@guidewire.com](mailto:support@guidewire.com)
- By phone – +1-650-356-4955

## 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

The following issues pertain to installing this release.

## InetSoft Evaluation

You can evaluate InetSoft Style Report Enterprise Edition Version 10.1 using the following license key:

L000-72C-ERX-0000BF00100001F-F2B025280327

If you want to use InetSoft and the PolicyCenter reporting module in a production environment, contact Guidewire Customer Support to request a full license key. For more information, see the *PolicyCenter Reporting Guide*.

---

**WARNING** The build number of the reporting install files must match the build number of the Guidewire PolicyCenter installation. If the build numbers do not match, you will have problems with your reporting installation.

---

## Changes in This Release

This section describes the product changes in this release.

- Base PCF File Changes
- Custom Upgrade Diff Report
- Change in Web Service Behavior for Committing Data to the Database (PL-14984)

## Base PCF File Changes

All links below require the `ReleaseNotes_files` directory on your local disk.

**PolicyCenter release 4.0.4 to 4.0.5**

- To view a report of the changes in the base PCF files in the `modules/pc` directory, [click here](#).
- To view a report of the changes in the base PCF files in the `modules/p1` directory, [click here](#).

## Custom Upgrade Diff Report

Guidewire provides a custom report detailing certain differences between the current release and the prior minor release. This report describes changes in display keys, entities, typelists, and the Gosu API. This custom report dynamically conforms to each customer's particular upgrade path. In addition, the report is regularly updated as the tools for generating it are enhanced to provide more information.

This report is available on the Guidewire Resource Center web portal.

## Change in Web Service Behavior for Committing Data to the Database (PL-14984)

The original integration design for Guidewire products focused on ensuring fail-safe integration with external systems and preventing real-time failures due to concurrent data changes. To reduce the chance of unexpected commit errors, during API calls to published web services, the application omitted verification of whether the entity is the most recent version in the database. In other words, the server skipped the check to determine if the entity changed between (a) when the application loaded it from the application cache and (b) when the bundle commits.

This means that you may be at risk to corrupting data in your database if any of the following situations exists in your web services published from a Guidewire product:

- External systems calling web services update data in a parallelized manner. If you call the same API concurrently with multiple threads to boost overall throughput, or from multiple integration points, there is risk of corrupting data.
- Web service implementations updating entities that users can modify from the application user interface, or similarly, simultaneous changes from other web service API calls, workflows, batches, or other background processes. Because it is possible for two different threads to modify the same entity, ignoring concurrent data change exception (CDCE) errors can possibly corrupt data. Recent changes can be overwritten.

### Impact

Recently, some Guidewire customers have reported issues involving the previously described scenarios. This can manifest as one or more of the following:

- The sequence generator utility returning duplicate (invalid) sequence numbers. When the application commits the bundle, the SQL row insert fails due to non-unique IDs.
- PolicyCenter financials consistency checks can fail.
- PolicyCenter can experience other data loss issues.

**IMPORTANT** Published web service APIs that perform only read-only (get) operations are completely unaffected by this issue.

### Solution

All Guidewire customers must do the following:

1. Analyze web services usage to see if the above scenarios affect your implementation.
2. Run the database consistency checker tool, also known as `dbcc checker`. This checks for certain types of potentially inconsistent data modified by web services APIs.

**IMPORTANT** This tool does **not** detect all possible types of corrupted data.

### New Web Services Configuration Parameter

Guidewire has added a new configuration parameter, `UseSafeBundleForWebServiceOperations`, that changes the behavior of bundle commits in web services published on this server. The default value is `true`.

- If set to `false`, the application ignores bean version conflicts as it commits a bundle.
- If set to `true`, the application detects (and does not ignore) bean version conflicts.

If you leave this parameter set to `true`, it is possible for a bundle commit to throw a `ConcurrentDataChangeException`. This can happen if another thread or cluster node modified this entity after PolicyCenter loaded it from database into the current bundle.

If this error condition occurs, then PolicyCenter catches and rethrows the `ConcurrentDataChangeException` as `gw.api.webservice.exception.SOAPRetryableException`. This is a new exception. Guidewire strongly recommends that you update all web service clients to catch this exception and retry the SOAP API call.

Leaving `UseSafeBundleForWebServiceOperations` with its default value of `true` sets the application behavior system-wide. It is possible to reverse this behavior (to make it unsafe) on a method-by-method basis by adding the `@WebServiceIgnoreBeanVersionConflicts` annotation to a method. This annotation only works immediately before a method declaration, not a class declaration, and only on classes with the `@WebService` annotation. There is a known issue that Gosu appears to permit `@WebServiceIgnoreBeanVersionConflicts` on the class level without a compile error. However, it has no effect at the class level.

---

**IMPORTANT** It is possible that you previously implemented a workaround for this problem by using the `setIgnoreVersionConflicts` method on the `CommitOptions` at the beginning of SOAP implementation methods. If so, then you must update your client-side logic for detecting and retrying in the case of a concurrent change exception to handle the new exception.

---

## Improvements and General Issues

This topic describes the primary improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This is not meant to be a comprehensive list. Many performance related issues were addressed in this release but are not listed here.

ID	Description
<b>Desktop</b>	
PC-13171	Improved performance of Administration → Organization Search.
<b>Job Infrastructure</b>	
PC-12914	To improve performance, the <code>XXPolicyLinesMethods.gs</code> for each line now overrides the <code>CostsVLs</code> property instead of the <code>Costs</code> property.  To improve performance, be sure to make this change in the policy lines you implement. For an example, see <code>gw.lob.pa.PAPolicyLineMethods.gs</code> . Search for:  <code>override property get CostVLs</code>
PC-12971	Fixed an issue where an exception that occurred during the copy process ( <code>SubmissionEnhancement.copyPolicyPeriod</code> ) left a duplicate Policy instance (a policy with the same policy number as another) in the database.
PC-13092	Fixed an issue with the Job Expire batch process. Setting a negative value for <code>JobExpirationCreateDateThreshold</code> now disables creation date checking. Previously, setting a negative value for the <code>JobExpirationCreateDateThreshold</code> parameter in <code>config.xml</code> always expired the job.
<b>Jobs</b>	
PC-12828	As of PolicyCenter 4.0.4, the <code>Account.createPolicyPeriod</code> method has been removed. Using this method, you were able to attach a draft policy period to a renewal. Instead, the appropriate way to convert a policy on renewal is to invoke <code>Account.createConversionRenewal</code> . The <code>API</code> and <code>ExternalPolicyPeriod</code> entity have been changed to reflect this new mechanism.

ID	Description
PC-13409	<p>Fixed an issue where the process start date of a final audit was the effective date of a policy change. Now both the start and due date are based on the cancellation effective date.</p> <p>For example:</p> <ol style="list-style-type: none"> <li>1. Cancel a policy with a cancellation effective date in the past.</li> <li>2. Submit the final audit.</li> <li>3. Bind a policy change with an effective date earlier than the cancellation effective date.</li> </ol> <p>Result:</p> <p>A new final audit is automatically scheduled when the out-of-sequence policy change is bound.</p> <p>Previous result:</p> <p>The process start date (<code>initDate</code>) and the due date are based on the effective date of the policy change.</p> <p>New result:</p> <p>The process start date and due date are based on the cancellation effective date (audit period end date).</p>
PC-13534	<p>In previous versions of PolicyCenter, the <code>RenewalProcessLeadTime</code> value in <code>config.xml</code> was incorrectly calculated by PolicyCenter. The <code>RenewalProcessLeadTime</code> parameter specifies the time (in number of days) before the policy expires and renewal processing should start. If you specified 165 in a previous version, then the time lead time was actually 164 days instead of 165 days.</p> <p>This issue was corrected in this version. Therefore, if you specify 165, the lead time will be 165 days.</p>
<b>Product Model</b>	
PC-13022	Fixed an issue with deleting coverages from a product model in Studio. In some cases, deleting a coverage took a long time, and Studio output stack traces to the errors console.
<b>Studio</b>	
PC-11474	In the <i>Configuration Guide</i> , there is now a section titled “Adding Icons to Studio for the Product and Policy Line”. This section provides step-by-step instructions on how to add icons to Studio for new products and policy lines.
<b>User Tools</b>	
PC-13176	Fixed an issue in the <b>Search &gt; Activities</b> screen. Previously if you did not enter a value in the <b>Assigned to</b> field, PolicyCenter returned all activities. The search results displayed activities assigned to other users in other organizations or on policies not allowed through producer code security.
PC-13383	<p>Fixed an issue with term number and withdrawn policy periods. Previously, when a renewal started on a policy with a withdrawn renewal, PolicyCenter incremented the term number again even though the renewal job was withdrawn.</p> <p>For example:</p> <ol style="list-style-type: none"> <li>1. Start new renewal on a policy. The renewal term number is set to 2.</li> <li>2. Withdraw the renewal with term number 2.</li> <li>3. Start a new renewal on the policy.</li> </ol> <p>Previously, PolicyCenter set the term number to 3. Now PolicyCenter sets the term number to 2.</p>
<b>Core</b>	
PL-11508	Corrected a regression that hid a number of methods previously available on Java <code>external.entity</code> interfaces. Guidewire has re-exposed these methods.
PL-11585	Corrected an issue in which the application did not recognize a modified version of <code>web.xml</code> , but instead utilized the original version.
PL-11931	Updated the description of properties <code>RateScale</code> and <code>NormalizedRate</code> on the <code>ExchangeRate</code> entity to consider them deprecated. Use the <code>Rate</code> property instead to set exchange rates.
PL-12255	Corrected an issue that could potentially cause a thread deadlock in the <code>GeocodeUtils.getCurrentGeocodePlugin</code> method.
PL-12464	Fixed an issue that caused an attempt to access bean of type "GroupUser" with a null bundle error when retrying the <code>assignGroupDynamically</code> and the <code>assignUserDynamically</code> assignment methods after a Validation warning.
PL-12989	Fixed an issue that caused the application to not be able to load root group extension definition during initial upgrade of an empty database or from the <code>import_tools</code> command.
PL-13733	Fixed an issue that caused a CDCE exception (ConcurrentDataChangeException) due to stale cache data.

ID	Description
PL-14130	Corrected an issue that caused policies with a large number of data entities (such common commercial policies) to experience significant performance issues. Policies with root entities that had a large number of child entities (a location that contained a large number of buildings, for example) were especially vulnerable to this type of performance issue.
PL-14322	Fixed an error with French locale using comma as a decimal separator. The CurrencyValueWidget improperly reformatted numbers from French locale to U.S. locale when running verifications.
<b>Database</b>	
PL-12629	0 is now a valid value for the attribute samplingpercentage in <databasestatistics> element (used with the <database> element in config.xml). It translates to a statistics estimation percentage of DBMS_STATS.AUTO_SAMPLE_SIZE in Oracle and 100% or FULLSCAN in Microsoft SQL Server.
PL-14389	Corrected an issue with SQL Server that occurred during recreation of primary keys with backing clustered indexes during an upgrade. The process did not account for self-referencing foreign keys as it dropped foreign keys prior to recreating the primary keys.
PL-14446	Corrected an issue that occurred if moving to clustered indexes on SQL Server during an upgrade and using file groups, the application moved the data from the original file group. The application now leaves the data in the original file group when moving to clustered indexes during an upgrade.
PL-14483	On application startup, PolicyCenter checks the database version against the required major version and minimum minor version. PolicyCenter generates an error if the major version does not match or if the minor version is less than the minimum minor version supported. This is a change from previous (pre-7.0.0) releases as later minor release versions would also generate an error. See the "Database Patch Strategy 2011" whiteboard or the Platform Support Matrix available on the Guidewire Resource Portal for more information.
<b>Gosu</b>	
PL-10812	Corrected an issue in which a finder in a code annotation caused Studio to throw an error, and fail to perform syntax error highlighting and code completion. This occurred even if Studio was connected to the application server (meaning the database).
<b>Integration</b>	
PL-13638	<p>Added support for the XML simple value EncodedString.</p> <p>Added the ability to access generated XSD files used for archiving at the following location:  <a href="http://servername:port/xx/service/xsd">http://servername:port/xx/service/xsd</a></p> <p>For example, for PolicyCenter, this is <a href="http://localhost:8180/xx/service/xsd">http://localhost:8180/xx/service/xsd</a>.</p>
PL-14984	See "Change in Web Service Behavior for Committing Data to the Database (PL-14984)" on page 241 for details.
<b>Reporting</b>	
PL-14418	Corrected an issue that caused a report to freeze if that report contained freehand tables along with large data sets. This issue was due to a bug in the InetSoft v9 reporting software.
<b>Studio</b>	
PL-10701	Improved performance with regards to the amount of time that it took to create a new class in Studio.
PL-12877	Fixed an issue that caused a Null Pointer Exception and Studio to stop responding. This occurred if you attempted to create a new rule category with the same name, but different case, of an existing rule category.
PL-13486	In Studio, as you switch between tabs, close tabs, press ALT-F1 to find in the project tree, and perform other related actions, Studio performs a search to find the node in the project tree that corresponds to the active editor. This fix improves performance in Studio as you switch between tabs, close tabs, and perform similar actions.
PL-13623	Fixed an issue where the PCF editor object was not garbage-collected after the PCF editor window was closed. This could cause sluggish Studio behavior or possibly even out-of-memory issues in long Studio sessions in which you created or modified a large number of PCF files.
PL-13626	Improved performance with respect to creating new classes, new PCF files, a new rule set, and new rules in a new rule set in Studio.
PL-14214	Reinstated the requirement that a Studio user must have the Administer Rules permission (code, ruleadmin) to access a running production server.

## 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 if they 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
- Platform Known Issues

### PolicyCenter Known Issues

#### [Studio: “Verify Changed Resources” is not implemented in the product model \(PC-6928\)](#)

**Issue:** The Studio Tools → Verify → Verify Changed Resources command does not work with the PolicyCenter product model.

**Workaround:** Use one of the other verification commands instead. See “Validating Studio Resources” in the *PolicyCenter Configuration Guide* for details.

#### [Duplicate contact when merging contacts through API \(PC-7421\)](#)

**Issue:** When you merge two instances of a contact through the API, you may get duplicate contacts for the following contact roles:

- PolicySecNamedInsured
- PolicyPriNamedInsured
- PolicyBillingContact

This was previously logged as CC-56177.

**Status:** Guidewire is aware of this issue.

#### [EvaluationException: \[251861\] in Account tab \(PC-7861\)](#)

**Issue:** Receive the following exception when creating a new submission from a newly created company account:

EvaluationException: [251861] Cannot load a temporary id from the backing entity source: Account

**Workaround:** In Studio, navigate to Page Configuration (PCF) → account → new. Click to open CreateAccount.pcf, and select the Code tab. Move the first line of the beforeCommitNewAccount() method (which calls setLastProducerSelection) to be the first line of the afterCommitNewAccount() method.

#### [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. This was previously logged as CC-54957.

**Status:** Guidewire is aware of this issue.

#### [Need a summary of underwriting issues across all policies in a job group \(PC-9102\)](#)

**Issue:** The JobGroupRiskAnalysisLV needs to show a summary of underwriting issues across all policies in the job group. This was previously logged as CC-54177.

**Status:** Guidewire is aware of this issue.

## 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. 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 original answer is still saved along with the other answers.

**Workaround:** The answers to these *irrelevant* questions 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)
        }
    }
}
```

## Duplicate descriptions for image/JPG mime types in the configuration file (PC-12320)

**Issue:** In the `config.xml` file, there are two entries under MIME types for jpeg: one for `pjpeg` and one for normal `jpeg`s. This causes a problem as the descriptions of both are identical. If you upload a normal jpeg and chooses the wrong jpeg MIME type, an error occurs.

**Workaround:** Change the description of the `pjpeg` so that duplicate entries do not appear in the MIME type drop down menu in Guidewire Studio.

## 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 screen of the agent
- 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.

## Platform Known Issues

### First time you click on the arrow of the typekey input, the drop-down menu will not open (PL-10134)

**Issue:** The drop-down menu does not open on the first click of the arrow on a typekey input. Instead, the help text opens.

**Workaround:** Turn off help text on focus by setting `InputHelpTextOnFocus` to `false` in the `config.xml` file. By doing that, the help text shows only if you mouse over the input and will not interfere with opening a drop down menu.

#### [Studio Rules do not use correct capitalization for root object's name \(PL-10740\)](#)

**Issue:** Rule set root objects are not named with first letter lower-cased.

**Workaround:** Rules engine will issue warnings when correct case for objects is not being used.

#### [Studio—with Subversion \(SVN\)—incorrectly copies .svn files \(PL-10932\)](#)

**Issue:** *This issue affects installations that use Subversion only.* If you modify the base configuration rules in Studio, Studio creates a copy of the rule in the `modules/configuration` folder. Studio then executes Subversion commands to add the files and folders to Subversion. In the process, Studio incorrectly copies the `.svn` folder and its children. This leads to Subversion problems.

**Workaround:** Manually delete the copied `.svn` folders.

#### [Geocode plugin calls `isSufficientlyCompleteToGeocode` method twice for each address \(PL-11578\)](#)

**Issue:** The Geocode plugin calls the `isSufficientlyCompleteToGeocode` method twice for each address.

**Workaround:** Guidewire is aware of this issue.

#### [Countries configured in zone-config.xml still generate a warning during regen-dictionary even when zone data is loaded for all of these countries \(PL-11947\)](#)

**Issue:** Countries configured in `zone-config.xml` still generate a warning during regen-dictionary even when zone data is loaded for all of these countries.

**Workaround:** Warning message is created in error and can safely be ignored.

#### [User interface cannot handle starting multiple instances of a batch process \(PL-12372\)](#)

**Issue:** The user interface cannot handle multiple instances of a batch process.

**Workaround:** If you need to execute multiple instances of a batch process, then you need to start each from the command line. Also, you need to ensure that the `BatchProcess.isExclusive` method returns `false` to allow the application to run multiple instances simultaneously.

#### [Type system refresh after PCF page title change does not update corresponding menu label \(PL-13057\)](#)

**Issue:** The type system refresh after the PCF page title change does not update corresponding menu label.

**Workaround:** After updating a page title, the server restart must be done to refresh menu labels and avoid null pointer exceptions due to stale reference.

#### [Length limitation on entity localization table names \(PL-13360\)](#)

**Issue:** There is a length limitation on entity localization table names.

**Workaround:** Ensure that `localization tableName` property specified in the entity extension file is less than 16 characters. The error message generated if the localization table name exceeds the maximum length indicates that 18 characters are allowed, but that does not account for two additional characters added by the application.

### *US-Locations.txt* file with the U.S. geodata from GreatData has special characters that cause validation problems with United States Postal Service (USPS) data (PL-13384)

**Issue:** The *US-Locations.txt* file contains information that does not conform to United States Postal Service (USPS) standards for bulk mailings.

**Workaround:** The provided *US-Locations.txt* file is intended only for use in geocoding to identify addresses for a location. You can process the *US-Locations.txt* file to conform to your particular address standards, and then import that version of the file instead.

### Use of *StringBuilder* class prevents debugging of web service (PL-13435)

**Issue:** Suppose that you have a web service in which you have a method with a parameter of type X and inside the web service method you reference a Gosu class of type Y. This can be an issue if the Gosu class Y has a *toString* method (overriding the default) that uses a *java.lang.StringBuilder* class. If you attempt to set a breakpoint in the web service method and try to step through the method, Studio generates errors.

**Workaround:** Studio does not generate errors if you call the web service in normal run mode rather than in debug mode.

### GX models that reference virtual fields and enhancements throw null pointers if null (PL-13560)

**Issue:** The GX models that reference virtual fields and enhancements throw null pointers when they are null.

**Workaround:** Ensure that null checks and error handling is included so that if referenced virtual fields or enhancements are null, then there will not be a null pointer exception.

### Email with file attachment with unicode filename not correctly handed over to the mail server (PL-13582)

**Issue:** Sending email with file attachment with unicode filename is not sent to the mail server correctly.

**Workaround:** Use Latin characters for file names on attached files.

### The GX model generated XSD cannot be parsed by JAXB (PL-13598)

**Issue:** The GX model generated XSD cannot be parsed by JAXB.

**Workaround:** Add JAXB annotation elements to the XSD to specify the necessary metadata (such as class names) for JAXB to generate the java class files. Contact Guidewire Support for an example XSD annotated in this way.

### *JavaToolkit.gs* has incorrectly hard coded memory which results in failed *regen-java-api* Ant task (PL-13663)

**Issue:** The *JavaToolkit.gs* has hard coded memory which can result in failed *regen-java-api* Ant tasks.

**Workaround:** Increase the size of the maximum heap setting on line 161 of *JavaToolkit.gs* in the Ant module. The default value is 512.

### Cannot make a field from a delegate a localized column (PL-13761)

**Issue:** You cannot make a field from a delegate a localized column.

**Workaround:** Move the column to be localized off the delegate and onto each of the implementing entities. Then, in order to make the column appear as though it exists on the delegate, define an enhancement property on the delegate, that *delegates* to the appropriate column, depending on the implementing entity.

## Copy and paste does not work from phone fields (PL-13792)

**Issue:** The copy and paste does not work for phone fields on a contact.

**Workaround:** Save the contact with the newly entered phone information before attempting to either copy or paste the value into another field.

## Cannot deploy custom *web.xml* file (PL-14482)

**Issue:** PolicyCenter does not deploy a custom *web.xml* file properly.

**Workaround:** Before attempting to deploy the application, do the following:

1. Open the following file for editing:

```
PolicyCenter\modules\ant\public\gw\ant\deploy.gs
```

2. Search for the following lines of code:

```
var deployDir = module.Dir.file("deploy")
if (deployDir.exists()) {
    new Copy() { :FileSet = deployDir.fileSet(), :ToDir = destDir }
        .execute()
}
```

3. Make the following change:

```
var deployDir = module.Dir.file("deploy")
if (deployDir.exists()) {
    new Copy() { :FileSet = deployDir.fileSet(), :ToDir = destDir, :Overwrite = true }
        .execute()
}
```

Notice that you need to add `:Overwrite = true` to the `Copy` method.

## Cannot print second-level list view (PL-14640)

**Issue:** Printing a second-level list view does not work correctly. The print job contains duplicates and does not include all of the items.

**Workaround:** Remove the lower level list view out of the hierarchical structure and include it at the top level.

## Superuser role unable to edit or delete calendar holidays (PL-14942)

**Issue:** It is not possible to edit or delete a calendar holiday as superuser.

**Workaround:** Log into PolicyCenter using su to be able to edit the holiday schedule.

## *RowSet* and *RowIterator* configured in the same *ListView* do not work correctly. (PL-14946)

**Issue:** Configuring a *RowSet* and *RowIterator* in the same *ListView* does not provide the right behavior for editability.

**Workaround:** Separate the *RowSet* and *RowIterator* that exist in the same *ListView* into separate components or into multiple *ListView* widgets.

## Masked input fields do not correctly handle ENTER keypress (PL-14955)

**Issue:** Input fields that have a field validator mask do not correctly handle the ENTER keypress. After you press ENTER, the field no longer displays the input mask and it is not possible to enter any further input.

**Workaround:** Cancel the page and re-enter the information.

## PolicyCenter does not recognize modified *Calendar.js* file (PL-14997)

**Issue:** PolicyCenter does not correctly recognize changes made to *Calendar.js* through Studio. You access this file in the following Studio location:

Resources → Web Resources → resources → javascript → global

**Workaround:** Instead, modify `global.js` directly. You access this file in the following installation file location:  
`PolicyCenter/webapps/pc/resources/javascript`

---

**WARNING** This is one of the very few occasions in which you modify an application file outside of the application configuration folder. Ensure that you modify `global.js` only. Otherwise, you can invalidate your installation.

---

#### [Studio debugger cannot debug `gw.sampledata` classes \(PL-15357\)](#)

**Issue:** Attempting to debug a test that loads or uses base application sample data causes an unimplemented bytecode error in the JVM. This means that you cannot use the Studio debugger to debug `gw.sampledata` classes.

**Workaround:** Guidewire is aware of this issue.

# Guidewire PolicyCenter 4.0.6 Release Notes

## Release 4.0.6.3

**Document Revision:** 20-May-2014

Copyright © 2001-2014 Guidewire Software, Inc. All rights reserved.

Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Reinsurance Management, Guidewire ContactManager, Guidewire Vendor Data Management, Guidewire Client Data Management, Guidewire Rating Management, Guidewire InsuranceSuite, Guidewire ContactCenter, Guidewire Studio, Guidewire Product Designer, Guidewire Live, Guidewire DataHub, Guidewire InfoCenter, Guidewire Standard Reporting, Guidewire ExampleCenter, Guidewire Account Manager Portal, Guidewire Claim Portal, Guidewire Policyholder Portal, ClaimCenter, BillingCenter, PolicyCenter, InsuranceSuite, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks, service marks, or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries.

All other trademarks are the property of their respective owners.

**This material is confidential and proprietary to Guidewire and subject to the confidentiality terms in the applicable license agreement and/or separate nondisclosure agreement.**

Guidewire products are protected by one or more United States patents.

## Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Support

- Changes in This Release
- Improvements and General Issues
- Known Issues and Limitations

## Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

### Version Number

- This release of Guidewire PolicyCenter is 4.0.6.3.
- If you are using the integration with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 3.0.1 or later.
- If you are using the integration with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 6.0.0 or later.
- For Standard Reporting, this release of PolicyCenter supports InetSoft StyleReport Enterprise Edition v10.1 20100113.

## 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

The following issues pertain to installing this release.

### InetSoft Evaluation

You can evaluate InetSoft Style Report Enterprise Edition Version 10.1 using the following license key:

L000-72C-ERX-0000BF00100001F-F2B025280327

If you want to use InetSoft and the PolicyCenter reporting module in a production environment, contact Guidewire Customer Support to request a full license key. For more information, see the *PolicyCenter Reporting Guide*.

---

**WARNING** The build number of the reporting install files must match the build number of the Guidewire PolicyCenter installation. If the build numbers do not match, you will have problems with your reporting installation.

---

## Support

For assistance, visit the Guidewire Resource Portal – <http://guidewire.custhelp.com>

## Changes in This Release

This section describes the product changes in this release.

- Base PCF File Changes

- Custom Upgrade Diff Report

## Base PCF File Changes

All links below require the `ReleaseNotes_files` directory on your local disk.

### PolicyCenter release 4.0.5 to 4.0.6

- To view a report of the changes in the base PCF files in the `modules/pc` directory, [click here](#).
- To view a report of the changes in the base PCF files in the `modules/pl` directory, [click here](#).

## Custom Upgrade Diff Report

Guidewire provides a custom report detailing certain differences between the current release and the prior minor release. This report describes changes in display keys, entities, typelists, and the Gosu API. This custom report dynamically conforms to each customer's particular upgrade path. In addition, the report is regularly updated as the tools for generating it are enhanced to provide more information.

This report is available on the Guidewire Resource Center web portal.

## Improvements and General Issues

This topic describes the primary improvements and issues corrected in this release. Guidewire attempts to provide information for issues of primary importance to our customers. This is not meant to be a comprehensive list. Many performance related issues were addressed in this release but are not listed here.

ID	Description
<b>Account</b>	
PC-14159	Improved performance of Account Search by Last Name.
PC-14175	The ZIP Code is now a minimal search criteria for Account Search since it is no less restrictive than City and State, which was allowed as a minimal search criteria.
PC-14192	Fixed a performance issue in Account searches.
<b>Batch Processing</b>	
PC-14813	If the <code>ConcurrentDataChangeException</code> occurs on processing a work item, system logs the error only if the number of attempts exceeds the configured value of the <code>WorkItemRetryLimit</code> .
<b>Core</b>	
PL-12485	Guidewire no longer exposes internal implementation details of a query through the undocumented <code>RawQuery</code> property of the query builder APIs. Guidewire has removed this property.
PL-13071	Added the ability to verify PCF files outside of Guidewire Studio through the use of <code>gwpc verify-types</code> command line utility.
PL-14589	Guidewire has implemented a new batch process, <code>ProcessCompletionMonitor</code> . The process wakes up at user-configurable intervals and examines the Process History and Work Queue for all batch processes. After the batch process completes, it invokes user plugin <code>IBatchCompletedNotification</code> to permit user code to react to the completion of the batch process. This process also marks <code>ProcessHistory.NOTIFICATIONSENT = true</code> for that batch process to ensure that the plugin is invoked only once.
PL-16856	Created a database consistency check to detect certain problems with <code>EffDated</code> localization entities. These include: <ul style="list-style-type: none"><li>• Out-of-sequence changes that result in multiple entries for the same language.</li><li>• Gaps in the revision history for localization entities. Setting the value to <code>null</code> causes gaps in the revision history. In general, gaps in the revision history are acceptable for <code>EffDated</code> entities, but they are not acceptable for localization entities.</li></ul>
PL-16857	Fixed a problem in which out-of-sequence changes to localized columns caused data inconsistencies for non-default locales. This fix changes how PolicyCenter handles localized columns in diff reports. Diffs now report changes in non-default locales as normal property changes.

<b>ID</b>	<b>Description</b>
<b>Database</b>	
PL-16620	<p>Database parameter checker.threads (in config.xml) controls the number of threads used to execute database consistency checks (DBCC). Previously, you could only control the number of threads using the synchronous methods of running DBCC, from the system_tools -checkdbconsistency command line and at application start-up. With this change, you can override the number of threads if you are running consistency checks as a batch job.</p> <p>If you set a value for checker.threads in config.xml, then PolicyCenter displays the configured number of threads on the (System Tools) Info Page → Consistency Checks page. You can override the configured value with a positive integer and submit new consistency checks with this new value on that page.</p> <p>If you do not set a value for checker.threads in config.xml, then PolicyCenter uses 1 as the default value and displays that value on the Consistency Checks page. You can still override this value on that page.</p>
PL-16762	Guidewire has added a new <database> parameter in config.xml that you use—in Oracle-based implementations—to disable rewrite queries using materialized views at a session level. The name of the parameter is queryRewriteEnabled. Set its value to false to disable rewrite queries.
PL-17133	Fixed an issue in which PolicyCenter would open a database connection and never close it if accessing an external data source using JNDI. This prevented database fail-over. This fix changes this behavior so PolicyCenter closes all connections if they are no longer in use.
<b>Data Model</b>	
PC-14839	The database report for policy period now includes a row for the number of temporary branches that exist. You can use this to determine if there is something happening with temporary branches. This number should remain relatively small.
<b>Financials</b>	
PC-14564	<p>The creation of cost amounts for rating, and the proration of those amounts when adjusting the rate for a partial period, were done using different formulas. For consistency, they now use the same formula.</p> <p>During merge, you may notice a change to methods on the Gosu class CostData and its subclasses.</p>
PC-14679	Fixed a problem in the processing of financial transactions which sometimes resulted in a reversal and new onset rather than simply a partial offset transaction. This problem did not impact correctness.
PC-14759	<p>The behavior of Cost.OverrideAmount and Cost.OverrideTermAmount has changed. In particular:</p> <ul style="list-style-type: none"> <li>• OverrideTermAmount is no longer scaled if the term length changes.</li> <li>• OverrideAmount on a cost is also rounded the same way the rating engine would round the standard amount.</li> </ul>
<b>Geocoding</b>	
PL-13499	<p>Guidewire has added an implementation of the Geocode plugin that connects to the Microsoft Bing Maps Geocode Service. The Bing Maps plugin implementation replaces the Microsoft MapPoint implementation.</p> <p>Guidewire deprecates the MapPoint implementation with this release. Microsoft announced plans to retire the MapPoint web service, effective November 18, 2011. If you currently use geocoding features and the MapPoint plugin, you must migrate from MapPoint to Bing Maps before November 18, 2011. Otherwise, geocoding features in the application cease to function on November 18, 2011, and afterwards.</p>
<b>Jobs</b>	
PC-14043	<p>Before a non-submission job is started, PolicyCenter creates a new policy period by cloning the based-on period. PolicyCenter also copies the JobID of the based-on period, so for a brief time the new policy period belongs to the job of the old based-on period. The cloning process commits the row to the database, so if an exception then happens before the new job has had a chance to start, the row is left in this inconsistent state with a temporary period attached to an already bound job. This can cause a variety of problems including rendering the Policy unusable.</p> <p>The fix for this was to filter out the temporary branches such that they are no longer visible from the application layer outside of custom or direct SQL queries by default. That is, using find or query will not return temporary branches. Temporary branches can be explicitly included if necessary using setFindTemporaryBranches(true) on the restriction.</p>
PC-14369	Fixed an issue that occurred after issuing a policy change. The link to a future rewritten renewal could go to a policy period with a flat cancellation. Now the link goes to the subsequently rewritten period, if any.
PC-14853	Fixed an issue that caused a stack dump when changing the primary named insured in an out-of-sequence policy change.
<b>Lines of Business - CPP</b>	

ID	Description
PC-14058	Fixed a null pointer exception on quote that occurred if you removed the general liability line from a commercial package policy in a policy change, then added general liability additional insureds in an out-of-sequence change.
<b>Lines of Business - WC</b>	
PC-13652	Fixed an issue where a policy change in a workers' compensation policy caused PolicyCenter to throw a duplicate key exception with a message that RatingPeriodStartDate already exists.  Removed the unique index on RatingPeriodStartDate as it causes duplicate key exceptions in certain cases where duplicate keys are valid. To compensate for this removed index, there is an additional validation in WCJurisdictionValidation. There is also a new database consistency check which detects duplicate RatingPeriodStartDate records.
<b>Product Model</b>	
PC-13988	Resolved an issue with updating the questions displayed from a question set when the availability of the questions has changed.
PC-14034	Updated the PolicyCondition entity to implement the ProductModelBasedEntity interface, so that GX Modeler can access virtual properties added by the product model.
PC-14209	Fixed an issue where availability lookup rows were not always being interpreted correctly. This could lead to situations such as a row added to a businessowners policy lookup table in Studio causing an exception during a personal auto quote.
PC-14634	Differences between the system table data stored in the XML files and the corresponding entity definition will now be caught by both Studio verification of resources and server start-up verification.
<b>Rating</b>	
PC-13269	In Studio, you set the Quote Rounding Level on the Basics tab of a product defined in Product Model → Products. If necessary, you can configure PolicyCenter to use different rounding levels within the product. For example, in the default configuration for personal auto, the quote rounding level is set to 0 for the product. However, the quote rounding level for taxes is 2 in New York and Florida. The rateTaxes method in gw.lob.pa.rating.PARatingEngine sets the quote rounding mode for taxes in those states. You can use this method as an example.  This issue was fixed in PolicyCenter 4.0.5.
PC-14682	In previous versions of PolicyCenter, the policy data could be modified while a quote (a set of costs) was being generated. In the event that the policy data was modified during the time that a quote was being generated (either internally or through an integration), it was possible that there would be inconsistencies between the values of rating parameters on the policy and for the costs returned. This would be relatively unlikely with few-second quote times, but would be more likely with longer quote times, such as with complex commercial policies or when using external rating engines.  The new rating code locks only the PolicyPeriod and the effective dated entities beneath it. As before, rating should only be dependent on parameters reflected in these entities. If you are using non-persisted entities for generating quotes, then you will likely need a different rating path that does not commit to the database, as this rating code does.
<b>Revisioning</b>	
PC-13616	Fixed an issue where adding or removing a line from a policy in an out-of-sequence policy change resulted in a null pointer exception.
PC-13984	In certain circumstances in an out-of-sequence policy change, entities were incorrectly removed from a future slice. This occurred in the following situation. Entity A has a foreign key to entity B in the current slice of the PolicyPeriod. In a future slice: <ul style="list-style-type: none"><li>• Entity B has been removed.</li><li>• Entity A no longer has a foreign key to entity B.</li></ul> Entity A would be incorrectly removed from that future slice if entity A was edited in the current slice.  For example, a vehicle is garaged at Location A in the current slice, then garaged at Location B in a future slice. If the vehicle was edited in the current slice and location A did not exist the future slice, the vehicle would be incorrectly removed and no longer exist in the future slice.
PC-14125	Added a DBConsistency check to make sure there are no orphan temporary branches in the database.
PC-14208	Fixed an issue that occurred when completing a policy change that resulted in a preemption. In certain cases, PolicyCenter incorrectly updated the term number to 0. If you were integrated with BillingCenter, this sent an invalid term number to BillingCenter.

ID	Description
<b>Studio</b>	
PC-14350	Fixed an issue where a stale reference to the product model resulted in PolicyCenter throwing an error with a message that the <code>productmodel.PersonalAutoLine</code> has been discarded and cannot be referenced.
PC-14541	Fixed an issue where the user incorrectly received a message that a coverage term code was already in use. The user received this message after deleting an option, package, or typekey coverage term pattern and then adding a coverage term pattern using the same code.
<b>User Interface</b>	
PC-13818	Fixed a user interface issue where temporary periods were shown as duplicates in <code>PolicyFile</code> or threw an exception. The fix included changes to PCF files to filter out jobs with status <code>Temporary</code> and added a check for the <code>CopySubmission</code> permission.
<b>Web</b>	
PL-16711	Improved performance by modifying the UI-rendering JavaScript so that it now only resizes visible widgets.

## 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 if they 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
- Platform Known Issues

### PolicyCenter Known Issues

#### Studio: “Verify Changed Resources” is not implemented in the product model (PC-6928)

**Issue:** The Studio Tools → Verify → Verify Changed Resources command does not work with the PolicyCenter product model.

**Workaround:** Use one of the other verification commands instead. See “Validating Studio Resources” in the *PolicyCenter Configuration Guide* for details.

#### Duplicate contact when merging contacts through API (PC-7421)

**Issue:** When you merge two instances of a contact through the API, you may get duplicate contacts for the following contact roles:

- `PolicySecNamedInsured`
- `PolicyPriNamedInsured`
- `PolicyBillingContact`

**Status:** Guidewire is aware of this issue.

#### EvaluationException: [251861] in Account tab (PC-7861)

**Issue:** Receive the following exception when creating a new submission from a newly created company account:

EvaluationException: [251861] Cannot load a temporary id from the backing entity source: Account

**Workaround:** In Studio, navigate to Page Configuration (PCF) → account → new. Click to open CreateAccount.pcf, and select the Code tab. Move the first line of the beforeCommitNewAccount() method (which calls setLastProducerSelection) to be the first line of the afterCommitNewAccount() method.

### 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. This was previously logged as CC-54957.

**Status:** Guidewire is aware of this issue.

### Need a summary of underwriting issues across all policies in a job group (PC-9102)

**Issue:** The JobGroupRiskAnalysisLV needs to show a summary of underwriting issues across all policies in the job group. This was previously logged as CC-54177.

**Status:** Guidewire is aware of this issue.

### 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. 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 original answer is still saved along with the other answers.

**Workaround:** The answers to these *irrelevant* questions 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)
        }
    }
}
```

### Duplicate descriptions for image/JPG mime types in the configuration file (PC-12320)

**Issue:** In the config.xml file, there are two entries under MIME types for jpeg: one for pjpeg and one for normal jpgs. This causes a problem as the descriptions of both are identical. If you upload a normal jpeg and chooses the wrong jpeg MIME type, an error occurs.

**Workaround:** Change the description of the pjpeg so that duplicate entries do not appear in the MIME type drop down menu in Guidewire Studio.

### 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 screen of the agent
- 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.

#### Contact.AccountHolderCount incorrectly counts the number of accounts (PC-14736)

**Issue:** Contact.AccountHolderCount incorrectly counts the number of accounts a particular contact is the account holder on.

**Workaround:** Query the database to count the accounts on which this contact has the account holder role.

### Platform Known Issues

#### First time you click on the arrow of the typekey input, the drop-down menu will not open (PL-10134)

**Issue:** The drop-down menu does not open on the first click of the arrow on a typekey input. Instead, the help text opens.

**Workaround:** Turn off help text on focus by setting `InputHelpTextOnFocus` to `false` in the `config.xml` file. By doing that, the help text shows only if you mouse over the input and will not interfere with opening a drop down menu.

#### Studio Rules do not use correct capitalization for root object's name (PL-10740)

**Issue:** Rule set root objects are not named with first letter lower-cased.

**Workaround:** Rules engine will issue warnings when correct case for objects is not being used.

#### Geocode plugin calls 'isSufficientlyCompleteToGeocode' method twice for each address (PL-11578)

**Issue:** The Geocode plugin calls the `isSufficientlyCompleteToGeocode` method twice for each address.

**Workaround:** Guidewire is aware of this issue.

#### Countries configured in 'zone-config.xml' still generate a warning during regen-dictionary even when zone data is loaded for all of these countries (PL-11947)

**Issue:** Countries configured in `zone-config.xml` still generate a warning during regen-dictionary even when zone data is loaded for all of these countries.

**Workaround:** Warning message is created in error and can safely be ignored.

#### User interface cannot handle starting multiple instances of a batch process (PL-12372)

**Issue:** The user interface cannot handle multiple instances of a batch process.

**Workaround:** If you need to execute multiple instances of a batch process, then you need to start each from the command line. Also, you need to ensure that the `BatchProcess.isExclusive` method returns `false` to allow the application to run multiple instances simultaneously.

## Type system refresh after PCF page title change does not update corresponding menu label (PL-13057)

**Issue:** The type system refresh after the PCF page title change does not update corresponding menu label.

**Workaround:** After updating a page title, the server restart must be done to refresh menu labels and avoid null pointer exceptions due to stale reference.

## Length limitation on entity localization table names (PL-13360)

**Issue:** There is a length limitation on entity localization table names.

**Workaround:** Ensure that localization `tableName` property specified in the entity extension file is less than 16 characters. The error message generated if the localization table name exceeds the maximum length indicates that 18 characters are allowed, but that does not account for two additional characters added by the application.

## 'US-Locations.txt' file with the U.S. geodata from GreatData has special characters that cause validation problems with United States Postal Service (USPS) data (PL-13384)

**Issue:** The `US-Locations.txt` file contains information that does not conform to United States Postal Service (USPS) standards for bulk mailings.

**Workaround:** The provided `US-Locations.txt` file is intended only for use in geocoding to identify addresses for a location. You can process the `US-Locations.txt` file to conform to your particular address standards, and then import that version of the file instead.

## GX models that reference virtual fields and enhancements throw null pointers if null (PL-13560)

**Issue:** The GX models that reference virtual fields and enhancements throw null pointers when they are null.

**Workaround:** Ensure that null checks and error handling is included so that if referenced virtual fields or enhancements are null, then there will not be a null pointer exception.

## Email with file attachment with unicode filename not correctly handed over to the mail server (PL-13582)

**Issue:** Sending email with file attachment with unicode filename is not sent to the mail server correctly.

**Workaround:** Use Latin characters for file names on attached files.

## The GX model generated XSD cannot be parsed by JAXB (PL-13598)

**Issue:** The GX model generated XSD cannot be parsed by JAXB.

**Workaround:** Add JAXB annotation elements to the XSD to specify the necessary metadata (such as class names) for JAXB to generate the java class files. Contact Guidewire Support for an example XSD annotated in this way.

## 'JavaToolkit.gs' has incorrectly hard coded memory which results in failed 'regen-java-api' Ant task (PL-13663)

**Issue:** The `JavaToolkit.gs` has hard coded memory which can result in failed `regen-java-api` Ant tasks.

**Workaround:** Increase the size of the maximum heap setting on line 161 of `JavaToolkit.gs` in the Ant module. The default value is 512.

## Cannot make a delegate field a localized column (PL-13761)

**Issue:** You cannot make a delegate field a localized column.

**Workaround:** Move the column that you want to localize off the delegate and onto each of the implementing entities. Then, in order to make the column appear as though it exists on the delegate, define an enhancement property on the delegate, that *delegates* to the appropriate column, depending on the implementing entity.

### [Cannot print second-level ListView \(PL-14640\)](#)

**Issue:** Printing a second-level list view does not work correctly. The print job contains duplicates and does not include all of the items.

**Workaround:** Remove the lower level list view out of the hierarchical structure and include it at the top level.

### [RowSet and RowIterator configured in the same ListView do not work correctly. \(PL-14946\)](#)

**Issue:** Configuring a RowSet and RowIterator in the same ListView does not provide the right behavior for editability.

**Workaround:** Separate the RowSet and RowIterator that exist in the same ListView into separate components or into multiple ListView widgets.

### [Masked input fields do not correctly handle ENTER keypress \(PL-14955\)](#)

**Issue:** Input fields that have a field validator mask do not correctly handle the ENTER keypress. After you press ENTER, the field no longer displays the input mask and it is not possible to enter any further input.

**Workaround:** Cancel the page and re-enter the information.

### [PolicyCenter does not recognize modified 'Calendar.js' file \(PL-14997\)](#)

**Issue:** PolicyCenter does not correctly recognize changes made to Calendar.js through Studio. You access this file in the following Studio location:

Resources → Web Resources → resources → javascript → global

**Workaround:** Instead, modify global.js directly. You access this file in the following installation file location:  
PolicyCenter/webapps/pc/resources/javascript

**WARNING** This is one of the very few occasions in which you modify an application file outside of the application configuration folder. Ensure that you modify global.js only. Otherwise, you can invalidate your installation.

### [Studio debugger cannot debug 'gw.sampledata' classes \(PL-15357\)](#)

**Issue:** Attempting to debug a test that loads or uses base application sample data causes an unimplemented bytecode error in the JVM. This means that you cannot use the Studio debugger to debug gw.sampledata classes.

**Workaround:** Guidewire is aware of this issue.

### [Configuration Upgrade Tool does not copy XSD files correctly \(PL-16754\)](#)

**Issue:** As part of the upgrade from ClaimCenter 5.0.x to ClaimCenter 6.0.x, Guidewire requires that you run the Configuration Upgrade Tool, using the following command:

```
ant -f upgrade.xml upgrade
```

This script works incorrectly and does not copy various XSD files to the correct target directory properly.

**Workaround:** After running the Configuration Upgrade Tool automated steps, copy all XSD files from the following ClaimCenter 5.0.x directory:

```
...\\configuration\\config\\registry\\
```

to the following ClaimCenter 6.0.x directory:

.....\modules\configuration\gsrc\

### Issues with Internet Explorer 9

**Issue:** If you are using the Internet Explorer 9 browser, you may notice issues such as screen flickering or an incorrect tab order for fields. According to public reports, IE 9 exhibits these and other issues with a variety of web sites and web applications.

**Workaround:** Because this is the behavior of the Internet Explorer 9 rendering engine, Guidewire cannot address these issues. However, there are reports of IE 9 users being able to reduce these issues by changing the new IE 9 Accelerated Graphics settings on the Advanced tab of the [Internet Options](#) dialog.



# Guidewire PolicyCenter 7.0.0

## Release Notes

### Release 7.0.0.429

Copyright © 2001-2011 Guidewire Software, Inc. All rights reserved. Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Reinsurance Management, Guidewire ContactManager, Guidewire Vendor Data Management, Guidewire Client Data Management, Guidewire InsuranceSuite, Guidewire ContactCenter, Guidewire Studio, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries.

**This documentation is published as Guidewire Confidential.** The contents of this documentation, including product architecture details and APIs, are considered confidential and are fully protected by customer licensing confidentiality agreements and signed Non-Disclosure Agreements (NDAs).

This product includes software developed by the Apache Software Foundation (<http://www.apache.org>).

## Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Known Issues and Limitations

## Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

---

**IMPORTANT** If you missed 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 7.0.0.429.

- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 3.0.1 or later.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 6.0.0 or later.

### Changes to Application Server Requirements

This release of PolicyCenter does not support the WebSphere or WebLogic application servers. See the *PolicyCenter Installation Guide* for the complete list of supported application servers.

**Note:** Contact your sales representative if you require support for the WebSphere or WebLogic application servers.

### Support

This document includes important information about the current release of Guidewire PolicyCenter. For assistance with this software release, contact Guidewire Customer Support:

- At the Guidewire Resource Center – <http://guidewire.custhelp.com>
- By email – [support@guidewire.com](mailto:support@guidewire.com)
- By phone – +1-650-356-4955

## 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

## Changes in This Release

This section describes the product changes in this release.

- Improvements to Upgrade Diff Report

### Improvements to Upgrade Diff Report

With previous releases, Guidewire provided a static report in the release notes detailing certain differences between the current release and the prior minor release. This report described changes in display keys, entities, typelists, and the Gosu API. Guidewire has improved this report to dynamically conform to each customer's

particular upgrade path. In addition, the report is regularly updated as the tools for generating it are enhanced to provide more information.

Because the new report is tailored to your particular upgrade requirements, it is no longer included with the general release notes. To obtain your custom Upgrade Diff Report, contact your Guidewire representative. In the future, this report will be available on the Guidewire Resource Center web portal, allowing you to view the latest version right when you need it.

## Known Issues and Limitations

This section describes known issues with this release of Guidewire PolicyCenter.

- PolicyCenter Known Issues
- Platform Known Issues

### PolicyCenter Known Issues

#### [Studio: “Verify Changed Resources” is not implemented in the product model \(PC-6928\)](#)

**Issue** – The Studio Tools → Verify → Verify Changed Resources command does not work with the PolicyCenter product model.

**Workaround** – Use one of the other verification commands instead. See “Validating Studio Resources” in the *PolicyCenter Configuration Guide* for details.

#### [Duplicate contact when merging contacts through API \(PC-7421\)](#)

**Issue** – When you merge two instances of a contact through the API, you may get duplicate contacts for the following contact roles:

- PolicySecNamedInsured
- PolicyPriNamedInsured
- PolicyBillingContact

**Status** – Guidewire is aware of this issue.

#### [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.

#### [Need a summary of underwriting issues across all policies in a job group \(PC-9102\)](#)

**Issue** – The JobGroupRiskAnalysisLV needs to show a summary of underwriting issues across all policies in the job group. This was previously logged as CC-54177.

**Status** – Guidewire is aware of this issue.

#### [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. 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 original answer is still saved along with the other answers.

**Workaround** – The answers to these *irrelevant* questions 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)
        }
    }
}
```

### [Reinsurance Management: Facultative agreement added in later slice is removed \(PC-11587\)](#)

**Issue** – After handling preemptions and applying changes, PolicyCenter removes a facultative agreement added in later slice. For example, this occurs in the following situation:

1. Issue a commercial property submission effective today.
2. Create a policy change effective two months in the future. This policy change adds a facultative agreement to a location. Quote, but do not bind, the policy change.
3. Create a policy change one month in the future. Quote and bind the policy change.
4. In the policy change that takes effect two months in the future, handle preemptions, apply changes, and quote.

**Result** – The facultative agreement you added in that policy change is no longer on the policy.

**Status** – Guidewire is aware of this issue.

### [Null pointer when opening submission in draft status from account screen \(PC-11739\)](#)

**Issue** – In some circumstances, PolicyCenter throws an error when you navigate into and out of the submission wizard before clicking **Next**.

**Status** – Guidewire is aware of this issue.

### [Entity lost after handling preemptions in an out-of-sequence policy change \(PC-12178\)](#)

**Issue** – In any policy line that uses matchers to merge duplicate additions to the policy, a duplicate entity can be lost after handling preemptions. Using personal auto as an example:

1. Issue a personal auto policy that starts on the current date.
2. In a mid-term policy change starting four months in the future, add a new car. Remember the VIN. Quote and issue the policy change.
3. In an out-of-sequence policy change starting two months in the future, add that same car using the same VIN.
4. Quote, but do not bind the policy change.
5. Issue a third policy change that preempts the policy change in step 3.
6. Handle the preemption on the policy change in step 3.

**Result** – The car disappears from the policy.

**Status** – 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 screen of the agent
- 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.

## Diffs not reporting some deletions properly across renewals (PC-12328, PL-16076)

**Issue** – In some situations, the diffs may not properly reflect the changes. This can occur in a renewal with multi-version quoting, when the user makes a deletion in version 2 or greater. The underlying policy data is correct, but the diff does not show the deletions.

**Status** – Guidewire is aware of this issue.

## Problem 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 problem can occur with fields that are not base data.

**Workaround** – Add code that copies this type of data to the other versions.

## Entities newly added in future change do not display out-of-sequence conflicts (PC-13257)

**Issue** – Users are not notified of out-of-sequence conflicts when entities are added in an out-of-sequence change that were already added in a future change, and certain fields in those entities have changed.

For example, in Commercial Auto, MedPay was added in the as of 12/31 with a limit of \$15,000 and then in a later change, effective 10/31, is added again with a limit of \$5,000. The user is not warned that there is an out-of-sequence conflict between those two values.

**Status** – Guidewire is aware of this issue.

## User applied changes to contact are lost at quote (PC-13667)

**Issue** – User applied changes to a contact are lost at quote when that contact is re-added in an out-of-sequence job. Policy Contact roles are not copying fields correctly in the case of a policy contact role being added in a future change and then again in an out-of-sequence change, so changes made by the user are lost. For example:

1. Create and issue a personal auto policy.
2. Create and issue a policy change effective 05/01/2011 and add a completely new contact Bob as driver.
3. Create a second policy change effective 04/01/2011. Add Bob as driver from the existing contacts list, and change his name to Robert.

#### 4. Quote the policy change.

**Result** – Check the driver's name. The name has been reset to Bob.

When the *new* Bob is matched in the future, the fields are not correctly copied from the new, out-of-sequence slice. The `effDatedCopiers` have not been properly implemented for `PolicyContactRoles`.

**Workaround** – Modify the existing `AbstractPolicyContactRoleCopier` to copy basic fields (first name, last name, company name, DOB, etc.). Additionally, you can implement role specific copiers to copy role specific fields. For `PolicyDriver`, it should copy Driver's License Number and Driver's License State.

#### Side-by-Side quoting performance issues (PC-13920)

**Issue** – Guidewire is aware that the Side-by-Side screen rendering time may be unacceptably long in some situations.

**Status** – Guidewire is aware of this issue.

#### Parts of filtered diff code should be read only (PC-13968)

**Issue** – Costs and Transactions need to be filtered in diff items for purposes in addition to generating diff pages. For example, costs and transactions need to be filtered when changing the effective date of a policy change. This filtering is done in `DiffHelper.gs`. Although you can modify this class, your changes can break PolicyCenter functionality.

**Workaround** – Do not remove the filtering of Costs and Transactions from `diffItems` in `DiffHelper.gs`.

#### Reinsurance plugin dependent upon Reinsurance Management (PC-14016)

**Issue** – The `IReinsurancePlugin` interface is dependent upon Guidewire Reinsurance Management. Therefore, it is not recommended that you alter this plugin to integrate with another reinsurance system.

**Workaround** – Wait for Guidewire to provide a revised plugin interface.

#### Usage of state jurisdiction mapping (PC-14019)

**Issue** – There are some usages of the `StateJurisdictionMappingUtil.getJurisdictionMappingForState` method where it would be preferable to use methods on `JurisdictionMappingUtil`.

**Workaround** – If you are implementing PolicyCenter outside of the United States, replaces usages of `StateJurisdictionMappingUtil.getJurisdictionMappingForState` with methods on `JurisdictionMappingUtil`. You can override the methods on `JurisdictionMappingUtil` to determine the appropriate jurisdiction based upon the policy data.

#### ContactAPI.updateContact does not throw BadIdentifierException when updating an ABUID that does not exist in PolicyCenter (PC-14023)

**Issue** – The `ContactAPI.updateContact` method throws a `BadIdentifierException` when a contact management system requests update to an ABUID that does not exist in PolicyCenter. However, the exception does not indicate the specific issue.

**Workaround** – Change `ContactAPI` so that it throws a specific exception.

#### Error in ClaimCenter when searching for personal auto policies (PC-14045)

**Issue** – If you have enabled the integration with ClaimCenter, ClaimCenter throws an error in when searching for a personal auto policy created in PolicyCenter. This error only occurs on personal auto policies which have rental coverage.

**Workaround** – Make the following change in Studio.

1. Edit the Gosu class:

```
gw.webservice.pc.pc700.ccintegration.lob.CCPAPolicyLineMapper
```

2. Go to the handleCovTermSpecialCases method.

3. Change:

```
ccCov.ExposureLimit = packageTerm.Value
```

To:

```
ccCov.setExposureLimit(packageTerm.Value)
```

4. Change:

```
ccCov.IncidentLimit = packageTerm.Value
```

To:

```
ccCov.setIncidentLimit(packageTerm.Value)
```

**Edits in the New Contact popup are not saved to PolicyCenter or the contact management system (PC-14096)**

**Issue** – There is an issue when using **Check for Duplicates** on the **New Contact** screen. The user selects a matching duplicate, and PolicyCenter retrieves it. If the user updates that contact, the updates are neither saved in PolicyCenter nor sent to the external contact management system.

**Note:** The **Check for Duplicates** button is only available if PolicyCenter is integrated with a contact management system that support checking for duplicates.

**Workaround** – Make the following configuration change to PolicyCenter. After the duplicate contact information is retrieved, make the PCF file for that screen read-only, so that no further updates can be made to the contact.

**Apply pending account data updates batch process and out-of-sequence policy changes (PC-14114)**

**Issue** – The **Apply Pending Account Data Updates** batch process incorrectly updates the account name if the name is changed in out-of-sequence policy changes. For example,

1. Create a new contact named Larry Smith.
2. Create a new account for Larry Smith.
3. Create a policy effective today.
4. Create a policy change effective ten days in the future. Change Larry's first name to Lawrence.
5. Create an out-of-sequence policy change effective five days in the future. Change Larry's last name to Smith-Jones.
6. Six days in the future, run the **Apply Pending Account Data Updates** batch process.

**Result** – The batch process updates the account contact's first and last name. However, six days in the future, only the last name change is effective.

**Workaround** – Make a change in the following files:

- gw.contact.AbstractPolicycontactRoleAccountSyncableImpl
- gw.lob.wc.contact.PolicyOwnerOfficerAccountSyncableImpl
- gw.policyaddress.PolicyAddressAccountSyncableImpl

In the **createPendingUpdate** method, change:

```
:PendingUpdateTime = PolicyPeriod.EditEffectiveDate,
```

To:

```
:PendingUpdateTime = _accountSyncable.EffectiveDate,
```

## Commercial Property: Cannot copy coverages from one building to other buildings (PC-14118)

**Issue** – In the commercial property line, the **Copy Coverages** button is not currently available for copying coverages from one building to other buildings.

**Workaround** – At this time, there is no workaround for this issue.

## Ant task incorrectly generates stub Java classes for WS-I (PC-16370)

**Issue** – If you run the `regen-soap-api` Ant task with the `gwpc` tool, it generates a directory `PolicyCenter/soap-api` with files relating to published SOAP APIs. It is intended behavior to generate WSDL for WS-I web services (the new style) and RPCE web services (the older style). However, there is a known issue where the Ant task generates Java class stub files for SOAP client use for both types of web services. The JAR files are supposed to contain only RPCE published web services, not WS-I web services.

**Workaround** – Do not use the Java SOAP client classes for WS-I web services that PolicyCenter generates in the `java-api` folder. They are unsupported.

If you need Java SOAP client classes, use the `wsimport` tool to generate these classes instead. For information about how to use the `wsimport` tool, see “Web Services” → “Publishing Web Services (WS-I)” in the *Integration Guide*.

## Platform Known Issues

### First time you click on the arrow of the typekey input, the drop-down menu will not open (PL-10134)

**Issue** – The drop-down menu does not open on the first click of the arrow on a typekey input. Instead, the help text opens.

**Workaround** – Turn off help text on focus by setting `InputHelpTextOnFocus` to `false` in the `config.xml` file. By doing that, the help text shows only if you mouse over the input and will not interfere with opening a drop down menu.

### Studio Rules do not use correct capitalization for root object's name (PL-10740)

**Issue** – Rule set root objects are not named with first letter lower-cased.

**Workaround** – Rules engine issues a warning if the correct case for objects is not being used.

### Countries configured in zone-config.xml still generate a warning during regen-dictionary even when zone data is loaded for all of these countries (PL-11947)

**Issue** – Countries configured in `zone-config.xml` still generate a warning during regen-dictionary even when zone data is loaded for all of these countries.

**Workaround** – Warning message is created in error and can safely be ignored.

### The user interface cannot handle starting multiple instances of a batch process (PL-12372)

**Issue** – The user interface cannot handle multiple instances of a batch process.

**Workaround** – If multiple instances of a batch process need to be executed, they must be started from the command line. Also, you should ensure that the `BatchProcess.isExclusive()` method returns `false` to allow multiple instances to be run simultaneously.

## Type system refresh after PCF page title change does not update corresponding menu label (PL-13057)

**Issue** – The type system refresh after the PCF page title change does not update corresponding menu label.

**Workaround** – After updating a page title, the server restart must be done to refresh menu labels and avoid null pointer exceptions due to stale reference.

## There is a length limitation on entity localization table names (PL-13360)

**Issue** – There is a length limitation on entity localization table names.

**Workaround** – Ensure that localization `tableName` property specified in the entity extension file is less than 16 characters. The error message generated if the localization table name exceeds the maximum length indicates that 18 characters are allowed, but that does not account for two additional characters added by the application.

## US-Locations.txt file with the US geodata from GreatData has special characters that cause validation problems with United States Postal Service (USPS) data (PL-13384)

**Issue** – The `US-Locations.txt` file contains information that does not conform to United States Postal Service (USPS) standards for bulk mailings.

**Workaround** – The provided `US-Locations.txt` file is intended only for use in geocoding to identify addresses for a location. You can process the `US-Locations.txt` file to conform to your particular address standards, and then import that version of the file instead.

## GX models that reference virtual fields and enhancements throw null pointers if null (PL-13560)

**Issue** – The GX models that reference virtual fields and enhancements throw null pointers when they are null.

**Workaround** – Ensure that null checks and error handling is included so that if referenced virtual fields or enhancements are null, then there will not be a null pointer exception.

## Sending email with file attachment with Unicode filename is not correctly handed over to the mail server (PL-13582)

**Issue** – Sending email with file attachment with Unicode filename is not sent to the mail server correctly.

**Workaround** – Use Latin characters for file names on attached files.

## The GX model generated XSD file cannot be parsed by JAXB (PL-13598)

**Issue** – The GX model generated XSD file cannot be parsed by JAXB.

**Workaround** – Add JAXB annotation elements to the XSD file to specify the necessary metadata (such as class names) for JAXB to generate the Java class files. Contact Guidewire Support for an example XSD annotated in this way.

## JavaToolkit.gs has incorrectly hard coded memory, which results in failed `regen-java-api` Ant task (PL-13663)

**Issue** – The `JavaToolkit.gs` has hard coded memory which can result in failed `regen-java-api` Ant tasks.

**Workaround** – Increase the size of the maximum heap setting on line 161 of `JavaToolkit.gs` in the Ant module. The default value is 512.

## Cannot make a field from a delegate a localized column (PL-13761)

**Issue** – You cannot make a field from a delegate a localized column.

**Workaround** – Move the column to be localized off the delegate and onto each of the implementing entities. Then, in order to make the column appear as though it exists on the delegate, define an enhancement property on the delegate, that *delegates* to the appropriate column, depending on the implementing entity.

### Studio test functionality not working correctly (PL-15153)

**Issue** – If attempt to run a test in the Tests folder in the Studio Resources tree, running the test gives the following exception:

```
Using Test Environment Delegate: com.guidewire.testharness.ConfigEnvTestEnvironmentDelegate
gw.internal.gosu.parser.RuntimeExceptionWithNoStackTrace: java.lang.ClassNotFoundException:
    qa.DothisTest
Caused by: java.lang.ClassNotFoundException: qa.DothisTest
at gw.internal.gosu.parser.TypeLoaderAccess.getIntrinsicTypeByFullName(TypeLoaderAccess.java:522)
at gw.internal.gosu.parser.TypeSystemImpl.getByFullName(TypeSystemImpl.java:139)
at gw.lang.reflect.TypeSystem.getByName(TypeSystem.java:116)
at com.guidewire.studio.junit.ui.RunTestCommand.buildTestSuite(RunTestCommand.java:109
...

```

**Workaround** – Run the following command before you attempt to run a test:

```
.../bin gwpc dev-deploy
```

### 'Fetch Updates' writes into the wrong directory (PL-15303)

**Issue** – Changes to the built-in WS-I web service collections have caused the following issue. When PolicyCenter Studio refreshes a web collection, Studio downloads WSDL and related files to the local disk. The issue is that Studio does not properly implement copy-on-write to the customer modifiable configuration directory. Instead, Studio downloads the files to the local disk in the original directory, not to the configuration directory. This causes a checksum error after restarting the server or Studio.

---

**IMPORTANT** This issue only affects .wsc files that Guidewire includes with the application. It does **not** affect customer-created web service collections because those .wsc files always exist in the configuration directory already.

---

**Workaround** – As a workaround, do the following:

1. Quit Studio and any associated PolicyCenter server.
2. Copy any web service collection files to the configuration module. For example, copy the following file:  
`.../modules/pc/gsrc/wsi/remote/gw/webservice/bc700.wsc`  
 to the following location:  
`.../modules/configuration/gsrc/wsi/remote/gw/webservice/...`
3. Delete any downloaded WSDL or XSD files and directories in the collection.
4. If you modified a .wsc file, then restore that .wsc file in the original directory with the version from your original Guidewire software distribution.
5. Restart Studio and navigate to **Web Services - WSI** in the Resources tree.
6. Open the web service collection, then click **Fetch All** to re-download the WSDL and any associated files into the correct directory location.

### DBstats does not properly generate index data on Oracle database (PL-16008)

**Issue** – The DBstats (*Database Statistics*) batch process does not properly generate index data for tables if using the Oracle database.

**Workaround** – Execute the following query to identify the problematic indexes and to generate update statistics commands for them:

```
select 'execute dbms_stats.gather_index_stats('''gwUser''' || index_name ||
'',',estimate_percent>null);'
from user_indexes where index_name not like 'XX000%' and index_name not like 'SYS%'
```

In the query:

- Parameter gwUser is the name of the Oracle user who owns the Guidewire application schema.
- Parameter XX is the two letter product code for the Guidewire application (either CC, BC, PC, or AB).

### Studio does not recognize file `suite-config.xml` (PL-16011)

**Issue** – Guidewire uses file `suite-config.xml` to override URLs for web services for integrations between Guidewire applications. For example, you can set an override in this file to the URL for all web service connections from PolicyCenter to ClaimCenter. This file exists in PolicyCenter 7.0.0. However, it is not visible in PolicyCenter 7.0.0 Studio.

**Workaround** – If you need to modify this file and simulate copy-on-write Studio behavior, do the following:

1. Make a copy of the following base configuration file:  
`PolicyCenter/modules/pl/config/suite/suite-config.xml`
2. Move your copy of the file into the following directory:  
`PolicyCenter/modules/configuration/config/suite/suite-config.xml`
3. Edit the file as necessary.

### Sree web service shows in Studio (PL-16173)

**Issue** – The Web Services editor in the PolicyCenter 7.0 Studio shows a sree web service. Do **not** use the web service under any circumstances. It does not actually exist.

---

**IMPORTANT** Do **not** attempt to use the sree web service, either to consume or publish a web service.

---

**Status** – Guidewire is aware of this issue.

### Problem with `regen-java-api` command and JAR files (PL-16351)

**Issue** – When you run the `PolicyCenter/bin/gwpc regen-java-api` command, PolicyCenter creates a `PolicyCenter/java-api/lib` directory and puts JAR files for Java development in this directory. There is a known issue in this release where some of the resulting JAR files are not copied to this directory.

**Workaround** – For missing library files, copy (or import into your Java development environment) the missing library files from:

`PolicyCenter/admin/lib`  
To:  
`PolicyCenter/java-api/lib`



# Guidewire PolicyCenter 7.0.1 Release Notes

## Release 7.0.1.75

**Document Revision:** 20-May-2014

Copyright © 2001-2014 Guidewire Software, Inc. All rights reserved.

Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Reinsurance Management, Guidewire ContactManager, Guidewire Vendor Data Management, Guidewire Client Data Management, Guidewire Rating Management, Guidewire InsuranceSuite, Guidewire ContactCenter, Guidewire Studio, Guidewire Product Designer, Guidewire Live, Guidewire DataHub, Guidewire InfoCenter, Guidewire Standard Reporting, Guidewire ExampleCenter, Guidewire Account Manager Portal, Guidewire Claim Portal, Guidewire Policyholder Portal, ClaimCenter, BillingCenter, PolicyCenter, InsuranceSuite, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks, service marks, or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries.

All other trademarks are the property of their respective owners.

**This material is confidential and proprietary to Guidewire and subject to the confidentiality terms in the applicable license agreement and/or separate nondisclosure agreement.**

Guidewire products are protected by one or more United States patents.

## Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Support

- Known Issues and Limitations

## Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

---

**IMPORTANT** If you missed 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 7.0.1.75.

- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 3.0.1 or later.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 6.0.0 or later.

### Changes to Application Server Requirements

In this release, the WebSphere and WebLogic application servers are supported. These application servers were not supported in PolicyCenter 7.0.0. See the *PolicyCenter Installation Guide* for the complete list of supported application servers.

## Installing This Release

Refer to the following:

- The *PolicyCenter Installation Guide* for general installation information
- The *Guidewire Contact Management Guide* to install ContactManager
- Prior PolicyCenter release notes for any versions that you have skipped

### Upgrade Issues

This section describes changes to the PolicyCenter base configuration that may cause upgrade issues.

#### Studio 'Find Usages' (ALT+F7) Returns Incorrect Usages (PL-15863)

Previously, performing a search in Studio (ALT+F7) for the usage of a method did not return the correct usages of that method. This fix corrects that problem. However, for the fix to work correctly, you must drop the Studio TypeInfo database before you start Studio for the first time after you apply the fix.

---

**IMPORTANT** You must drop the TypeInfo database, otherwise the problem persists.

---

To drop the Studio TypeInfo database, do the following:

1. Navigate to the following location in your Guidewire application install directory:  
...\\studio\\db
2. Delete the following files:
  - studio.gwdb
  - studio.h2.db

- `studio.lock.db`
  - `studio.trace.db`
3. Restart PolicyCenter Studio. PolicyCenter automatically regenerates these files upon Studio start-up.

## Support

For assistance, visit the Guidewire Resource Portal – <http://guidewire.custhelp.com>

## Changes in This Release

This section describes the product changes in this release.

- Rate Table Maintenance
- Base PCF File Changes
- Base Resources Changes
- Upgrade Diff Report

### Rate Table Maintenance

New in this version of PolicyCenter is rate table maintenance. Rate table maintenance provides the ability to rate policies internally or by integrating with an external rating engine. Rate table maintenance enhances PolicyCenter internal rating capability by providing a set of tools to manage and maintain rate book and rate tables in PolicyCenter.

Rate table maintenance is a part of Guidewire Rating Management.

---

**IMPORTANT** To determine whether your Guidewire PolicyCenter license agreement includes Guidewire Rating Management, contact your Guidewire sales representative. Rating Management requires an additional license key. For instructions on obtaining and installing this key, contact your Guidewire support representative.

---

### Base PCF File Changes

All links below require the `readme_files` directory on your local disk.

#### PolicyCenter release 7.0.0 to 7.0.1

- To view a report of the changes in the base PCF files in the `modules/pc` directory, [click here](#).
- To view a report of the changes in the base PCF files in the `modules/p1` directory, [click here](#).

### Base Resources Changes

All links below require the `readme_files` directory on your local disk.

#### PolicyCenter release 7.0.0 to 7.0.1

- To view a report of the changes in the base resources in the `modules/bc` directory, [click here](#).

## Upgrade Diff Report

Guidewire provides a report that describes changes in display keys, entities, typelists, and the Gosu API. To view this report, visit the Guidewire Resource Portal.

## 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 meant to be a comprehensive list.

- PolicyCenter Improvements and General Issues
- Platform Improvements and General Issues

### PolicyCenter Improvements and General Issues

ID	Description
<b>Account</b>	
PC-13027	<p>Items on the Account Relationship popup menu had the following order:</p> <ul style="list-style-type: none"> <li>• Parent of</li> <li>• Common Ownership</li> <li>• Child of</li> </ul> <p>This was changed to the following order:</p> <ul style="list-style-type: none"> <li>• Parent of</li> <li>• Child of</li> <li>• Common Ownership</li> </ul>
PC-13035	Until now, in related accounts it was possible to set the same account as its own parent or its own child. This change prevents an account from being related to itself.
PC-13073	The IndustryCode field on NamedInsured now defaults to the IndustryCode of the owning Account. For example, if you designate the AccountHolder of an Account to also be a NamedInsured, then the IndustryCode will be copied from the Account. Previously the IndustryCode field on NamedInsured would have been left empty.
PC-13871	Previously, in the Account → Submission Manager screen, the Total Cost included the costs of Not Taken, Withdrawn, and Declined submissions. Now, these types of submissions do not display a Total Cost and are not included in the cumulative Total Cost.
PC-14142	When a source account is merged into a target account, PolicyCenter creates a history event on the target account. The history event includes the account number of the source account that was merged.
PC-14172	Improved the performance of Account Search by Last Name.
PC-14179	The Postal Code is now a minimal search criteria for Account Search since it is generally no less restrictive than City/State, which was allowed as a minimal search criteria.
PC-14184	Fixed an issue in Account Search criteria. Blanks (white space) in Account Search criteria fields are no longer considered minimally acceptable criteria.
PC-14199	Fixed a performance issue in Account searches.
PC-14204	In PolicyCenter 7.0.0, the AccountSearchCriteria entity was deprecated and the AccountSearchCriteriaEnhancement that worked with it was removed. To preserve the deprecated functionality, AccountSearchCriteriaEnhancement has been restored. The non-deprecated replacement functionality is in a Gosu class named AccountSearchCriteria, which was previously named AccountSearchCriteriaGosu in PolicyCenter 7.0.0. Although they have the same name, the AccountSearchCriteria entity and AccountSearchCriteria Gosu class do not conflict and can not be used together.
<b>Administration</b>	
PC-14116	On the Team screen, the header for column Primary Insured has been changed to Account Holder in the Renewals and Other Work Orders tabs. This column displays the Account Holder details of the work orders, not the Primary Insured's details.

ID	Description
PC-14354	Previously on the Team tab, PolicyCenter did not sort the User column. A sort was added to the assign multiple jobs/policies screen (AssignUserRoleOwnerPopup.pcf). The sort is alphabetical by last name then first name for a person, and by name for a company.
<b>Contacts</b>	
PC-7271	<p>Fixed an issue where the user could delete an address in use as a PolicyAddress. This occurred in the following sequence:</p> <ol style="list-style-type: none"> <li>1. Have at least one contact on an account other than the account holder. That contact should have at least two addresses.</li> <li>2. Start a submission.</li> <li>3. On the Policy Info page, change the Primary Named Insured to the contact in step 1. The policy address is changed to their primary address.</li> <li>4. Click the link for that contact to open Primary Named Insured page.</li> <li>5. Click the Addresses tab.</li> <li>6. Set the primary address to the other address for the contact. Then remove what was the primary address (and is currently the policy address).</li> </ol> <p>Expected result: User should not be allowed to remove that address because it is in use as a policy address.</p> <p>This issue was resolved by eliminating the user's ability to delete the contact's address in the user interface when the contact is edited from a popup triggered within the context of a policy. The check box is no longer available next to the address entry and a description on the form explains why. When a contact's addresses are edited outside the context of a policy, the user is presented with a validation error when attempting to delete an address that is a primary policy address.</p>
PC-7441	<p>A new implementation of the <code>Mergeable</code> interface has been created for singleton policy contact roles. For example, a Primary Named Insured should only ever exist once on a policy. Given this, there should be no need to merge two primary named insured contacts. This implementation, <code>SingletonPolicyContactRoleMergeableImpl</code>, simply throws an <code>IllegalStateException</code>.</p> <p>This applies to:</p> <ul style="list-style-type: none"> <li>• <code>PrimaryNamedInsured</code></li> <li>• <code>PolicyBillingContact</code></li> <li>• <code>SecondaryNamedInsured</code></li> </ul>
PC-11030	Switched the order of the Associated Work Orders and Addresses tabs on the Account File Contacts screen. The Addresses tab now precedes Associated Work Orders.
PC-13067	The behavior when contact data is synchronized between a open job and the account has changed. This means that in many cases, temporary changes made to a job but not bound are no longer copied to the account.
PC-13667	<p>Previously, user applied changes to a contact were lost at quote when that contact was re-added in an out-of-sequence job. Policy Contact roles were not copying fields correctly in the case of a policy contact role being added in a future change and then again in an out-of-sequence change, so changes made by the user were lost.</p> <p>This was previously a PolicyCenter 7.0.0 Known Issue.</p>
PC-13832	<p>Fixed an issue where changes to a policy contact was not copied to siblings that existed only in future slices. This occurred in the following circumstance:</p> <ol style="list-style-type: none"> <li>1. Create a personal auto policy effective today. Add a secondary named insured to the policy. Bind and issue.</li> <li>2. Start a policy change effective one month in the future. Add the secondary named insured as a driver to the policy. Issue.</li> <li>3. Start a second policy change effective 10 days in the future. Change the name of the secondary named insured. Issue.</li> <li>4. Look at the policy as of the date of the first policy change.</li> </ol> <p>Previous results: If you looked at the contact from the secondary named insured link, it had the new name (correct). If you look at the contact from the drivers link, it had the old name (incorrect).</p>
PC-14295	The Gosu class, <code>ProductCodeFilterSet</code> , which provides filtering functionality for restricting query results by product code, is now available in Studio.

ID	Description
PC-14411	<p>Fixed an issue with the <b>Withdraw</b> button. Previously, clicking <b>Withdraw</b> called <b>SaveDraft</b>. This resulted in a synchronization of policy contact information even though <b>Save</b> was not clicked.</p> <p>The steps to reproduce this problem are:</p> <ol style="list-style-type: none"> <li>1. Create a new Submission.</li> <li>2. Change the Primary Named Insured's first name.</li> <li>3. Without clicking Next, Quote, or Save Draft, click <b>Withdraw Work Order</b>.</li> <li>4. Click on the account and view that the name change has been updated on the account.</li> </ol> <p><b>SaveDraft</b> is no longer called when clicking <b>Withdraw</b>. As a result, the policy contact is no longer updated after you click <b>Withdraw</b>. Also, information on the current screen is no longer persisted.</p>
PC-14477	<p>Pending updates are populated only with fields that change, not a snapshot of all fields. For example, if a policy change is effective on a future date and only the first name is changed, the pending update contains only the first name. The pending update does not contain the first, last, marital status, or other fields.</p> <p>When PolicyCenter applies a pending update, if a field is null but <code>IsNull</code> is <code>false</code>, then PolicyCenter does not update the field. This behavior is consistent with PC 7.0.0.</p> <p>The code for creating pending updates and detecting how they should be populated was moved from an enhancement of pending updates to the <code>gw.api.domain.account.AbstractDateAwareAccountSyncedFieldImpl</code> API.</p>
PC-14518	Fixed an issue with synchronization between TaxId and OfficialIds.
<b>Financials</b>	
PC-12085	In <code>gw.rating.CostData</code> , added more logging to help determine why costs are or are not merged. Also changed logging level from INFO to DEBUG, because these log messages are for debugging and should not be displayed in a production system.
PC-14048	<p>A <code>PaymentSystemURL</code> parameter was added to <code>config.xml</code>. This affects the <b>Add</b> button on the <b>Payments</b> screen. If the value is <code>pc</code>, the <b>Add</b> button launches the demonstration payment system.</p> <p>The <code>BillingInvoiceStreamInputSet</code> PCF file contains the configuration parameter for the payment system URL and the PolicyCenter URL. These are set to the same value, but should be changed by removing:</p> <pre>var returnUrl = gw.api.system.PCConfigParameters.PaymentSystemURL.Value</pre> <p>and remove the comment from:</p> <pre>// gw.plugin.Plugins.get(gw.plugin.webconfig.IPolicyCenterWebConfigPlugin).PolicyCenterURL</pre>
PC-14151	Previously, when returning from the demonstration payment system, PolicyCenter displayed the default step ( <b>Quote</b> screen) in the policy. Now PolicyCenter returns to the <b>Payment</b> screen.
PC-14472	<p>The creation of cost amounts for rating, and the proration of those amounts when adjusting the rate for a partial period, were done using different formulas. For consistency, they now use the same formula.</p> <p>During merge, you may notice a change to methods on the Gosu class <code>CostData</code> and its subclasses.</p>
<b>Forms</b>	
PC-11472	<p>The same two typecodes, quote and bind, were defined in both <code>FormInferenceTime.tti</code> and <code>FormInferenceTime.ttx</code>.</p> <p>Removed the redundant <code>FormInferenceTime.ttx</code>.</p>
PC-14032	Fixed an issue where all jurisdictional forms were being updated mid-term, rather than just the one for the affected jurisdiction.
PC-14070	<p>Added <b>Form Name</b> and <b>Group Code</b> search fields to the <b>Policy Form Patterns</b> search screen.</p> <p>To search by <b>Form Number</b>, <b>Form Name</b>, or <b>Group Code</b> only, you must enter at least three characters in the text box. If you specify two or more fields, you can enter fewer than three characters in these fields.</p>
PC-14072	In <b>Administration</b> → <b>Policy Form Patterns</b> , there is a new menu item on the <b>Form Pattern</b> → <b>Jobs</b> tab. When you select <b>Add</b> , there is now an <b>All Issuance and Change Jobs</b> menu item. When you choose <b>All Issuance and Change Jobs</b> , PolicyCenter adds the Submission, Renewal, Rewrite, Policy Change, and Rewrite New Account jobs to the form pattern. If these five jobs are already on the form pattern, then the <b>All Issuance and Change Jobs</b> option is not visible. If some of the five jobs are on the form pattern and you select <b>All Issuance and Change Jobs</b> , then PolicyCenter adds the remaining jobs.

ID	Description
PC-14074	<p>Fixed an issue in Administration → Policy Form Patterns on the Form Pattern screen. Previously, if you selected a row in the Jurisdictions listview, then clicked Duplicate, PolicyCenter added a new row. However the previously selected row was unselected.</p> <p>The Duplicate button is now a drop-down list that lets you create one to 10 duplicates. The check box remains selected after duplicating.</p>
PC-14076	<p>Fixed an issue on the Form Pattern → Inference tab.</p> <p>Previously, if a monoline product was selected, the Policy line defaulted to &lt;none selected&gt;. Now, the policyline for the monoline product is selected by default.</p>
PC-14091	<p>Relaxed form pattern validation so it does not require all form patterns in the same group to have matching policy lines, products, and jobs. A form pattern's group is determined by its Group Code; if no Group Code is specified, its Form Number is used as its Group Code.</p>
PC-14104	<p>Removed the Value and Description fields from the FormPatternCovTermValue entity since the Value property is not used and the Description property can be retrieved from the covterm pattern. The Description property is now an enhancement on the FormPatternCovTerm entity. An upgrade is required to drop the Value and Description columns.</p>
PC-14171	<p>Previously, there was validation that would prevent creation of form patterns with the same groupcode and clause pattern. This validation was only at form pattern creation time in Administration → Policy Form Patterns, but not carried out while importing form patterns. Now, this unnecessary validation (form patterns with same groupcode must have same clause pattern) has been removed. The validation has been removed from FormPatternValidation and getLookupDates methods in the inference classes GenericCovTermSelectionForm and GenericClauseSelectionForm.</p>
PC-14415	<p>Removed the Name and Description fields from the FormPatternCovTerm entity as these values can be retrieved from the coverage term pattern. The Name and Description properties are now enhancements on the FormPatternCovTerm entity. An upgrade is required to drop the Name and Description columns.</p>
<b>Infrastructure</b>	
PC-10832	<p>PolicyCenter only supports the import of zone data through staging tables. However, prior to PolicyCenter 7.0, most PolicyCenter tables were defined in the metadata as loadable. Therefore, PolicyCenter would create a staging table for each table. A database upgrade trigger drops the staging tables and drops the LoadCommandID column from each table defined as loadable in prior versions, with the exception of zone data tables.</p>
<b>Integration</b>	
PC-13662	Fixed an issue where policy search from SOAP ignored the taxID field in CCPCTSearchCriteria.gs.
PC-14045	<p>Previously, if you had enabled the integration with ClaimCenter, ClaimCenter threw an error in when searching for a personal auto policy created in PolicyCenter. This error only occurred on personal auto policies with rental coverage.</p> <p>This was previously a PolicyCenter 7.0.0 Known Issue.</p>
PC-14088	PolicyCenter is not yet multi-currency-aware. To improve integration with external systems (such as ClaimCenter) that are multi-currency aware, DefaultApplicationCurrency is now set in config.xml.
PC-14112	Updated the Javadoc for startNewRenewal in PolicyRenewalAPI to reflect how the code currently works.
PC-14194	The isCCPolicyLocationUnused method in CCPolicyGenerator.gs was only checking for actual type of property risk units (CCLocationBasedRU and CCVehicleRU). The method now correctly checks for CCLocationBasedRU (and any of its subtypes) and CCVehicleRU (and any of its subtypes).
PC-14233	Fixed an issue in the integration with ClaimCenter. Previously, PolicyCenter sent a coverage with no deductible as having a 0 deductible. Now, PolicyCenter sends a coverage with null deductible.
PC-14323	The integration with BillingCenter now creates an alternate billing account in BillingCenter if it does not already exist.
PC-14466	<p>Previously, account level Billing Settings were displayed on Account → Billing tab, and the user could edit Billing Settings.</p> <p>However, in the default configuration PolicyCenter creates the payment instrument at the invoice stream. In this scenario the default billing settings at the account level are not used.</p>
<b>Job Infrastructure</b>	

<b>ID</b>	<b>Description</b>
PC-12970	<p>Fixed an issue with <b>Actions → Copy Submission</b>. This action called the copySubmission method on the <b>Code</b> tab of the PCF file. Previously, if copySubmission failed, a duplicate instance of the submission remained in the database, and PolicyCenter threw an exception.</p> <p>For example, if a user selected <b>Actions → Copy Submission</b> but did not have the <b>CopySubmission</b> permission, PolicyCenter threw an exception. This fixes the issue in the user interface. The fix includes changes to PCF files to filter out jobs with status <b>Temporary</b> and adds a check for the <b>CopySubmission</b> permission.</p> <p>This issue is fixed in the database by filtering out the temporary periods when retrieving a policy.</p>
PC-13257	<p>Previously, the user was not notified of out-of-sequence conflicts when entities were added in an out-of-sequence change that were already added in a future change, and certain fields in those entities had changed.</p> <p>For example, previously in Commercial Auto, MedPay was added as of 12/31 with a limit of \$15,000, and then in a later change effective 10/31, was added again with a limit of \$5,000. Previously, the user was not warned that there was an out-of-sequence conflict between those two values.</p> <p>This was previously a PolicyCenter 7.0.0 Known Issue.</p>
PC-13843	<p>The following labels were changed:</p> <ul style="list-style-type: none"> <li>• <b>Actions → Spin Policy</b> becomes <b>Spin-off Policy from this One</b></li> <li>• <b>Actions → Split Policy</b> becomes <b>Split Policy into Two</b></li> </ul> <p>On the <b>Policy Summary</b> screen, the <b>Split/Spin Policies</b> label becomes <b>Split Off Policies</b>.</p>
PC-13844	Null strings now display as blank in out-of-sequence conflict screens.
PC-13971	Fixed an issue with two side-by-side versions where withdrawing #1 Selected did not result in #2 becoming the active version.
PC-14039	Fixed an issue where an Activity was not editable from the <b>Team → Activities</b> screen.
<b>Jobs</b>	
PC-8245	Previously, users could not quick-jump to a policy with a policy change that was still in progress (that is, not yet completed). Now, users can quick-jump to a policy with a policy change that is in progress.
PC-8867	Fixed an issue where PolicyCenter threw an <b>EvaluationException</b> when a producer and underwriter attempted to access a business auto policy file at the same time.
PC-11617	The <b>CancellationAPI</b> now requires that the reason code and the source match.
PC-11727	Previously, a producer could not make a submission <b>Not Taken</b> that had activities associated with it that were created by another user and assigned to another user. An example of this is an activity created by the underwriter and assigned to himself. Previously PolicyCenter threw an error because the producer did not have permission to withdraw activities if he was not the creator or assigned to that activity.
PC-11739	<p>Previously, PolicyCenter threw a null pointer exception when opening a submission in draft status from the <b>Account</b> screen.</p> <p>This was previously a PolicyCenter 7.0.0 Known Issue.</p>
PC-11830	In Side-by-Side or Multi-Version quoting, the <b>Policy Versions</b> screen the label on the first column has been changed from <b>Status</b> to <b>Version Name</b> . This change is in the <b>ManageBranchesScreen.pcf</b> file.
PC-11891	When a cancellation is done on a policy with an existing future renewal, the renewal will also be canceled. This open cancellation for the future renewal no longer prohibits the reinstatement of the originally canceled policy.
PC-12766	<p>Fixed an issue where the <b>Workflow</b> batch process did not rescind multiple same-day cancellations. For example:</p> <ol style="list-style-type: none"> <li>1. Create 2 future cancellations scheduled on the same day.</li> <li>2. Move the PolicyCenter clock forward to the scheduled day.</li> <li>3. Run the <b>Workflow</b> batch process.</li> </ol> <p>The <b>Workflow</b> processed the first cancellation but did not rescind the second cancellation if it was on the same day. The second cancellation stayed in pending status with preemptions needing to be resolved.</p> <p>Now, same day cancellations are now properly rescinded, and a cancellation will no longer be issued if any prior job in the workflow rescinded it.</p>
PC-12773	Improved the error message displayed when an attempt is made to set the expiration date of a canceled policy to a date earlier than its cancellation date.

ID	Description
PC-13173	For consistency, the job wizards were updated to use the variable OpenforEdit instead of PolicyPeriod.OpenforEdit.
PC-13588	Previously, PolicyCenter calculated the cancellation effective date incorrectly. For example, if you set the LeadTime in the NotificationConfig system table to 30, PolicyCenter would provide 29 days of notification and cancel the policy on the 30th day. Now PolicyCenter calculates this value correctly, and provides 30 days of lead time, and cancels on the 31st day.
PC-13881	If the user initiated a policy change with an effective date later than the original policy effective date, and then used the Work Orders tool to select the two work orders and compare them, the comparison would display the original data incorrectly under policy change and vice-versa. This has been fixed so that data are shown appropriately under the work order they belong to.
PC-13920	Improved the performance of rendering the Side-by-Side Quoting screen.  This was previously a PolicyCenter 7.0.0 Known Issue.
PC-13954	When data is copied from a policy to a work order, the history item description now includes policy number when the source job is associated with a policy.
PC-13959	In the sample data, the Document Create permission was added for audit users.
PC-13972	In the sample data, the Advance Audit permission was added to the Audit Supervisor and Premium Auditor roles.
PC-14103	Fixed an issue with the gw.policy.PolicyPeriodBaseEnhancement.PreemptedJobsIfBoundNow property. Previously this property returned all open jobs on the policy term changed by the newly bound job. It now only returns jobs that have policy periods that are based on a different policy period than the policy period just created by the job.
PC-14114	Previously, the Apply Pending Account Data Updates batch process incorrectly updates the account name if the name is changed in out-of-sequence policy changes.  This was previously a PolicyCenter 7.0.0 Known Issue.
PC-14128	Fixed an issue that occurred after issuing a policy change. The link to a future rewritten renewal could go to a policy period with a flat cancellation. Now the link goes to the subsequently rewritten period, if any.
PC-14141	Fixed an issue that caused an exception after clicking on the Source Account link after merging a source account into a target account.  The Source Account link in the Policy File Summary page is now inactive when that source account is merged with another account.
PC-14145	Users without permission to edit audit jobs are now prevented from handling preemptions on audit jobs. A warning message explains why the handle preemptions button is unavailable in such cases. When an audit job is preempted, PolicyCenter now generates an activity for the auditor.
PC-14224	Fixed an issue where two policy periods were linked to the same Job. This caused the policy to become unusable and required deleting the offending data.
PC-14257	Previously, when a cancellation was rescinded as the result of another cancellation being completed, PolicyCenter still attempted to complete that cancellation on the scheduled cancellation date.  Now, cancellations are properly rescinded and a cancellation will no longer be issued if any prior job in the workflow rescinded it.
PC-14311	This fix applies to general liability and workers' compensation policies. If after waiving a scheduled expiration Final Audit, there was a policy change or cancellation on the policy, PolicyCenter scheduled the final audit again. Now, PolicyCenter does not schedule the final audit again.
PC-14343	In the default configuration, permissions were changed for consistency.  The advanceaudit permission was removed from the following roles: producer, underwriter, and underwriter_supervisor.  The following permissions were added to the underwrite_asst role: advancecancellation, advancesubmission, advanceissuance, editissuance, advancepolchange, editpolchange, advancereinstat, advancerenewal, editrenewal, advancerewrite, advancerewrnewacct.
PC-14501	Fixed an issue that caused a stack dump when changing the primary named insured in an out-of-sequence policy change.
<b>LOB - BA</b>	
PC-14166	The BANonOwnedLiabCovCost was deprecated, so the BANonOwnedLiabCovCostData class is also deprecated.

ID	Description
<b>LOB - BA, PA</b>	
PC-14499	Previously, when you attempted to add the vehicle twice, GarageLocation was ignored. Now GarageLocation is included, but in some cases the correct garage location cannot be set. In these cases the GarageLocation will be set to null, in order to guarantee that the user sets the correct location.
<b>LOB - CP</b>	
PC-14118	Previously in the commercial property line, the Copy Coverages button was not available for copying coverages from one building to other buildings.  This was previously a PolicyCenter 7.0.0 Known Issue.
<b>LOB - CPP</b>	
PC-14012	Fixed a NullPointerException when adding an Inland Marine line in a policy change and an out-of-sequence policy change.
PC-14057	Fixed a null pointer exception on quote that occurred if you removed the general liability line from a commercial package policy in a Policy Change, then added general liability Additional Insureds in an out-of-sequence change.
<b>LOB - IM</b>	
PC-14326	Added a new validation for signs to make sure that duplicate signs are not added. Duplicate signs are determined by location, sign type, description, and interior values.
PC-14332	Previously, inland marine coverage parts (signs, contractors equipment, and account receivables) were not auto-numbered correctly in out-of-sequence and preemption. Changes include: <ul style="list-style-type: none"> <li>• In out-of-sequence, renumber the auto-number sequence if duplicates were merged.</li> <li>• In preemption, renumber when applying changes to a future job.</li> </ul>
PC-14397	Fixed an issue in inland marine where viewing the policy as of a particular date did not show all the account receivables. Made a change to IMAccountReceivableMatcher making AccountsReceivables unmatchable as it was in PolicyCenter 4.0.
<b>LOB - PA</b>	
PC-12328	Previously, diffs did not report some deletions across renewals.  This was previously a PolicyCenter 7.0.0 Known Issue.
PC-13847	In Personal Auto → Vehicles → Vehicle Details, you could not select a vehicle by clicking on its row, except by first clicking on the little triangle on the far left of the row. You can now select a vehicle by clicking on its row.
PC-14041	The MotorVehicleRecord entity has been removed. In PolicyCenter 7.0.0, the motor vehicle records in personal auto was introduced and uses different entities.
PC-14042	Made a cosmetic change to the Personal Auto Quote → CostChange detail page. Previously the detail was stretched across the whole page. Now it is contained inside the detail view.
PC-14115	Fixed an issue where PolicyCenter threw an exception after clicking the All ordered MVRs received - Clear activity link in the Team Screen. This happened if the corresponding policy was not issued or bound.
PC-14150	Fixed an issue in Quick Quote for personal auto where PolicyCenter threw an error after entering the driver and vehicle information, and clicking Next. Made changes to PALiabilityCovEnhancement, that keeps a map of PALiability package values per state which are used to initialize PALiabilityCov.
PC-14255	In Studio, the Evaluation → PolicyEvalNew → PersonalAuto → StandardJobs rule was modified to use a statement rather than a method body. Method bodies are not officially supported. Therefore this rule reported an error when viewed in Studio. The functionality of the rule is unchanged. This is a cosmetic difference only.
<b>LOB - WC</b>	
PC-12134	Codes and display keys labels were changed to refer to Monopolistic states, not monopoly states.

ID	Description
PC-14437	<p>Fixed an exception on the Premium → Premium Details tab when handling a preempted audit and calculating premiums on it. This exception was eliminated by making the premium amount calculation return 0 rather than null.</p> <p>This exception occurred with the following steps:</p> <ol style="list-style-type: none"> <li>1. Login as aarmstrong, create a worker's compensation policy (bound). Copy the policy number.</li> <li>2. Login as svisor and go to PolicyFile.</li> <li>3. Click on AuditSchedule.</li> <li>4. Edit the final audit.</li> <li>5. Change the process date of audit as today.</li> <li>6. Run the audit batch process.</li> <li>7. Go to PolicyFile, you will see an activity for Adam Auditor as <b>A new audit has been assigned</b>.</li> <li>8. Start a cancellation (reason-fraud, canc.eff.date - default is shown in future, cancel now). You will see cancellation preempts audit.</li> <li>9. Login as aauditor and handle the preempted audit by clicking the activity.</li> <li>10. Click Handle Preemption.</li> <li>11. Enter mandatory fields on the Summary and Details pages.</li> <li>12. On the Details page, click Calculate Premiums and click the Premium Details tab.</li> </ol>
<b>Localization</b>	
PC-14017	
PC-14017	In Studio, Boolean fields in native panels and popup dialogs can now be localized. This does not cover all instances, however, such as localizing Boolean fields in PCF editor panels.
PC-14019	<p>Previously, there were some usages of the <code>StateJurisdictionMappingUtil.getJurisdictionMappingForState</code> method where it would have been preferable to use methods on <code>JurisdictionMappingUtil</code>.</p> <p>This was previously a PolicyCenter 7.0.0 Known Issue.</p>
PC-14062	Fixed an issue where GLRatingEngine was referencing State instead of Jurisdiction.
<b>Locations</b>	
PC-7511	Updated the address auto-fill methods so that the City field is populated with auto-fill using zip code. If there are multiple cities with the same zip code, then the city will not be auto-filled.
PC-8971	Address auto-fill added some new methods and deprecated others. In PolicyCenter, address auto-fill helper functions were updated.
PC-13306	Fixed an issue with address auto fill.
<b>Miscellaneous</b>	
PC-7536	Fixed an issue with Auto Save and notes. Previously, you were able to see a New Note from the Unsaved Work drop-down list, but unable to link to it.
PC-7591	Fixed an issue with Auto Save and activities. Previously, you were able to see a New Activity from the Unsaved Work drop-down list, but unable to link to it.
PC-11251	Fixed issue where links to the work order are available even if the user does not have permission to view the work order.
PC-13569	<p>Two changes were made to the activities that are created for back-dated contact changes:</p> <ol style="list-style-type: none"> <li>1. The activity is now assigned to the user who made the contact change. Previously the activity was left unassigned.</li> <li>2. Only one activity is created for any individual <code>PolicyContactRole</code> change. Previously multiple, duplicate activities were created in out-of-sequence jobs.</li> </ol>
PC-13896	A notes array is now available on <code>PolicyPeriod</code> and <code>Job</code> .
PC-13968	<p>Previously, parts of filtered diff code should have been marked as read only. For example, Costs and Transactions need to be filtered in diff items for purposes in addition to generating diff pages. For example, costs and transactions need to be filtered when changing the effective date of a policy change. This filtering is done in <code>DiffHelper.gs</code>. If you modified this class, your changes could have broken PolicyCenter functionality. Now these classes are marked as read-only.</p> <p>This was previously a PolicyCenter 7.0.0 Known Issue.</p>
PC-14013	Previously, the Administration → Workflows screens were unable to search for a workflow that was not based on the <code>PolicyPeriodWorkflow</code> supertype. This is now possible. In addition, the workflow screen have been updated to be more consistent with other Guidewire applications.

ID	Description
PC-14020	The <code>gw.api.Copier</code> class is now available in Studio.
PC-14021	Removed unnecessary workflow file warning messages on server start-up.
PC-14049	Fixed a performance issue with the <b>Release Lock</b> toolbar button and the <code>showConfirmMessage</code> property.
PC-14092	Added a condition to <code>JobGroupForward.pcf</code> to display an appropriate error message if the user does not have permissions to continue.
PC-14307	A number of files in <code>gw/plugin/messaging</code> folder are now available in Studio.
<b>Policy File</b>	
PC-10563	Previously, the participants roles were not copied down from a job to policy. For example, in a Policy Change job, if an underwriter role was changed, the role was not reflected at policy level. This behavior was for all jobs except submission. Now, the participants are copied from all jobs to the policy and vice versa (with exception for creator role and Audit job).
<b>Product Model</b>	
PC-7116	In Studio, there is now an Existence tab for Coverage, Condition, and Exclusion clauses. On this tab, you can specify whether the existence for the selected clause is Electable, Required, or Suggested. You can also enter Gosu code in an Existence Script. You can use the existence script to conditionally make a coverage required, suggested, or electable based upon the criteria you define. For example, assume that uninsured motorist coverage is required in Wisconsin but electable in Pennsylvania. You can add a script to check the state, and set existence for the coverage based upon the state.
PC-12107	Resolved an issue with updating the questions displayed from a question set when the availability of the questions has changed.
PC-12200	Previously, if an <code>outputConversion</code> function was declared for a <code>CovTermDirectInput</code> , it was not executed. This issue has been resolved so that an <code>outputConversion</code> , if one is declared for a <code>CovTermDirectInput</code> , is recognized and executed.
PC-12392	In Studio, in the <b>Coverages</b> tab of a product line, <b>Coverage Term</b> , <b>Coverage Term Opt</b> , and <b>Coverage Term Pack</b> all now display in format: name [code].
PC-13897	Previously when starting Studio, the <code>copy-starter-resources</code> target ran only if there was no <code>config</code> directory present in the configuration module. Among other things, <code>copy-starter-resources</code> is responsible for copying the <code>productmodel</code> into the configuration module if one is not already present. Previously, if the <code>config</code> directory was present in configuration module but there was no <code>productmodel</code> , the <code>productmodel</code> was not copied. Now Studio checks to determine whether or not a specific file or directory should be copied. In this case, <code>copy-starter-resources</code> now checks specifically for existence of <code>productmodel</code> directory. This resolves an issue where some users had experienced Studio editing product model files in the <code>pc</code> module instead of the configuration module.
PC-13953	Changed Studio lookup verification to raise a warning rather than an error when there are no lookups.

ID	Description
PC-14047	<p>Previously, when a coverage was added, it was not possible to configure the diff tree such that the coverage terms are displayed. Some of the methods in DiffTree were made protected. You can override these methods in a class that extends DiffTree.</p> <p>To configure the diff tree to display coverage terms, create the following Gosu class. The package name is provided as an example. You can create the package anywhere.</p> <pre>package gw.plugin.diff.myDiffTreeSubclassConfig.gsrc.gw.diff.tree uses gw.api.diff.DiffItem uses gw.api.diff.DiffAdd uses gw.api.diff.node.ChangeDiffTreeNode uses gw.api.diff.tree.DiffTree uses gw.api.diff.node.AbstractOrderedDiffTreeNode class PADiffTree extends DiffTree { construct(items : List&lt;DiffItem&gt;, configFile : String, diffReason : DiffReason) { super(items, configFile, diffReason) } override function addNodeForAdd(diffAdd : DiffAdd) : AbstractOrderedDiffTreeNode { var addNode = super.addNodeForAdd(diffAdd) if (diffAdd.Bean.typeis Coverage) { diffAdd.Bean.CovTerms.each(\ term -&gt; { var node = new ChangeDiffTreeNode(null, term.DisplayName, null, "''", term.DisplayValue, 1) addNode.addChild(node) }) } return addNode } }</pre>
PC-14144	Previously, availability script verification did not report errors if the return type was not Boolean. Now Studio will verify an availability script when the return type is not Boolean.
PC-14147	<p>Made a change to the ModifierInput widget to fix issues with reflection.</p> <p>If you use the ModifierInput widget as a reflector, the ModifierInput widget and the widget that uses it as a reflector must be in a separate and same InputSet PCF file.</p>
PC-14210	Fixed an issue where a row added to businessowners policy lookup table in Studio caused a personal auto quote to fail.
PC-14282	Added verification on server startup to ensure that each coverage clause has a valid coverage category.
<b>Rating</b>	
PC-14467	<p>In previous versions of PolicyCenter, the policy data could be modified while a quote (a set of costs) was being generated. In the event that the policy data was modified during the time that a quote was being generated (either internally, or through an integration), it is possible that there would be inconsistencies between the values of rating parameters on the policy and for the costs returned. This should have been relatively unlikely with few-second quote times, but would be more likely with longer quote times, such as with complex commercial policies, or when using external rating engines.</p> <p>In this version of PolicyCenter, the rating code locks the PolicyPeriod and the effective dated entities beneath it. As before, rating should only be dependent on parameters reflected in these entities. If you are using non-persisted entities for generating quotes, then you will likely need a different rating path that does not commit to the database, as the default rating code does.</p>
PC-14517	<p>(1) In this release, the existing rating plugin implementation class <code>gw.plugin.policyperiod.impl.DemoRatingPlugin</code> was renamed <code>SysTableRatingPlugin</code>. Also, the previous version of the personal auto rating engine <code>PARatingEngine</code> now has the name <code>PASysTableRatingEngine</code>.</p> <p>(2) An optional new rating plugin implementation implements part of the Rating Management feature. However, the right to use Rating Management is determined by your license agreement. To fully enable Rating Management, you must add a license key to your production configuration. Contact your Guidewire support representative for instructions. Additionally, this feature requires you change the registered plugin implementation for <code>IRatingPlugin</code> to the new class <code>gw.plugin.policyperiod.impl.PCRatingPlugin</code>. For the personal auto line of business, this plugin uses the new rating engine class <code>PARatingEngine</code> (what <code>PASysTableRatingEngine</code> was called in PolicyCenter 7.0.0). For all other lines of business, the plugin calls its superclass <code>SysTableRatingPlugin</code> to create rating engine instances.</p>
<b>Reinsurance</b>	
PC-13559	Improved the user interface of Ceded Premiums → All costs for an agreement. Previously, the display iterated over slices. Now it iterates over agreements. Headings were updated or removed if they did not apply.

<b>ID</b>	<b>Description</b>
PC-14016	<p>Previously, the <code>IReinsurancePlugin</code> interface was dependent upon Guidewire Reinsurance Management. Therefore, it was recommended that you not alter the default implementation of this plugin to integrate with another reinsurance system. The default implementation of this plugin is <code>PCReinsurance.gs</code>. You can now alter <code>PCReinsurance.gs</code> to integrate with another reinsurance system. See the "Reinsurance Management" chapter in the <i>Integration Guide</i>.</p> <p>This was previously a PolicyCenter 7.0.0 Known Issue.</p>
PC-14148	<p>In the default configuration, if a reinsurance program is attached to at least one policy, then you cannot delete the program. Now, if you want to stop PolicyCenter from attaching this program to any more policies, set the effective date and the expiration date of the program to the same value.</p>
PC-14235	<p>The Reinsurance plugins were restructured in this release to make integration and product creation easier. This only affects you if you began Reinsurance integration in the 7.0.0 release.</p>
PC-14253	<p>Fixed a display issue with facultative agreements on a PCF screen for reinsurance if there were two <code>View As Of</code> date ranges. Previously, for the second date range, the screen displayed the facultative agreements that applied to the first date range. This was a display issue only that was corrected after saving as draft.</p>
<b>Reporting</b>	
PC-14619	<p>Fixed this PolicyCenter 7.0.1 Known Issue. Previously, for a PolicyCenter user to be able to view the <code>Report</code> tab, the user must have had a user role that included the <code>PolicyCenter View Report</code> tab (<code>reporting_view</code>) permission. Previously in the base configuration, this permission was set to retired. This permission is no longer set to retired.</p>
<b>Revisioning</b>	
PC-11223	<p>Certain non-revisioned fields are now copied into the new preemption branch from the preempted branch by <code>PolicyPeriodPlugin.gs</code>. Non-revisioned fields are fields which always apply to the entire policy-term. These field values are only copied into the preemption branch if their values changed from the period on which they are based. Unchanged values are not copied, and the preemption branch gets the field values from the preempting branch. If you have implemented your own <code>PolicyPeriodPlugin</code>, you need to copy the new functionality into your custom class. See the <code>copyNonEffDatedFieldsForPreemption</code> method.</p> <p>PolicyCenter copies the following non-revisioned fields:</p> <ul style="list-style-type: none"> <li>• <code>BaseState</code></li> <li>• <code>BillImmediatelyPercentage</code></li> <li>• <code>BillingMethod</code></li> <li>• <code>DepositAmount</code></li> <li>• <code>DepositCollected</code></li> <li>• <code>DepositOverridePct</code></li> <li>• <code>InvoiceStreamCode</code></li> <li>• <code>NewInvoiceStream</code></li> <li>• <code>Offering</code></li> <li>• <code>OverrideBillingAllocation</code></li> <li>• <code>PaymentPlanID</code></li> <li>• <code>PaymentPlanName</code></li> <li>• <code>PeriodEnd</code></li> <li>• <code>ProducerCodeOfRecord</code></li> <li>• <code>RateAsOfDate</code></li> <li>• <code>ReportingPatternCode</code></li> <li>• <code>Segment</code></li> <li>• <code>UWCompany</code></li> <li>• <code>WaiveDepositChange</code></li> </ul>
PC-12178	<p>Previously, an entity was lost after handling preemptions in an out-of-sequence policy change.</p> <p>This was previously a PolicyCenter 7.0.0 Known Issue.</p>
PC-13552	<p>Fixed an issue which occurred if a second cancellation was initiated on a policy which already has a scheduled cancellation. If the second job threw an exception during bind, the policy went into an undesired state.</p>

ID	Description
PC-14009	<p>Fixed a null pointer exception that occurred when a user edited a slice on an out-of-sequence change from a wizard step that is not accessible from the other slice.</p> <p>For example:</p> <ol style="list-style-type: none"> <li>1. Create a CPP submission with 2 lines, and 2 coverables in one line.</li> <li>2. Remove a line/coverable in future change and remove the only other line/coverable in an out-of-sequence Change, thereby creating an out-of-sequence conflict.</li> </ol> <p>Previously, if you navigated to a step which is not available in other slice, PolicyCenter threw a null pointer exception.</p> <p>As part of the fix, a new method was added to JobWizardHelper to handle the case where a period-to-view is changed using slice selector and the wizard step is not accessible. If the step is not available, PolicyCenter displays an error message: <b>The current wizard step is not available as of {date}. Please select different step to change the date.</b></p>
PC-14038	Fixed an issue where differences were appearing for Offerings and Product Modifiers in the Policy Review screen on an out-of-sequence policy change even when no such changes have been made.
PC-14109	<p>Fixed an issue where diff results were not sorted, and appeared in seemingly random order. The diff results are now sorted.</p> <p>In the <code>PolicyPeriodDiffPlugin</code>, the following methods now return a sorted list of diff items:</p> <ul style="list-style-type: none"> <li>• <code>compareBranches(DiffReason, PolicyPeriod, PolicyPeriod) : List&lt;DiffItem&gt;</code></li> <li>• <code>filterDiffItems(DiffReason, PolicyPeriod, List&lt;DiffItem&gt;) : List&lt;DiffItem&gt;</code></li> </ul>
<b>Security</b>	
PC-14119	Fixed an issue in the sample data where users had the permissions of producer code roles that had been removed from the producer code.
<b>Studio</b>	
PC-13482	Fixed unhandled exception in Studio when attempting to add a coverage if no lookup table is defined for the coverable. This exception is now caught and displays an error message.
<b>UI Configuration</b>	
PC-12165	Previously, the unsaved work link of a job wizard indicated that there was unsaved work in situations where the user had not actually made any data changes. The unsaved work link now only indicates that there is unsaved work if the user has made a data change or if the system has automatically made a data change that has not yet been saved. For example, if the user enters a wizard step and makes a change, then navigates away to the account file without saving a draft, the unsaved work link will indicate that there is unsaved work on that wizard step. For another example, if the user enters a wizard step that adds a new required coverage to the policy, the unsaved work link will indicate that there is unsaved work and the user will be notified about the additional coverage with a message in the worksheet area.
PC-13964	Previously, you could set the following user Preferences with very high values: Maximum Recent Accounts, Maximum Recent Policies And Job, and Maximum Recent Contacts. Because this caused run time errors, the value of these fields must be less than 10.
<b>Underwriting Authority</b>	
PC-13919	Fixed an issue where the user received an exception in Risk Analysis on Risk Approval Details - History Table screen.
PC-14252	The <code>gw.job.JobProcessUWIssueEvaluator</code> class was modified to include Referral in the list of underwriting issues checked at all blocking points for submission and renewal. Referral should always be checked, and it should not be omitted from submission or renewal.
PC-14344	On the Risk Analysis screen, there is an iterator that displays a list of UW Issues. If the user invoked quote, bind, or issue on that page and some of the UW Issues were auto-resolved during one of those operations, PolicyCenter threw an <code>IllegalStateException</code> because the iterator was holding on to stale references of UW Issues. This fix invalidates iterators displayed on Risk Analysis step when one of the three actions, quote, bind, or issue, is performed. Specifically, a new method <code>invalidateIterators</code> that was defined on <code>JobWizardHelperEnhancement</code> is invoked.
<b>Upgrade</b>	
PC-14346	<p>Fixed an issue where PolicyCenter threw an exception if you did both of the following:</p> <ol style="list-style-type: none"> <li>1. Quoted a policy in PolicyCenter 4.0</li> <li>2. Using the same database, issued the policy in PolicyCenter 7.0.</li> </ol>
<b>User Tools</b>	

ID	Description
PC-12767	Previously, If virtual properties on a Job like AllOpenActivities, Documents, or AllActivities are accessed when the job is uncommitted, a stack trace was shown in the user interface. Now, an empty query result of the property type is returned.
PC-13966	The <b>Assign Activity</b> user interface has changed so that it goes to a popup screen within the worksheet where you can search for and assign users.
PC-14385	The Related to field on the <b>Activity Detail</b> screen always displayed empty. Therefore, this field has been removed.

## Platform Improvements and General Issues

ID	Description
<b>Core</b>	
PL-9920	Modified <code>gw.api.database.Query</code> to take a Bean type instead of <code>KeyableBean</code> .
PL-11256	Gosu now throws <code>IllegalArgumentException</code> if you attempt to call <code>get/setFieldValue</code> using an <code>IPropertyInfo</code> that comes from a delegate.
PL-11789	Guidewire has removed the <code>Server Tools → Info Pages → Database Info</code> page. For more information on the database instrumentation available, please see the documentation.
PL-13605	Guidewire has added a new query restriction, <code>compareIgnoreCase</code> . You can use it to compare a character-based column against a character value or another character-based column.
PL-13812	Within the application interface, the <code>Desktop → My Activities</code> and <code>Desktop → My Queues</code> views now show a star icon to indicate a new or updated activity.
PL-15395	It is now possible to concatenate values in SQL during an upgrade through <code>BeforeUpgradeDBFunction.Concatenate()</code> .
PL-15431	Guidewire now logs the <code>ConcurrentDataChangeExceptions</code> generated by distributed workers at different levels depending on context. If the <code>ConcurrentDataChangeException</code> occurs on processing the work items, PolicyCenter logs the error only if the number of attempts exceeds the configured value of the <code>WorkItemRetryLimit</code> configuration parameter. Otherwise, PolicyCenter logs the debug message instead.
PL-15432	Guidewire has implemented a new batch process, <code>ProcessCompletionMonitor</code> . It will wake up at user-configurable intervals and examine the Process History and Work Queue for all batch processes. Then, it will invoke the user plugin <code>IBatchCompletedNotification</code> after the batch process is done to permit user code to react to the completion of the batch process. It will also set <code>ProcessHistory.NOTIFICATIONSENT</code> to true for that batch process to ensure that PolicyCenter invokes the plugin a single time only.
PL-15826	Guidewire has added a new configuration parameter, <code>GlobalCacheSizeMegabytes</code> that you can use to set the cache size to be larger than 2 GB. If you specify this value, it takes precedence over <code>GlobalCacheSizePercent</code> .
PL-15850	During upgrade, if there is a column that is being encrypted or decrypted as part of the upgrade, then the Upgrader drops the index on that column, if one exists.
PL-15888	Added a new optional attribute, <code>forceEqMatchType</code> , that you can add to the <code>Criterion</code> element in a search configuration file. If this attribute is present, it must be the name of a Boolean property on the criteria entity. <ul style="list-style-type: none"> <li>• If this attribute evaluates to true, the <code>Criterion</code> uses an <code>eq</code> (equality) match.</li> <li>• If this attribute evaluates to false, the <code>Criterion</code> uses the <code>matchType</code> specified in the <code>Criterion</code>.</li> </ul> <p>For example:</p> <pre>&lt;Criterion property="StringProperty"            forceEqMatchType="FlagProperty"            matchType="startsWith"/&gt;</pre> <p>This search criteria uses a <code>startsWith</code> match for <code>StringProperty</code> unless the <code>FlagProperty</code> on the criteria entity is true, in which case it uses an <code>eq</code> match type.</p>
PL-15962	Fixed an issue in which stopping the global cache did not stop any currently running background thread(s) and actually started them if they were not already started.
PL-15963	Corrected an issue with a null pointer in a global cache reaper thread that prevented the PolicyCenter server running on WebSphere from shutting down gracefully. The issue caused the following error to occur:
	<code>ERROR Unexpected error while in global cache reaper thread: java.lang.RuntimeException: java.lang.NullPointerException</code>
PL-16066	Guidewire provides a <code>dependencies.xls</code> file in the root of the application <code>modules</code> directory. This file lists the following information for all non-Guidewire generated JAR files (third-party JAR files): <ul style="list-style-type: none"> <li>- Jar name</li> <li>- Publisher</li> <li>- License</li> </ul>

ID	Description
PL-16087	<p>Guidewire has modified the behavior of copy-starter-resources in two ways:</p> <ul style="list-style-type: none"> <li>For files, each time it runs, it checks to see if the file is missing from the destination, and copying the file if it is.</li> <li>For directories, each time it runs, it checks to see if the root of that directory exists in the destination. If not, it copies over the directory and its contents. Again, it only does this if the root of that directory is missing, it does not check the contents.</li> </ul> <p>For example, if the directory config/resources/productmodel is missing from the destination module it will be copied. However, if that directory exists in the destination, then it is not copied even if some of the contents are missing. This allows you to prevent the re-copying of files by leaving an empty root directory.</p>
PL-16171	<p>Guidewire has added a new method, DateDiff, to class DBFunction. The method takes two ColumnDef parameters to date fields in the database:</p> <ul style="list-style-type: none"> <li>a start date</li> <li>an end date</li> </ul> <p>The method returns the interval between the two. It has an additional parameter that you use to specify the unit of measure for the result:</p> <ul style="list-style-type: none"> <li>DAYS</li> <li>HOURS</li> <li>SECONDS</li> <li>MILLISECONDS</li> </ul> <p>If the end date precedes the start date, the function returns a negative value instead of a positive one.</p>
PL-16553	<p>Guidewire no longer exposes internal implementation details of a query through the undocumented RawQuery property of the query builder APIs. The RawQuery property is removed.</p>
<b>Database</b>	
PL-8018	<p>Configurable linguistic searching is now available on SQL Server.</p>
PL-10022	<p>Guidewire now stores Oracle LOB (large objects, CLOBs and BLOBs) in a separate tablespace if a LOB tablespace is configured. This is optional.</p>
PL-12214	<p>Modified the behavior of (and slightly renamed) the DBMS Counter Threshold (millis) option on the (Server Tools) Guidewire Profiler configuration page:</p> <ul style="list-style-type: none"> <li><i>Name change</i> – Renamed DBMS Counter Threshold (millis) to DBMS Instrumentation Capture Threshold for each Action (millis).</li> <li><i>Behavior change</i> – Formerly, this option generated a DBMS report if any database operation exceeded a preset threshold value. Now, the profiler generates a DBMS report if the action (start of profiling session, end of profiling session) exceeds the threshold value set by this option.</li> <li><i>New option parameter Diff DBMS Instrumentation Counters</i> – (Boolean) Set to capture the DBMS counters at the beginning of the profiling session and to include analysis of the differences in the DBMS-specific report. You must set the DBMS Counter Threshold option for this option to be meaningful.</li> </ul>
PL-15852	<p>The upgrade process drops the upgrade instrumentation tables during a database upgrade. If any table contains data, the upgrade utility logs a warning message, rather than throwing an upgrade error.</p>
PL-15859	<p>Fixed an issue in which upgrade was not triggering the MigrateColumnToLargeID version trigger as it should. This version trigger migrates ID columns to the larger datatype.</p>
PL-15883	<p>Guidewire now persists the results of consistency checks to the database for use of performance monitoring. You can monitor the progress of a particular consistency check run by using SQL queries against the associated tables, which are:</p> <pre>* xx_dbconsistcheckrun * xx_dbconsistcheckqueryexec</pre> <p>The prefix, xx, is the application prefix (ab, bc, cc, or pc).</p>
	<p>Entity DBConsistCheckRun represents a run of the consistency checker. It contains an array of DBConsistCheckQueryExec entities, which represent the execution of an individual consistency check within a run.</p>
PL-15947	<p>If you are configuring an Oracle database's tablespace mapping, it is possible to optionally configure a tablespace for LOB (large objects) to be stored. For example,</p> <pre>&lt;tablespacemapping logicalname="LOB" physicalname="GW_LOB"/&gt;</pre>
PL-16008	<p>Guidewire database statistics process now gathers statistics for indexes defined in extensions regardless of the case in which the index name is specified.</p>

ID	Description
PL-16643	Previously, if you turned off <b>Web Profiling</b> , you had to navigate to <b>Guidewire Profiler</b> through the <b>Internal Tools</b> page, losing your current context, and generating extra mouse clicks. With this change, you now see your profile analysis data page immediately. You can analyze it, then return back to your previous location in the application.
<b>Geocoding</b>	
PL-15467	<p>Improved the performance of proximity search. To support this change, Guidewire has added the following:</p> <ul style="list-style-type: none"> <li>A new configuration parameter — <code>ProximityRadiusSearchDefaultMaxResultCount</code>, which is the maximum number of results to return if performing a radius (n miles or kilometers) search from PolicyCenter. The default is 1000. This parameter has no effect on ordinal (nearest n) proximity searches. This parameter does not have to match the value of the corresponding parameter in the <code>ContactManager config.xml</code> file. This parameter exists within ContactCenter as well, but behaves slightly differently. See the ContactCenter documentation for details.</li> <li>A new field, to <code>ProximitySearchParameters</code> — <code>RadiusSearchMaxResults</code> indicates the maximum number of items to return on a radius search: <ul style="list-style-type: none"> <li>If <code>DistanceBasedSearch</code> is <code>false</code>, then PolicyCenter ignores this value</li> <li>If <code>DistanceBasedSearch</code> is <code>true</code>, and the value of <code>RadiusSearchMaxResults</code> is <code>null</code>, zero, or negative, PolicyCenter limits the search by the value of <code>ProximityRadiusSearchDefaultMaxResultCount</code>.</li> </ul> </li> </ul> <p>In addition, Guidewire has changed the default value of <code>ProximitySearchOrdinalMaxDistance</code> from 100 to 300.</p>
<b>Gosu</b>	
PL-8320	Corrected an issue that occurred while printing large <code>BigDecimal</code> values, in which PolicyCenter would incorrectly truncate the value.
PL-8587	Guidewire now formally disallows a <code>return</code> , <code>break</code> , or <code>continue</code> statement in a <code>finally</code> statement block.
PL-10189	Guidewire has deprecated the ability to partially define the package structure of a class in a <code>uses</code> statement and to complete the rest of the package structure in a variable typing definition.
PL-16263	Gosu now supports <code>!==</code> for identity comparison.
PL-16279	Gosu generates <code>VerifyError</code> if you declare a function to return a primitive and the function contains a statement that returns <code>null</code> .
PL-16281	<p>Added the ability for Gosu to support top-level enumerations, outside the class, for example:</p> <pre>package bar     enum Language { .... }     class Foo{         ...     }</pre> <p>Thus, you no longer need to embed the enum declaration within a class by doing the following:</p> <pre>class Foo {     enum Language { .... } //No longer required to be within a class declaration }</pre>
<b>Integration</b>	
PL-8969	Guidewire now provides a <code>compatibility-xsd.xml</code> file, copies of which exist in <code>.../modules/p1/config/registry</code> and in <code>.../modules/pc/config/registry</code> . This file contains a list of the XSD files that exist in the application.
PL-10705	Guidewire no longer supports Gosu class methods that have a parameter type or return type of <code>org.apache.axis.types.HexBinary</code> in web services published as RPC-Encoded SOAP services. Instead, make the parameter or return type a <code>String</code> value and encode this string as a Hex values. If you want, you can also use a more optimized encoding scheme.
PL-13916	Fixed an issue in which a <code>soap.*.entity.*</code> array type's <code>ComponentType</code> was not properly proxied

ID	Description
PL-14321	<p>Added configuration parameter <code>UseSafeBundleForWebServiceOperations</code>, which changes the behavior of bundle commits in web services published on this server. The default value is <code>true</code>.</p> <ul style="list-style-type: none"> <li>If set to <code>false</code>, the application ignores entity version conflicts as it commits a bundle.</li> <li>If set to <code>true</code>, the application detects (and does not ignore) entity version conflicts.</li> </ul> <p>This bundle behavior affects RPC-Encoded web services only.</p> <p>See the description of this configuration parameter in the <i>PolicyCenter Configuration Guide</i> for more details.</p>
PL-15303	Fixed an issue with WSDL refreshment. If you refreshed the WSDLs that Guidewire defines for a base configuration application-to-application integration, PolicyCenter placed the files into the wrong application folder. This resulted in a checksum error the next time that you started the application server.
PL-15467	<p>Modified how PolicyCenter handles proximity searching:</p> <ul style="list-style-type: none"> <li>Added a <code>RadiusSearchMaxResults</code> property to the <code>ProximitySearchParams</code> non-persistent entity. This property indicates the maximum number of items to return on a radius search. PolicyCenter ignores this value if the value of <code>DistanceBasedSearch</code> is <code>false</code>. If the value of <code>DistanceBasedSearch</code> is <code>true</code>, and the value of <code>RadiusSearchMaxResults</code> is <code>null</code>, zero, or negative, then PolicyCenter limits the search by the value of <code>ProximityRadiusSearchDefaultMaxResultCount</code>.</li> <li>Added configuration parameter <code>ProximityRadiusSearchDefaultMaxResultCount</code> with a default of 1000. This parameter sets the maximum number of results that a proximity search is to return if using a fixed radius (n miles/km). PolicyCenter ignores this parameter if performing an ordinal (nearest-n) search.</li> </ul>
PL-15985	Fixed an issue in which GX models were re-declaring namespaces whenever an external model was referenced. All namespaces are now declared on the root element.
PL-16011	Fixed an issue in which the Find Type (CTRL-N) functionality in Studio did not correctly find file <code>suite-config.xml</code> . File <code>suite-config.xml</code> now properly appears in Studio and does not require any work-arounds. This file overrides URLs for web services for integrations between Guidewire applications.
PL-16017	Changed the namespaces of all base configuration web services to have a consistent naming convention. This will cause any WSDL generated by a previous version of PolicyCenter to be invalid and will necessitate a re-fetch by any remote application.
PL-16195	Fixed an issue in which <code>Wsd12Gosu</code> and the Studio <b>Webservice Collection</b> editor would not fetch resources referenced by <code>xs:include</code> or <code>xs:redefine</code> .
PL-16310	Improved the performance for XML generation of large Guidewire XML models.
PL-16338	Guidewire has added a new configuration file <code>suite-config.xml</code> and reworked the Guidewire application-to-application integrations to refer to the URL of the other products of the suite using <code>#{xc}</code> as a placeholder in other locations of the configuration. (Valid xc values are ab, bc, cc, and pc.)
PL-16599	For a WS-I annotated Gosu class, it is not possible to use a parameter and return type of <code>wsi.local.*</code> . Guidewire designates <code>wsi.local.*</code> and <code>soap.local.*</code> as restricted package names. Do not use these package names for any class that you create.
PL-16698	WS-I now use the SOAP 1.1 envelope XSD specified by the WS-I BP, rather than the official one from the W3C. This allows better interoperability with .NET web services, which expect to be able to set the <code>xml:lang</code> attribute on the <code>faultstring</code> element in a response envelope.
PL-16799	Fixed an issue with the copy-on-edit functionality in Studio that created checksum errors resulting in verification failing.
<b>Miscellaneous</b>	
PL-16996	Pressing ALT+Enter in a Studio editor to create a new display key did not work as expected. Instead, it generated errors.
<b>Persistence</b>	
PL-10982	Setting <code>Use Entity Name</code> to <code>true</code> for a type of the entity path that is not an entity type generates an error at server start-up rather than throwing a <code>RuntimeException</code> . This behavior also occurs if you use the <code>Verify</code> command in Studio on the Display Name editor.
PL-16777	Fixed an issue with missing methods for external entities. The fix required a change to the <code>Entity</code> interface to expose the following methods:
	<ul style="list-style-type: none"> <li><code>public Entity[] getChangedArrayElements(String array)</code></li> <li><code>public boolean isArrayElementAddedOrRemoved(String array)</code></li> <li><code>Entity shallowCopy()</code></li> </ul>
<b>Studio</b>	

ID	Description
PL-12800	Studio now updates the Entity TypeLoader on every save of the data model editor. This eliminates the need to restart Studio before data model changes can be reflected elsewhere (in Studio).
PL-13244	Users can now access resources dropped into Studio as XSD files by Studio's Open Type feature (Ctrl-N).
PL-15863	Fixed an issue in which performing a search in Studio (ALT+F7) for the usage of a method did not return the correct usages of that method. You <b>must</b> drop the TypeInfo database after you apply the fix.  See discussion under "Upgrade Issues" on page 276 for more details.
PL-15918	Improved Studio memory usage.
PL-16027	Hitting the <b>Escape</b> key while a confirmation dialog box is open now closes the dialog box. This is the same behavior as hitting the <b>Cancel</b> button.
PL-16336	Resolved performance issues related to running the Studio Verify All command after renaming a PCF file.
PL-16948	Fixed a Studio performance issue in which running the Verify All command multiple times resulted in successively longer and longer execution times.
<b>Web</b>	
PL-11323	Added the ability to specify an <code>onEnter</code> attribute on a wizard step element within a wizard PCF. The Gosu in such an attribute runs whenever the step is entered in the user interface.
PL-16258	Autocomplete now works when multiple shared PCF sections exist on the same page.
PL-16576	Added an additional PCF configuration option ( <code>isDraftOnEnter</code> ) to determine, upon entering a wizard, whether or not unsaved work gets highlighted prior to changes being made.
PL-16975	Fixed a PolicyCenter security vulnerability that permitted a phishing attack through the manipulation of frames.

## Known Issues and Limitations

This section describes known issues with this release of Guidewire PolicyCenter.

- PolicyCenter Known Issues
- Studio/Platform Known Issues

### PolicyCenter Known Issues

#### Studio: “Verify Changed Resources” is not implemented in the product model (PC-6928)

**Issue** – The Studio Tools → Verify → Verify Changed Resources command does not work with the PolicyCenter product model.

**Workaround** – Use one of the other verification commands instead. See “Validating Studio Resources” in the *PolicyCenter Configuration Guide* for details.

#### Provide a way to filter out added and removed subtentities 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.

#### Need a summary of underwriting issues across all policies in a job group (PC-9102)

**Issue** – The JobGroupRiskAnalysisLV needs to show a summary of underwriting issues across all policies in the job group.

**Status** – Guidewire is aware of this issue.

## [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. 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 original answer is still saved along with the other answers.

**Workaround** – The answers to these *irrelevant* questions 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)
        }
    }
}
```

## [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 screen of the agent
- 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.

## [Problem 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 problem can occur with fields that are not base data.

**Workaround** – Add code that copies this type of data to the other versions.

## [Rate Table Maintenance: Rate book detail localization table visible when there is only one language configured; it shouldn't be \(PC-13799\)](#)

**Issue** – Localization tables are made available for features when Policy Center is configured to support multiple locales. However, the **Rate Book Detail** view displays the localization table when there is only one locale supported by the Policy Center application.

**Status** – Guidewire is aware of this issue.

## Rate Table Maintenance: Rate table definition Usage tab not refreshed properly on cancel edit to reflect changes to usage (PC-13868)

**Issue** – When accessing the rate table definition detail page for a particular definition, the related information is loaded in the view. For performance, PolicyCenter stores the related information just in case the user wishes to cancel changes. If you leave a rate table definition in edit mode, and rate books are updated to include or exclude this definition within this time, cancelling the rate table definition edit will *not* refresh the **Usage** tab to reflect rate book changes. Cancelling the update restores the rate table definition to the original values initially loaded when you accessed this page.

**Workaround** – If you know changes have been made to the usage of your rate table definition, cancel the edit. Then return to the rate table definition search and re-select the rate table definition. The **Usage** tab will be reloaded with the latest changes. In general, do not leave rate table definitions in edit mode for a long period of time.

## Rate table maintenance: navigation back to rate book lost after import validation error raised (PC-14010)

**Issue** – When importing rate table details from a Microsoft Excel file and validation errors are encountered preventing the import, you need to cancel the import to return to the rate table detail view. When you return to the detail view, the link that is normally available to allow a user to return to the related rate book disappears.

**Status** – Guidewire is aware of this issue.

## Rate table maintenance: During rate table import from Excel, UI displays raw exception when importing a non-Excel file (PC-14127)

**Issue** – When selecting a non-Excel file for import into a rate table, PolicyCenter reports a system error instead of displaying a user-friendly error message.

**Status** – Guidewire is aware of this issue.

## Rate table maintenance: Import rate book raises error rate table exists on first import, passes on second attempt (inconsistently) (PC-14134)

**Issue** – In some cases when importing rate books, PolicyCenter raises an import error indicating that a non-existent rate table exists in the target environment. The error is incorrect, and import should be successful in this scenario.

**Workaround** – Import the rate book a second time. The error is not reported and the import is successful on the second attempt.

## Rate table maintenance: Slow performance loading page for large rate table in rate book (PC-14492)

**Issue** – Slow performance when loading the page for a rate table with over 3000 rows.

**Workaround** – Guidewire is aware of this issue.

## Rate Table editing: factor of data type double only saves total of four digits on import from Excel (PC-14657)

**Issue** – This issue occurs if you are using Excel to edit rate table information. If you enter factor values (factor is using data type double) exceeding four digits, factors get truncated to 4 digits on import. For example:

- 11.12345 would be imported as 11.12
- 1.12345 would be imported as 1.123

**Workaround** – Import the data and update the values accordingly within PolicyCenter. The limitation on digits does not exist in PolicyCenter. Alternatively, if the values are generally using more than 4 digits, limit your data entry to PolicyCenter.

#### [Linked address is not updated when contact management system updates a contact \(PC-14719\)](#)

**Issue** – When two or more contacts have linked addresses, and one of those addresses is updated in an external contact management system, the address is updated on both contacts in PolicyCenter as it should be. However, the address does not get updated in the contact management system on the second contact, leaving the contact inconsistent between the external contact management system and PolicyCenter. This is a specific instance of the general problem seen in PC-14763.

**Workaround** – See the workaround for PC-14763 below.

#### [Contact.AccountHolderCount incorrectly counts the number of accounts \(PC-14736\)](#)

**Issue** – Contact.AccountHolderCount incorrectly counts the number of accounts a particular contact is the account holder on.

**Workaround** – Query the database to count the accounts on which this contact has the account holder role.

#### [Contacts created from the Contact tab are not autosaved \(PC-14761\)](#)

**Issue** – Contacts created from the **Contacts** tab are not getting autosaved. The **UnsavedWork** light correctly goes on, but when the user clicks elsewhere, the work is lost.

**Workaround** – Set the **autosaveable** attribute on the **NewContact.pcf** to **true**.

#### [After calls to updateContact from contact management system, updates are randomly lost \(PC-14763\)](#)

**Issue** – After calls from a contact management system to **updateContact** in PolicyCenter, updates from PolicyCenter to the contact management system can be lost because of issues with the thread local resources.

**Workaround** – Do not set the token in **updateContact** to **ab**. You may need to set up your integration between PolicyCenter and the contact management system to not rebroadcast updates back to the contact management system.

## Studio/Platform Known Issues

#### [Cluster fails on Linux using IPv6 IP stack \(PL-9240\)](#)

**Issue** – There are issue starting an application cluster if you are using the IPv6 IP stack on Linux.

**Workaround** – Specify the following as a JVM start-up option:

```
-Djava.net.preferIPv4Stack=true
```

Set this option for all the JVM's participating in cluster.

#### [First time you click on the arrow of the typekey input, the drop-down menu will not open \(PL-10134\)](#)

**Issue** – The drop-down menu does not open on the first click of the arrow on a typekey input. Instead, the help text opens.

**Workaround** – Turn off help text on focus by setting `InputHelpTextOnFocus` to `false` in the `config.xml` file. By doing that, the help text shows only if you mouse over the input and will not interfere with opening a drop down menu.

### XML API upgrade feature missing from documentation (PL-10257)

**Issue** – The *Integration Guide* describes a new set of XML APIs based on the `XmlElement` class. (Legacy APIs are based on the `XmlNode` class.) You can continue to use the legacy APIs. However, the *Integration Guide* omits mentioning an additional upgrade-specific feature.

**Workaround** – For backwards compatibility only, you can import an XML schema into the Gosu type system using the legacy XML system by following these instructions:

1. Copy:

`PolicyCenter/modules/p1/config/registry/compatibility-xsd.xml`

To:

`PolicyCenter/modules/configuration/config/registry/compatibility-xsd.xml`

2. Add an entry for your schema. Set the value of the `namespace` attribute to the Gosu package name of the schema. For example, if the schema is in the package location `my.package` and is called `myschema.xsd`, set the value of `namespace` to `my.package.myschema`.

### Studio Rules do not use correct capitalization for root object's name (PL-10740)

**Issue** – Rule set root objects are not named with first letter lower-cased.

**Workaround** – Rules engine issues a warning if the correct case for objects is not being used.

### Countries configured in zone-config.xml still generate a warning during regen-dictionary even when zone data is loaded for all of these countries (PL-11947)

**Issue** – Countries configured in `zone-config.xml` still generate a warning during regen-dictionary even when zone data is loaded for all of these countries.

**Workaround** – Warning message is created in error and can safely be ignored.

### The user interface cannot handle starting multiple instances of a batch process (PL-12372)

**Issue** – The user interface cannot handle multiple instances of a batch process.

**Workaround** – If multiple instances of a batch process need to be executed, they must be started from the command line. Also, you should ensure that the `BatchProcess.isExclusive()` method returns `false` to allow multiple instances to be run simultaneously.

### Type system refresh after PCF page title change does not update corresponding menu label (PL-13057)

**Issue** – The type system refresh after the PCF page title change does not update corresponding menu label.

**Workaround** – After updating a page title, the server restart must be done to refresh menu labels and avoid null pointer exceptions due to stale reference.

### There is a length limitation on entity localization table names (PL-13360)

**Issue** – There is a length limitation on entity localization table names.

**Workaround** – Ensure that localization `tableName` property specified in the entity extension file is less than 16 characters. The error message generated if the localization table name exceeds the maximum length indicates that 18 characters are allowed, but that does not account for two additional characters added by the application.

## [US-Locations.txt file with the US geodata from GreatData has special characters that cause validation problems with United States Postal Service \(USPS\) data \(PL-13384\)](#)

**Issue** – The `US-Locations.txt` file contains information that does not conform to United States Postal Service (USPS) standards for bulk mailings.

**Workaround** – The provided `US-Locations.txt` file is intended only for use in geocoding to identify addresses for a location. You can process the `US-Locations.txt` file to conform to your particular address standards, and then import that version of the file instead.

## [NULL sort order reversed in Oracle versus SQL Server \(PL-13559\)](#)

**Issue** – Gosu queries in a PolicyCenter instance that runs on Oracle orders nulls at the end of the result, not the beginning. This is the opposite of the null ordering for SQL Server queries.

**Workaround** – Guidewire is aware of this issue.

## [GX models that reference virtual fields and enhancements throw null pointers if null \(PL-13560\)](#)

**Issue** – The GX models that reference virtual fields and enhancements throw null pointers when they are null.

**Workaround** – Ensure that null checks and error handling is included so that if referenced virtual fields or enhancements are null, then there will not be a null pointer exception.

## [Sending email with file attachment with unicode filename is not correctly handed over to the mail server \(PL-13582\)](#)

**Issue** – Sending email with file attachment with unicode filename is not sent to the mail server correctly.

**Workaround** – Use Latin characters for file names on attached files.

## [The GX model generated XSD cannot be parsed by JAXB \(PL-13598\)](#)

**Issue** – The GX model generated XSD cannot be parsed by JAXB.

**Workaround** – Add JAXB annotation elements to the XSD to specify the necessary metadata (such as class names) for JAXB to generate the Java class files. Contact Guidewire Support for an example XSD annotated in this way.

## [JavaToolkit.gs has incorrectly hard coded memory, which results in failed `regen-java-api` Ant task \(PL-13663\)](#)

**Issue** – The `JavaToolkit.gs` has hard coded memory which can result in failed `regen-java-api` Ant tasks.

**Workaround** – Increase the size of the maximum heap setting on line 161 of `JavaToolkit.gs` in the Ant module. The default value is 512.

## [Cannot make a field from a delegate a localized column \(PL-13761\)](#)

**Issue** – You cannot make a field from a delegate a localized column.

**Workaround** – Move the column to be localized off the delegate and onto each of the implementing entities. Then, in order to make the column appear as though it exists on the delegate, define an enhancement property on the delegate, that *delegates* to the appropriate column, depending on the implementing entity.

## Failed to reload changed Gosu plugin class with HotSwap JVM installed (PL-13913)

**Issue** – Attempting to modify the plugin registry by specifying a different plugin class generates the following exception while performing tasks within PolicyCenter:

```
Transaction-1284594079621 4 16:41:19,720 WARN Configuration Class not found during reload:  
    libraries.NoteUITest  
Transaction-1284594079621 4 16:41:19,737 INFO Configuration reloadChangedClasses found a total of 0  
    classes to reload  
Transaction-1284594079621 4 16:41:19,737 INFO Configuration Finished reloading Gosu classes in 67ms  
SocketListener0-3 4 16:41:19,737 INFO Server.RunLevel PCFs reloaded.  
...
```

This exception occurs if you update the plugin registry to specify that a plugin uses a different plugin class. (You do this by entering a different class name in the **Class** field in the plugin registry screen for that plugin.) The same exception occurs if you also do the following:

- Attempt to reload the changed plugin class name in a method body.
- Create a new method that uses the changed plugin class name.

**Workaround** – In all cases, simply restart the PolicyCenter application server. This causes PolicyCenter to recognize the changed class name.

## Studio test functionality not working correctly (PL-15153)

**Issue** – If attempt to run a test in the **Tests** folder in the Studio Resources tree, running the test gives the following exception:

```
Using Test Environment Delegate: com.guidewire.testharness.ConfigEnvTestEnvironmentDelegate  
gw.internal.gosu.parser.RuntimeExceptionWithNoStackTrace: java.lang.ClassNotFoundException:  
    qa.DothisTest  
Caused by: java.lang.ClassNotFoundException: qa.DothisTest  
at gw.internal.gosu.parser.TypeLoaderAccess.getIntrinsicTypeByFullName(TypeLoaderAccess.java:522)  
at gw.internal.gosu.parser.TypeSystemImpl.getByFullName(TypeSystemImpl.java:139)  
at gw.lang.reflect.TypeSystem.getByName(TypeSystem.java:116)  
at com.guidewire.studio.junit.ui.RunTestCommand.buildTestSuite(RunTestCommand.java:109  
...
```

**Workaround** – Run the following command before you attempt to run a test:

```
.../bin/gwpc dev-deploy
```

## Problem with *regen-java-api* command and JAR files (PL-16351)

**Issue** – If you run the `PolicyCenter/bin/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 resulting 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
```

into the following directory:

```
PolicyCenter/java-api/lib
```

## Ant task incorrectly generates stub Java classes for WS-I (PL-16370)

**Issue** – If you run the `regen-soap-api` Ant task with the `gwpc` tool, it generates directory `PolicyCenter/soap-api` with files relating to published SOAP APIs. It is intended behavior to generate WSDL for WS-I web services (the new style) and RPCE web services (the older style). However, there is a known issue in which the Ant task generates Java class stub files for SOAP client use for both types of web services. The JAR files are supposed to contain only RPCE published web services, not WS-I web services.

**Workaround** – Do not use the Java SOAP client classes for WS-I web services that PolicyCenter generates in the `java-api` folder. They are unsupported.

If you need Java SOAP client classes, use the `wsimport` tool to generate these classes instead. For information about how to use the `wsimport` tool, see “Publishing Web Services (WS-I)” in the *Integration Guide*.

### [Renaming method or property throws ‘ParseResultsException’ \(PL-16633\)](#)

**Issue** – If you rename a property or method, or 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 into PolicyCenter using an administrative account.
2. Access **Internal Tools** → **Reload**.
3. Click **Reload Workflow Engine**.

### [functionaljava-3.0 JAR removed \(PL-16840\)](#)

**Issue** – Guidewire does not support the use of the `functionaljava-3.0` JAR in production environments and is removing it from the application install files.

**Workaround** – Do not use this JAR file for any reason.

### [Keyboard navigation in IE 9 goes to URL bar instead of first editable field \(PL-16915\)](#)

**Issue** – If you are using keyboard commands to navigate within Microsoft Internet Explorer 9, IE highlights the URL bar on the first tab, instead of the first editable field.

**Workaround** – This issue is caused by a change in the IE 9 rendering engine.

### [Expanding or collapsing a tree node in IE 9 causes screen ‘flash’ \(PL-16920\)](#)

**Issue** – If you are using a tree table within Microsoft Internet Explorer 9 and you expand or collapse a tree node, IE refreshes the screen, causing it to ‘flash’ momentarily.

**Workaround** – This issue is caused by a change in the IE 9 rendering engine.

### [PolicyCenter throws NPE when changing Gosu in Studio connected to application server \(PL-16935\)](#)

**Issue** – The following sequence of actions causes a Null Pointer Exception to show within PolicyCenter:

1. Start the PolicyCenter application server.
2. Start PolicyCenter Studio.
3. Log into PolicyCenter.
4. Connect Studio the running application server.
5. Modify any Gosu class within Studio.
6. Navigate to any screen within PolicyCenter.

This sequence generates a Null Pointer Exception similar to the following:

```
[java] ... 2011-06-29 13:45:56,166 ERROR UserInterface Unhandled exception during Render
[java] java.lang.NullPointerException
[java] at gw.internal.gosu.parser.GosuProgram_Proxy.
      getBackingClass(gw.internal.gosu.parser.GosuProgram_Proxy:2)
...

```

**Workaround** – Do one of the following:

1. If you are logged into PolicyCenter under an administrative account, press ALT+SHIFT+L to reload the PCF files into the PolicyCenter rendering framework.

2. Log out of PolicyCenter, then back into it.

### Exposed methods on entity and delegate interfaces (PL-17223)

**Issue** – Guidewire has unintentionally exposed certain methods and properties on entity and delegate interfaces to Gosu. *Guidewire intends to hide these methods from Gosu in a future release. Do not use the following methods in your implementation.*

On most entities:

- `entity.calculateNextVersion()`
- `entity.shouldTriggerValidation()`
- `entity.OkToRetire`

On Contact, Person, Company, and similar contact subtypes:

- `entity.iterateContactGraph()`
- `entity.OfficialIDBases`
- `entity.addToOfficialIDs( OfficialIDType type, State state)`

*Guidewire currently exposes the following to Gosu. However, contact Guidewire Support before using them.*

On Contact subtypes and PolicyLocation:

- `entity.ValidationLevel`
- `entity.DefaultValidationLevel`

On subtype entities:

- `subtype.Subtype`

**Workaround** – Guidewire is aware of this issue.

### Changing ‘Range Radio Button Input’ widget in IE 9 causes screen ‘flash’ (PL-17366)

**Issue** – If you are using Microsoft Internet Explorer 9 and you change the choice on a Range Radio Button Input screen widget, IE refreshes the screen, causing it to ‘flash’ momentarily.

**Workaround** – This issue is caused by a change in the IE 9 rendering engine.



# Guidewire PolicyCenter 7.0.2 Release Notes

## Release 7.0.2

**Document Revision:** 20-May-2014

Copyright © 2001-2014 Guidewire Software, Inc. All rights reserved.

Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Reinsurance Management, Guidewire ContactManager, Guidewire Vendor Data Management, Guidewire Client Data Management, Guidewire Rating Management, Guidewire InsuranceSuite, Guidewire ContactCenter, Guidewire Studio, Guidewire Product Designer, Guidewire Live, Guidewire DataHub, Guidewire InfoCenter, Guidewire Standard Reporting, Guidewire ExampleCenter, Guidewire Account Manager Portal, Guidewire Claim Portal, Guidewire Policyholder Portal, ClaimCenter, BillingCenter, PolicyCenter, InsuranceSuite, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks, service marks, or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries.

All other trademarks are the property of their respective owners.

**This material is confidential and proprietary to Guidewire and subject to the confidentiality terms in the applicable license agreement and/or separate nondisclosure agreement.**

Guidewire products are protected by one or more United States patents.

## Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Support

- Issues and Major Changes
- Improvements and General Issues
- Known Issues and Limitations

## Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

### **IMPORTANT**

**IMPORTANT** If you have missed 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 7.0.2.

- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 3.0.1 or later.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 6.0.0 or later.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 7.0.0.

## 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 Resource Portal – <http://guidewire.custhelp.com>

## Issues and Major Changes

This section contains issues and major changes that may affect your installation.

- Upgrade Tool does not handle rules upgrade properly (PL-18021)
- Archiving Not Supported
- ContactManager Integration
- Studio Product Model Improvements
- Geocoding Using Bing and MapPoint (PL-16708)
- Rate Table Maintenance Improvements
- Base PCF File Changes

- Base Resources Changes
- Changes in This Release Provided in Upgrade Diff Report

## Upgrade Tool does not handle rules upgrade properly (PL-18021)

**Issue** – The upgrade step to handle rules upgrades in PolicyCenter 7.0 does not work correctly.

**Workaround** – Do the following:

1. Create the following subdirectory:

```
PolicyCenter7\modules\configuration\config\rules\rules\PRIORCUSTOM
```

2. Copy the rules subdirectories from the following location:

```
PolicyCenter7\tmp\cfg-upgrade\modules\configuration\config\rules\rules
```

to the following location:

```
PolicyCenter7\modules\configuration\config\rules\rules\PRIORCUSTOM
```

**Note:** If you are upgrading from PolicyCenter 3.x to PolicyCenter 7.x, the rules subdirectories in `PolicyCenter7\tmp\cfg-upgrade` does not have the required .grs files. In this case, copy the rules subdirectories from the directory `PolicyCenter6\modules\configuration\config\rules\rules` instead.

3. Start or restart studio and manually move and merge the customer rules from PRIORCUSTOM into the target base configuration rule sets as required.
4. After you merge all of the customer rules, delete the PRIORCUSTOM folder from Studio file structure.

There are some known limitations with this workaround:

- The Verify tool shows errors for all the rules under PRIORCUSTOM, if you ran it during the rules merge. After you merge all the rules and you delete the PRIORCUSTOM folder, this problem does not continue
- The upgrade process does not copy rules not touched by the customer in a prior base configuration over into the PRIORCUSTOM folder.
- In PolicyCenter 3.x, the upgrade tool copied the prior custom rules in the target folder and moved the new base configuration PolicyCenter 3.x rules into a new subdirectory. This is not a functional limitation, but is something to note.

## Archiving Not Supported

This release of PolicyCenter includes code related to archiving which is not supported at this time. Therefore, any code, objects, methods, or APIs related to archiving are subject to change without notice. Do not perform configuration related to archiving or proof-of-concept work using this release because this functionality is not officially supported.

## ContactManager Integration

This release of PolicyCenter provides an integration with Guidewire ContactManager. For more information, see the “Contact Management System Integration” topic in the *Application Guide* and the *Contact Management Guide*.

## Studio Product Model Improvements

This release includes a number of changes to improve the user interface for the product model in Studio. This topic describes some of these changes.

See the *Configuration Guide* for more information.

## Rearrange Fields and Tabs on the Policy Line Editor (PC-14609)

The following changes were made to the policy line editor:

- Rename the **Basics** tab to **Basics & Coverages**. The **Basics & Coverages** tab includes the following fields and sections:
  - **Code**
  - **Name**
  - **Description**
  - **Priority**
  - The contents of the **Coverages** tab
- Add a new **Advanced** tab. The **Advanced** tab includes the following fields and sections:
  - **Policy Line Subtype**
  - **Territory Code Needed**
  - **Reference Code**
  - **Initialization Script**
- On the **Coverages**, **Exclusions**, and **Conditions** tree views, the default **Group By** selection is **Category**.

## Rearrange Fields and Tabs on the Studio Product Editor (PC-14610)

The following changes were made to the product editor in Studio:

- The **Basics** tab includes the following fields and sections:
  - **Code**
  - **Name**
  - **Description**
  - **Priority**
  - **Abbreviation**
  - **Default Policy Term**
  - **Product Account Type**
  - **Offering Required**
  - **Policy Lines**
  - **Question Sets**
  - **Policy Terms**
- Added a new tab called **Advanced**. The **Advanced** tab includes the following fields and sections:
  - **Quote Rounding Level**
  - **Quote Rounding Mode**
  - **Days Until Quote Needed**
  - **Reference Code**
  - **Document Templates**
  - **Initialization Script**

## Line of Business Fatures Are Visible Only on Policy Lines Where They Apply (PC-14611)

Line of business features are now only visible on policy lines where they apply. For example:

- The **Coverage Symbol Groups** tab and **Coverage Symbol Group** drop-down list is only visible for the Commercial Auto policy line.
- The **Official IDs** tab is only visible for the Workers' Compensation policy line.
- The **Blanket Group Type** field is only visible for the Commercial Property policy line.

If you add a new policy line, you can enable or disable these tabs and fields from displaying in Studio.

## Advanced Tab for Clauses (PC-14794)

When any coverage, condition, or exclusion is selected in the tree, there is now an additional **Advanced** tab. The **Advanced** tab includes the following:

- Reference Date By
- Integration Fields
- Reference Code
- Covered Party Type
- Initialization script
- Removal script
- Existence script

The Existence drop-down list has been moved to the Basics tab.

## Advanced Tab for Coverage Terms, Options, and Packages (PC-14944)

There is now an **Advanced** tab for coverage terms that contains the following fields:

- Model Type
- Aggregation Model
- Restriction Model
- Reference Code

There is a new **Advanced** tab for options and packages that contains the following field:

- Reference Code

## Geocoding Using Bing and MapPoint (PL-16708)

Guidewire has added an implementation of the GeocodePlugin that connects to the Microsoft Bing Maps Geocode Service. The Bing Maps plugin implementation replaces the Microsoft MapPoint implementation.

Guidewire deprecates the MapPoint implementation with this release. Microsoft announced plans to retire the MapPoint web service effective November 18, 2011. If you currently use geocoding features and the MapPoint plugin, you must migrate from MapPoint to Bing Maps before November 18, 2011. Otherwise, geocoding features in the application cease to function on November 18, 2011, and afterwards.

## Rate Table Maintenance Improvements

This release includes changes and additions to Rate Table Maintenance. This topic describes some of these changes and additions.

### (PC-14648)

Rate books can now apply on different effective dates depending on whether the job is a new policy or a renewal policy.

1. The **Renewal Effective Date** of a new rate book is automatically set to the value of **Effective Date**.
2. The user can change the **Renewal Effective Date**.
3. After **Renewal Effective Date** has a value, PolicyCenter will not automatically change it. This is true even if user changes **Effective Date**.

### New Match Operator (PC-14650)

There is a new Rate Table Maintenance match operator which finds the row with the longest substring match to a string argument. For example, a table has a ZIP column where some rows have ZIP +4 while other rows just have ZIP. If you pass in ZIP+4, the match operator looks for a matching row in the table with ZIP+4. If ZIP+4 is not

found, the match operator looks for a row with the matching ZIP. If a matching ZIP is not found, the match operator finds the row with no ZIP.

### Value Provider Selection (PC-14772)

On the Rate Table Definition user interface, the Value Provider Type drop-down list had Custom Value Provider and Typelist Value Provider as options. These have been removed. The Custom Value Provider Type field has also been removed. Now there is a Value Provider field with the following items:

- Coverage Value provider
- Coverage Term Value Provider
- Coverage Term Opt Value Provider
- Termless Coverage Provider
- Reference Value Provider
- TypeList Value Provider

## Base PCF File Changes

All links below require the `ReleaseNotes_files` directory on your local disk.

### PolicyCenter release 7.0.1 to 7.0.2

- To view a report of the changes in the base PCF files in the `modules/pc` directory, [click here](#).
- To view a report of the changes in the base PCF files in the `modules/p1` directory, [click here](#).

## Base Resources Changes

All links below require the `ReleaseNotes_files` directory on your local disk.

### PolicyCenter release 7.0.1 to 7.0.2

- To view a report of the changes in the base resources in the `modules/bc` directory, [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 Resource Portal.

## 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 meant to be a comprehensive list.

ID	Description
<b>Account</b>	
PC-14797	Both the account and policy history screens now have a link in the job column of the event list view for Rewrite New Account jobs.
PC-14887	Corrected the javadoc for transferPolicies in AccountAPI.gs. This method uses the account number of from/to accounts and not the public id of accounts.
<b>Clustering</b>	
PL-7124	Guidewire now requires a unique server ID for each node in a cluster. The application verifies its server ID for uniqueness across all other cluster members as the node starts.
<b>Contacts</b>	
PC-14502	A subset of primary address data was denormalized onto the contact to improve query speed for some searches involving contacts and their addresses. The change includes logic to maintain the denorm fields, as well as an upgrade trigger.  This was previously a PolicyCenter 7.0.1 Known Issue.
PC-14719	When two or more contacts have linked addresses, and one of those addresses is updated in an external contact management system, the address is updated on both contacts in PolicyCenter as it should be. Previously, the address did not get updated in the contact management system on the second contact, leaving the contact inconsistent between the external contact management system and PolicyCenter.
PC-14763	Fixed an issue with updates to the contact management system. After calls from a contact management system to updateContact in PolicyCenter, updates from PolicyCenter to the contact management system were lost.  This was a known issue in PolicyCenter 7.0.1.
<b>Data Model</b>	
PC-14839	The database report for policy period now includes a row for the number of temporary branches that exist. You can use this to determine if there is something unexpected happening with temporary branches. This number should remain relatively small.
PL-17540	It is now possible to add an event to an entity through an extension, even if the base configuration definition of the entity contains one or more events.
<b>Database</b>	
PL-11896	Guidewire now bundles the 3.0 version of the Microsoft SQL Server JDBC driver with the application, replacing the 2.1 version. The file name has also changed from sqljdbc.jar to sqljdbc4.jar, to reflect the fact that this runs only with the 4.0 version of JDBC included with Java 1.6 or higher.  If you use the WebLogic application server and its bundled connection pool data source, then you need to manually copy the version of the JDBC driver bundled by Guidewire into WebLogic and configure its class-path to find it before creating your data source connection pool.
PL-15138	Guidewire now sets, through internal flags, the mode of both the application server and the application database. The mode is either production or development. It is now impossible for a development mode server to connect to a production mode database, or the reverse.
PL-16726	The Upgrade utility now supports always-execute AfterUpgradeVersionTriggers, in a similar manner to BeforeUpgradeVersionTriggers.
PL-17045	The upgrade framework now allows relational operation(<, >, <=, >= and =) with subquery. The following example illustrates the type of query that you can now execute during upgrade.
	<pre>UPDATE table   SET x = y  WHERE x &lt; (SELECT value               FROM table_other              WHERE value = 'something')</pre>

ID	Description
PL-17333	Fixed performance slowdown issues (that occurred under certain scenarios) by eliminating duplicate queries on the database catalog that occurred during population of table registry and while performing clean up work.
PL-17368	Database upgrade no longer runs certain statistics commands automatically. Database upgrade no longer includes the following: <ul style="list-style-type: none"> <li>Upgrade does not update statistics on indexes as both SQL Server and Oracle have options to automatically create statistics on indexes during database creation.</li> <li>Upgrade does not update statistics on tables and columns for new database.</li> </ul>
PL-17388	Modified the length of the columns created to support linguistic searching (the so-called DENORM columns) to reduce them in size from twice the original column's size to exactly the original column's size. PolicyCenter applies this change during the next database upgrade at application start-up.
<b>Financials</b>	
PC-13691	The user interface flow on the <b>Payment</b> screen has changed. When selecting the list bill billing method, the user must now select an alternate billing account before the list of installment plans is populated. PolicyCenter displays the installment plans and existing invoice options for the selected payer.
PC-14439	Removed all functions in PolicyCenter that update account billing settings, eliminated checking account and credit card fields from <code>BillingPaymentInfo</code> , and added a payment token field to <code>BillingPaymentInstrument</code> . Refactored the PolicyCenter plugins to reflect function name changes in the <code>BillingAPI</code> .  In the user interface, new payment instruments on the submission screen are now limited to credit card or ACH types.
PC-14676	Fixed a problem in the processing of financial transactions which sometimes resulted in a reversal and new onset rather than simply a partial offset transaction. This problem did not impact correctness.
PC-14758	The behavior of <code>Cost.OverrideAmount</code> and <code>Cost.OverrideTermAmount</code> has changed. In particular: <ul style="list-style-type: none"> <li><code>OverrideTermAmount</code> is no longer scaled if the term length changes.</li> <li><code>OverrideAmount</code> is scaled if the term length changes and is split and prorated if the rated object is split.</li> <li><code>OverrideAmount</code> on a cost is also rounded the same way the rating engine would round the standard amount.</li> </ul>
<b>Gosu</b>	
PL-11256	Gosu now throws <code>IllegalArgumentException</code> if you attempt to call <code>get/setFieldValue</code> using an <code>IPropertyInfo</code> that comes from a delegate.
PL-14507	Guidewire has modified <code>CoreIterableOfIterablesEnhancement.flatten</code> method so that it now returns an <code>Iterable</code> . Thus, many, though not all, operations previously available only to <code>Collection</code> objects are now available on all <code>Iterable</code> objects. In some cases, you must modify an existing usage of the <code>flatten</code> method by adding a following <code>toList</code> method to preserve the existing behavior. In certain rare cases, you must do this in order for the code to compile.
PL-15360	Fixed an issue in which Guidewire allowed <code>@WebServices</code> annotations at the function level in some cases. Now, you can use this annotation at the class level only.
PL-15806	Fixed an issue in which using the <code>==</code> operator in an overridden equals method caused a stack overflow.
PL-16840	Guidewire has removed <code>functionaljava-3.0</code> JAR from the application install files.
PL-17223	Guidewire has removed the following methods and properties on entity and delegate interfaces in Gosu.  On most entities: <ul style="list-style-type: none"> <li><code>entity.calculateNextVersion()</code></li> <li><code>entity.shouldTriggerValidation()</code></li> <li><code>entity.okToRetire</code></li> </ul> On Contact, Person, Company, and similar contact subtypes: <ul style="list-style-type: none"> <li><code>entity.OfficialIDBases</code></li> <li><code>entity.addToOfficialIDs( OfficialIDType type, State state )</code></li> </ul>
PL-17334	Class <code>gw.api.system.server.ServerUtil.Product</code> now returns a class in a public Guidewire API that follows the Guidewire standard. The new class, <code>gw.api.system.server.CWProduct</code> , provides the following new methods: <ul style="list-style-type: none"> <li><code>getProductCode</code></li> <li><code>getProductName</code></li> </ul>
PL-17987	Guidewire has removed the extraneous <code>Bundle</code> property on the <code>Bundle</code> object ( <code>Bundle.Bundle</code> ).

ID	Description
PL-7602	Enhanced GosuDoc to catch all throwables thrown when attempting to load a type. Previously, it only caught Runtime Exceptions.
<b>Integration</b>	
PC-14822	In the previous version, if ContactManager or another contact management system merged two contacts in the contact management system but only one contact was in PolicyCenter, PolicyCenter threw an exception. In this version, PolicyCenter updates the identifier (AddresBookUID) to that of the surviving contact if only the deleted contact was in PolicyCenter. If only the surviving contact was in PolicyCenter, it now does nothing.
PC-14841	Fixed a null pointer exception that occurred when trying to create a commercial auto claim using a retrieved policy.
PC-15008	The Paid status on the Account billing page is now populated.
PL-16370	The regen-soap-api command now generates RPC-Encoded and WSI artifacts into different directories.
PL-16779	Fixed an issue with regen-soap-api that caused it to fail if specific MQ libraries existed in configuration/plugins/shared/lib or gosu/lib. Executing regen-soap-api with these libraries now logs a warning rather than forcing an outright failure.
PL-17350	Fixed an issue with regen-soap-api that caused it to not properly generate the template/toolkit-javadoc/gw-pc-plugin directory.
PL-17479	The base configuration no longer contains a suite-config.xml file that contains default URL values of localhost:port/productname. Instead, this file is empty. Guidewire considers any product not configured in suite-config.xml (which is the default for all products) to be non-existent in the suite.
<b>Jobs</b>	
PC-14768	A conversion on renewal policy period has a null basedOn field and a renewal job subtype. Made several changes to the CancellationEnhancement Gosu class to account for conversion on renewal policy periods. Previously, all renewal jobs were assumed to have a basedOn field, and the null basedOn field cause an error.
PC-14821	Fixed an issue where PolicyCenter generated a high-priority activity with subject “Blocked-Pending Renewal...” when a batch renewal job for personal auto could not be completed. Now PolicyCenter is able to process the renewal job. This problem also occurred in businessowners renewals.
PC-14893	When PolicyCenter starts a rewrite job, PolicyCenter now creates a copy of all auto-numbered sequences rather than reusing the sequences already in the policy. This is consistent with the renewal job and results in elements being renumbered sequentially in a rewritten policy period.
PC-15017	PolicyCenter now handles auto-numbering in out-of-sequence jobs and is aware of elements in future slices. When auto-numbering entities that are part of the revision graph, there are newly rewritten methods that take an array that is being auto-numbered. These methods expect that array to contain elements from both current and future slices. In the default configuration, there are examples that show how to build this array and how to use the auto-numbered sequence. See the PolicyLocationEnhancement for auto-numbering buildings.
PC-15087	Previously after quoting in the side by side screen, coverage level cost data on the side by side screen did not always refresh properly, showing old values from a previous quote. This fix explicitly forces an update of the associated bundle after you click Quote All; all coverage level cost data is updated.
PC-15100	When copying base data between side by side periods, if an entity containing an array of children marked for cascade delete needed to be removed, the side by side base data copy sometimes failed. This issue has been fixed.
PC-15112	Previously, after quoting in the side by side screen, attempting to bind one of the side by side periods failed due to a concurrent data exception. The resolution to this issue exposes a method that allows the wizard’s bundle to be explicitly refreshed. This method is called immediately after quoting in the side by side screen. If your code uses the parallel quote method in the context of a job wizard, you need to call this method to ensure that beans in the wizard’s bundle are refreshed. This method is JobWizardHelper.refreshBundle.
PC-15257	Fixed an issue with personal auto that resulted in an exception being thrown while quoting. The problems was that there was no longer any location marked as primary on the policy after changing the garaging locations of vehicles. Now, there is always a primary location.
<b>Performance</b>	
PC-14744	Improved performance with Gosu fragments that occur multiple times, such as in availability logic.
PC-15052	Denormalized some of the address fields on the contact and address tables to improve performance of Account queries. This improves performance on the Search Account pages in PolicyCenter.
<b>Policy File</b>	

<b>ID</b>	<b>Description</b>
PC-14765	Fixed an issue where PolicyCenter threw a <code>NullPointerException</code> when the user clicked on the informational text: "This policy is being rewritten to a new account" in the <b>Account File History</b> screen on a policy rewritten to a new account.
<b>Product Model</b>	
PC-14445	Studio verification of resources and server start-up verification will now catch differences between the system table data stored in the XML files and the corresponding entity definition.
PC-14646	Studio now generates a validation error if a required field in a system table is not set. A required field is part of the database index or a non-nullable field without a default value.
<b>Profiling</b>	
PL-17533	Fixed an issue with the <b>Server Tools → Web Profiler</b> in which the <b>Group Frames</b> result page did not show detail information. The page now shows the detail information.
<b>Rating</b>	
PC-13807	For a rate table factor, now Date and Boolean type are restored, users should be able to select Date or Boolean type for the rate table factor column.
PC-13945	PolicyCenter now raises a warning when importing a table if a referenced typelist in the import table is not present in the system being imported into. If a typelist actively being used by a rating table on the same system is removed, PolicyCenter does not raise a warning. The existing values are maintained and not removed.
PC-14127	Previously, when importing a rate table from Excel, PolicyCenter displayed an exception when importing a non-Excel file. PolicyCenter now displays a message.
PC-14492	Fixed an issue with slow performance in Rate Table Maintenance. Improved performance of loading and updating a rate table with more than 3000 rows.
PC-14638	Improved verification in the Rate Table Match Op Definition system table. It is no longer possible to create entries that fail the following validation criteria: 1. The entry must have a unique OpCode (no duplicates allowed) 2. The entry has a valid class name for <code>Impl Class</code> 3. The <code>Impl Class</code> provided extends <code>gw.rating.rtm.matchop.MatchOperationFactory</code>
PC-14657	Fixed an issue with importing rate table information from Excel. If the factor values in the rate table exceeded four digits (factor uses data type double), on import the factors got truncated to four digits. For example: <ul style="list-style-type: none"> <li>• 11.12345 was imported as 11.12</li> <li>• -1.12345 was 1.123</li> </ul> <p>This was a Known Issue in PolicyCenter 7.0.1.</p>
PC-14687	Fixed an issue where premiums did not appear for all versions in side by side submission view when an offering was not selected and the submission is quoted.
PC-14716	Fixed an issue with range overlaps not being detected in rating.  For example, the following table has a Min/Max param pair for vehicle age (max inclusive), a Jurisdiction param (exact matchop), and one Factor column: <pre>Min, Max, Jurisdiction, Fctr 0,2,null,100 2,4,null,101 // Overlaps previous row</pre> In this case, PolicyCenter detected the range overlap.  However, when another row with a different jurisdiction is added between the rows, PolicyCenter did not detect the overlap: <pre>Min, Max, Jurisdiction, Fctr 0,2,null,100 0,2, CA,105 2,4,null,101 // Overlaps row 1</pre> This issue has been fixed for both max inclusive and max exclusive operators.
PC-14755	Previously, any user could import a rate book. Now, only users with the correct permissions can import a rate book.
PC-15089	Previously when validating contents of a rating table with multiple range parameters, range overlap was detected erroneously and inconsistently. In this release, the validation logic was changed.

ID	Description
PC-15093	The class gw.rating.rtm.matchop.MatchOp is available in Studio. You can use this class to implement match operators. Existing match operators are read-only. You can use these as a model to create your own.
PC-15129	Improved the performance of XML export of a rate book containing 80 tables in Rate Table Maintenance.
<b>Reinsurance</b>	
PC-13936	<p>Max ceding was changed to reflect the amount of ceded risk that can be ceded to an attachment when taking into account possible overlaps with other agreements and the total risk. In the default configuration, the ceded risk will always be the same as the max ceding unless you override the value. You can do this with surplus treaties in the reinsurance plugin.</p> <p>For net excess of loss treaties and facultative net excess of loss treaties with “apply to gross retention” set to true, the amount ceded is limited to the layer below the gross retention that is not ceded to a proportional treaty.</p>
PC-14468	When there is an overlap that causes a split in layers, multiple agreements with the same agreement number will be returned. The information will be the same for these duplicate agreements except for the attachment point, coverage limit, and recovery limit. Those values will represent the values for that reinsurance layer.
PC-14989	Reinsurance is now correctly disabled when a reinsurance parameter does not enable it.
<b>Reporting</b>	
PL-16825	<p>Added the ability to track performance of the LDAP cache refresh if using Cognos integration for reporting.</p> <p>Added output to the system console (if LDAP debugging is turned on) that informs the user what percentage of the cache refresh timer the current cache refresh is taking.</p>
<b>Revisioning</b>	
PC-7406	Fixed an issue in the revisioning algorithm which resulted in an entity with the autoSplit=false attribute being split anyway during merge in out-of-sequence or preemption scenarios.
PL-16683	Fixed an issue in which out-of-sequence policy changes that affected localized columns caused data inconsistencies for non-default locales. This modification changes how PolicyCenter handles localized columns in policy comparison conflict reports. A conflict difference now reports changes in non-default locales as property changes.
<b>Studio</b>	
PC-14829	For type key modifiers, PolicyCenter now stores the type key code. In PolicyCenter 7.0.1, PolicyCenter stored the type key display name. This problem appeared in PolicyCenter 7.0.1.
PL-13845	Guidewire has modified the PCF New Folder dialog so that it is no longer possible to create a folder that differs in capitalization only with an existing folder.
<b>UI Configuration</b>	
PC-14849	Fixed an issue that caused PolicyCenter to throw a null pointer exception when the user clicked on the message, “This policy term is pending payment confirmation”.
<b>Upgrade</b>	
PC-14390	Added a database upgrade trigger that initializes CPB1anketAutoNumberSeq on existing Commercial Property lines. The trigger creates a sequence for each new term. This is consistent with new behavior where Rewrite jobs now clone auto-numbered sequences. However, if a Rewrite was started prior to upgrade, it will share its autonumber sequence with the previous term.
PC-14616	The configuration update tool now sets the description value for jurisdiction and state type list extension files.
<b>User Tools</b>	
PC-12637	Previously, participants were editable even when the user did not have permission to edit the job. Now, participants are not editable when the user does not have permission to edit the job.
PC-14603	The Gosu class, PolicyPeriodReinsuranceEnhancement.gsx, is now editable.
PC-14627	On the AdminProducerCodeSearch screen, the producer variable was not being recalculated and was producing an IllegalStateException when the page screen was reset. This change sets <code>recalculateOnRefresh="true"</code> for the producer variable.

ID	Description
PC-14633	Moved the populateSimpleValues enhancement method from gw.xml.XmlTypeInstance to the gw.webservice.pc.pc700.gxmodel.SimpleValuePopulator Utility class. You should change usages of the enhancement method to use the utility method instead. For example, a call that looked like:  this.populateSimpleValues(account) becomes:  SimpleValuePopulator.populate(this, account)
PC-14813	If the ConcurrentDataChangeException occurs while processing a work item, PolicyCenter logs the error only if the number of attempts exceeds the configured value of the WorkItemRetryLimit.
PC-14892	When PolicyCenter creates an account activity, PolicyCenter set the priority to the priority of the activity pattern. Previously, PolicyCenter always set the priority to high.
<b>Web</b>	
PL-16391	Guidewire has updated web.xml to reflect support for the Java Servlet 2.4 specification.
PL-16853	Modified the @Servlet annotation to work properly without requiring manual intervention in the configuration.
PL-16986	Guidewire has updated the embedded Apache Xerces library from version 2.9 to version 2.10.
PL-17115	Modified the Studio RPC-Encoded Web Services editor to show User Name and Password fields.
PL-17170	@WsiWebService now returns an HTTP 400 Bad Request in the event of receiving invalid XML. Previously, it would return an HTTP 500 Internal Server Error.
PL-17248	An exception declared on any method of a WsiWebService now behaves as if that exception were declared on every method of the WsiWebService at service invocation time. The WSDL generation is not affected. The visible effect of this change is that a previously undeclared exception would result in a lack of a <detail> child. Now, the system generates a <detail> child. On the Gosu client side, this means that instead of receiving a Wsd1Fault, the client now receives a more specific subtype of a Wsd1Fault.  The endpoint address for calling WsiWebServices using SOAP 1.1 has changed. SOAP 1.1 and SOAP 1.2 now uses distinct URLs and auto-detection of SOAP version no longer occurs. The address change is reflected in the associated WSDLs. Any client using SOAP 1.1 to talk to a WsiWebService will need to use the new WSDL.
PL-17412	Guidewire now provides a new class, WsiRequestLocal, for use with webservice request-based data storage. You can use this class to communicate values between various parts of a webservice implementation in the scope of a single request.
PL-17568	Fixed an XML parse exception (gw.xml.XmlException) that occurred if you invoked a web service that used a schema with multiple imports
<b>Workflow</b>	
PL-16885	Modified the Purge Workflow batch process so that it correctly handles custom extensions.
<b>XML</b>	
PL-17275	The Gosu tester in Studio now properly handles control characters in program output when connected to a running server.

## Known Issues and Limitations

This section describes known issues with this release of Guidewire PolicyCenter.

- PolicyCenter Known Issues
- Studio/Platform Issues

**Note:** For maintenance releases, Guidewire often defers fixing configuration issues if they 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

### Studio: “Verify Changed Resources” is not implemented in the product model (PC-6928)

**Issue** – The Studio Tools → Verify → Verify Changed Resources command does not work with the PolicyCenter product model.

**Workaround** – Use one of the other verification commands instead. See “Validating Studio Resources” in the *PolicyCenter Configuration Guide* for details.

### 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.

### 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. 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 original answer is still saved along with the other answers.

**Workaround** – The answers to these *irrelevant* questions 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)
        }
    }
}
```

### 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 screen of the agent
- 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.

### Leap Year Override Proration (PC-12893)

**Issue** – Leap year days are counted as part of the policy period. This affects the policy period premium amounts, since the calculated amount is assumed to be 365 instead of 366 days.

**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 can occur with fields that are not base data.

**Workaround** – Add code that copies this type of data to the other versions.

### Rate Table Maintenance: Rate book detail localization table visible when there is only one language configured; it shouldn't be (PC-13799)

**Issue** – Localization tables are made available for features when Policy Center is configured to support multiple locales. However, the **Rate Book Detail** view displays the localization table when there is only one locale supported by the Policy Center application.

**Status** – Guidewire is aware of this issue.

### Rate Table Maintenance: Rate table definition Usage tab not refreshed properly on cancel edit to reflect changes to usage (PC-13868)

**Issue** – When accessing the rate table definition detail page for a particular definition, the related information is loaded in the view. For performance, PolicyCenter stores the related information just in case the user wishes to cancel changes. If you leave a rate table definition in edit mode, and rate books are updated to include or exclude this definition within this time, cancelling the rate table definition edit will *not* refresh the **Usage** tab to reflect rate book changes. Cancelling the update restores the rate table definition to the original values initially loaded when you accessed this page.

**Workaround** – If you know changes have been made to the usage of your rate table definition, cancel the edit. Then return to the rate table definition search and re-select the rate table definition. The **Usage** tab will be reloaded with the latest changes. In general, do not leave rate table definitions in edit mode for a long period of time.

### Rate table maintenance: Import rate book raises error rate table exists on first import, passes on second attempt (inconsistently) (PC-14134)

**Issue** – In some cases when importing rate books, PolicyCenter raises an import error indicating that a non-existent rate table exists in the target environment. The error is incorrect, and import should be successful in this scenario.

**Workaround** – Import the rate book a second time. The error is not reported and the import is successful on the second attempt.

### Contact.AccountHolderCount incorrectly counts the number of accounts (PC-14736)

**Issue** – `Contact.AccountHolderCount` incorrectly counts the number of accounts a particular contact is the account holder on.

**Workaround** – Query the database to count the accounts on which this contact has the account holder role.

### [Contacts created from the Contact tab are not autosaved \(PC-14761\)](#)

**Issue** – Contacts created from the **Contacts** tab are not getting autosaved. The **UnsavedWork** light correctly goes on, but when the user clicks elsewhere, the work is lost.

**Workaround** – Set the `autosaveable` attribute on the `NewContact.pcf` to `true`.

### [StopGap typecode incorrect removed rather than retired \(PC-14968\)](#)

**Issue** – In PolicyCenter 7.0.1, the typecode value `AllMonopolyStates` was removed from the `StopGap` typelist and replaced with `AllMonopolisticStates`. This change prevented a database upgrade from succeeding if there was any data already referencing that typecode value. Rather than removing `AllMonopolyStates`, that typecode should have been retired.

**Workaround** – If you have data referencing `AllMonopolyStates`:

- Add the `AllMonopolyStates` typecode back as an extension in your configuration.
- Retire the `AllMonopolisticStates` typecode.

### [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.

### [The `isOOSJob` method returns incorrect results \(PC-15119\)](#)

**Issue** – The `isOOSJob` method of a job returns incorrect results after the job is completed. It is correct while the job is open, but after a job is completed, it returns `true` if there are any jobs with a later effective date, regardless of when they were completed.

**Workaround** – The `isOOSJob` method is not available in Studio. If you need to use this method, you must write your own method that correctly assesses this, by determining if there are any jobs with a later effective date than this one but at an earlier completion date.

### [Matching row is not the same after import and export of rate table \(PC-15259\)](#)

**Issue** – When no row matches a rate table search, the query returns different values depending on whether the table is in memory or in the database. In both cases, PolicyCenter should return an error but does not.

**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.

**Workaround** – Guidewire is aware of this issue.

## BillingCenter integration: Removing Alt 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, a WsdlFault is thrown if the Alt Billing Account gets removed.

This is an issue when integrating with BillingCenter 7.0.0.

**Workaround** – Guidewire is aware of this issue.

## BillingCenter integration: New Invoice Streams Are Not Created When Issuing Policy With Alt 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.

**Workaround** – Guidewire is aware of this issue.

## Studio/Platform Issues

### Issues with Internet Explorer 9

**Issue** – If you are using the Internet Explorer 9 browser, it is possible to see issues such as screen flickering or an incorrect tab order for fields. According to public reports, IE 9 exhibits these and other issues with a variety of web sites and web applications.

**Workaround** – Because this is the behavior of the Internet Explorer 9 rendering engine, Guidewire cannot address these issues. However, there are reports of IE 9 users being able to reduce these issues by changing the new IE 9 Accelerated Graphics settings on the Advanced tab of the **Internet Options** dialog.

### First time you click on the arrow of the typekey input, the drop-down menu will not open (PL-10134)

**Issue** – The drop-down menu does not open on the first click of the arrow on a typekey input. Instead, the help text opens.

**Workaround** – Turn off help text on focus by setting `InputHelpTextOnFocus` to `false` in the `config.xml` file. By doing that, the help text shows only if you mouse over the input and will not interfere with opening a drop down menu.

### XML API upgrade feature missing from documentation (PL-10257)

**Issue** – The *Integration Guide* describes a new set of XML APIs based on the `XmLElement` class. (Legacy APIs are based on the `XMLNode` class.) You can continue to use the legacy APIs. However, the *Integration Guide* omits mentioning an additional upgrade-specific feature.

**Workaround** – For backwards compatibility only, you can import an XML schema into the Gosu type system using the legacy XML system by following these instructions:

1. Copy:

`PolicyCenter/modules/pl/config/registry/compatibility-xsd.xml`

To:

`PolicyCenter/modules/configuration/config/registry/compatibility-xsd.xml`

2. Add an entry for your schema. Set the value of the `namespace` attribute to the Gosu package name of the schema. For example, if the schema is in the package location `my.package` and is called `myschema.xsd`, set the value of `namespace` to `my.package.myschema`.

### 'ListDetailPanel' throws exception (PL-10316)

**Issue** – It is possible for PolicyCenter to throw an exception if the user cancels out of a `ListDetailPanel` widget if `StartInEditMode` is also `True`.

**Workaround** – Set `StartInEditMode` to `False` for the screen that contains the `ListDetailPanel`. As a consequence, the user must click `Edit` to modify that screen.

### Countries configured in 'zone-config.xml' still generate a warning during 'regen-dictionary' even when zone data is loaded for all of these countries (PL-11947)

**Issue** – Countries configured in `zone-config.xml` still generate a warning during regen-dictionary even when zone data is loaded for all of these countries.

**Workaround** – Warning message is created in error and can safely be ignored.

### There is a length limitation on entity localization table names (PL-13360)

**Issue** – There is a length limitation on entity localization table names.

**Workaround** – Ensure that localization `tableName` property specified in the entity extension file is less than 16 characters. The error message generated if the localization table name exceeds the maximum length indicates that 18 characters are allowed, but that does not account for two additional characters added by the application.

### ORDER BY ASCENDING different for Oracle, SQL Server (PL-13559)

**Issue** – In SQL Server, `ORDER BY ASCENDING` sorts NULL values first. In Oracle, by default, `ORDER BY ASCENDING` sorts NULL values last. This inconsistency is known to cause problems in PolicyCenter with the `NotificationPlugin`.

**Workaround** – Guidewire is aware of this issue.

### GX models that reference virtual fields and enhancements throw null pointers if null (PL-13560)

**Issue** – The GX models that reference virtual fields and enhancements throw null pointers when they are null.

**Workaround** – Ensure that null checks and error handling is included so that if referenced virtual fields or enhancements are null, then there will not be a null pointer exception.

### Sending email with file attachment with unicode filename is not correctly handed over to the mail server (PL-13582)

**Issue** – Sending email with file attachment with unicode filename is not sent to the mail server correctly.

**Workaround** – Use Latin characters for file names on attached files.

### 'JavaToolkit.gs' has incorrectly hard coded memory, which results in failed 'regen-java-api' Ant task (PL-13663)

**Issue** – The `JavaToolkit.gs` has hard coded memory which can result in failed `regen-java-api` Ant tasks.

**Workaround** – Increase the size of the maximum heap setting on line 161 of `JavaToolkit.gs` in the Ant module. The default value is 512.

### Problem with 'regen-java-api' command and JAR files (PL-16351)

**Issue** – If you run the `PolicyCenter/bin/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 resulting 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*

into the following directory:

*PolicyCenter/java-api/lib*

### [Client reflection not supported for ModifierInput \(PL-16583\)](#)

**Issue** – The `ModifierInput` widget does not support client reflection.

**Workaround** – Do not attempt to use client reflection functionality with the `ModifierInput` widget. Guidewire is aware of this issue.

### [Bing Geocode plugin implementation not visible in Studio \(PL-17675\)](#)

**Issue** – The Bing Geocode plugin implementation (`BingMapsPlugin.gs`) is not visible in Studio.

**Workaround** – To access the Gosu source code for this plugin, access it from the file system within your Guidewire application directory at the following location:

*...modules/p1/gsrc/gw/plugin/geocode/impl*

---

**WARNING** Do not modify the file in this folder as that invalidates the application checksum. To modify `BingMapsPlugin.gs`, first copy it to *.../configuration/gsrc/gw/plugin/geocode/impl*.

---

# Guidewire PolicyCenter 7.0.3 Release Notes

## Release 7.0.3

**Document Revision:** 20-May-2014

Copyright © 2001-2014 Guidewire Software, Inc. All rights reserved.

Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Reinsurance Management, Guidewire ContactManager, Guidewire Vendor Data Management, Guidewire Client Data Management, Guidewire Rating Management, Guidewire InsuranceSuite, Guidewire ContactCenter, Guidewire Studio, Guidewire Product Designer, Guidewire Live, Guidewire DataHub, Guidewire InfoCenter, Guidewire Standard Reporting, Guidewire ExampleCenter, Guidewire Account Manager Portal, Guidewire Claim Portal, Guidewire Policyholder Portal, ClaimCenter, BillingCenter, PolicyCenter, InsuranceSuite, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks, service marks, or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries.

All other trademarks are the property of their respective owners.

**This material is confidential and proprietary to Guidewire and subject to the confidentiality terms in the applicable license agreement and/or separate nondisclosure agreement.**

Guidewire products are protected by one or more United States patents.

## Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Support

- Issues and Major Changes
- Improvements and General Issues
- Known Issues and Limitations

## Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

---

**IMPORTANT** If you missed 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 7.0.3.

- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 7.0.1 or later. For more information, see “Integrating with BillingCenter” on page 325.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 6.0.0 or later.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 7.0.0.

## 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 Resource Portal – <http://guidewire.custhelp.com>

## Issues and Major Changes

This section contains issues and major changes that may affect your installation.

- “Integrating with BillingCenter” on page 325
- “Archiving in PolicyCenter” on page 325
- “Changes to Search Requirements” on page 325
- “Policy Location Search API” on page 326
- “Studio Product Model Improvements” on page 326
- “Guidewire Standard Reporting” on page 326
- “Base PCF File Changes” on page 328
- “Base Resources Changes” on page 328
- “Changes in This Release Provided in Upgrade Diff Report” on page 328

## Integrating with BillingCenter

This version of PolicyCenter requires BillingCenter 7.0.1 or higher, which has not yet been released.

If you are integrating with BillingCenter, Guidewire recommends that you either:

- Upgrade to PolicyCenter 7.0.3 when BillingCenter 7.0.1 is released.
- Upgrade to PolicyCenter 7.0.3 before BillingCenter 7.0.1 is available, but wait until you have BillingCenter 7.0.1 before enabling and customizing the BillingCenter integration.

## Archiving in PolicyCenter

*Archiving* is the process of moving the data associated with a policy from the active PolicyCenter database to a document storage area. New in this version of PolicyCenter, you can archive a policy term. In turn, PolicyCenter archives the data associated with all policy periods in that policy term. You can search for and restore archived policy terms. While archived, the data associated with the policy term occupies less space in the active database.

Once archiving is enabled on a given database, it may not be disabled, although it may be configured that no further archiving or policy terms is performed.

Enabling archiving may have a significant impact on your PolicyCenter configuration and integration. Archived policy data is no longer immediately available in the active database. Your implementation of archiving may require data model changes.

If you are working on a PolicyCenter implementation that may implement archiving in the future, Guidewire suggests that you:

- Evaluate your data model against the requirements for archiving.
- Understand the implications of archived data on PCF files, code, and database searches.

See the *Application Guide* for more information and links to other documentation.

## Changes to Search Requirements

The following changes were made to search requirements.

### Minimum Search Requirements

To improve system performance, changes were made to the minimal search requirements on the following search screens:

- **Search Policies**
- **Search Accounts**
- **Search Contacts**

You can search on the following fields without specifying additional search information:

- **Official ID**
- **Account Number**
- **Policy Number**
- **Phone**

### Personal Names

Both the first and last name fields have a check box to indicate whether the name must be an exact match.

The name of the person requires the following:

- First and last name are both required.

- For the first or last name, the name must be an exact match. If the exact match box is not checked, you must provide the first three letters of the name.
- If the last name is not an exact match, you must provide either city and state or postal code.

### Company Names

There is now a check box to specify whether the company name is an exact match.

The name of the company requires the following:

- The name must be an exact match, or you must provide the first five letters of the name.

### Producer Code

You can enter a producer code without specifying additional search information.

## Phone Number Search Requirements

Most searches based on phone number have been changed to an exact match to improve performance. The extension for a phone number is stored in the same field as the phone number itself. Therefore, a search omitting an extension, if one is present, no longer matches in the default search configuration.

## Policy Location Search API

New in this version is a policy location search API to retrieve a list of policy location summaries within a rectangular geographic bounding box.

Use the `PolicyLocationSearchAPI` web service to retrieve summary information about policy locations with a rectangular geographic bounding box. This web service is WS-I compliant.

PolicyCenter provides this web service for integration with ClaimCenter. ClaimCenter uses the returned policy location information to plot policy locations within the areas of interest on claim catastrophe maps.

This API is intended to be called during periods of low system load.

For more information, see the *Integration Guide*.

## Studio Product Model Improvements

This release includes a change to improve the user interface for the product model in Studio. This topic describes this change.

See the *Configuration Guide* for more information.

## Offerings Tab for Question Sets (PC-14795)

In this release, there is now an **Offerings** tab for question sets to specify which offerings include the question set, and whether the question set is automatically disabled in new offerings.

See the *Configuration Guide* for more information.

## Guidewire Standard Reporting

---

**IMPORTANT** Perform the steps outlined in this release note before you start the installation and integration procedures listed in the *PolicyCenter Reporting Guide*.

---

In addition to the steps outlined in “Installing Guidewire Standard Reporting” (available in the *PolicyCenter 7.0.3 Reporting Guide*), perform the following steps before you perform the procedures listed in that guide:

- Step 1: Update and Modify Class ‘CognosPlugin.gs’
- Step 2: Update and Modify ‘CognosReportPage.pcf’
- Step 4: Update and Modify Plugin CognosPlugin
- Step 4: Restart the Application Server

#### Step 1: Update and Modify Class ‘CognosPlugin.gs’

You must modify Gosu file CognosPlugin.gs.

1. Copy the following file:

```
...\\modules\\pl\\gsrc\\gw\\plugin\\cognos\\CognosPlugin.gs
```

to the following directory:

```
...\\modules\\configuration\\gsrc\\gw\\plugin\\cognos
```

---

**IMPORTANT** Do not modify this file in its original location. If you do so, you invalidate your PolicyCenter installation.

---

2. Verify that the file is writable. If not, you need to make it writable.

3. Add the @Exportable annotation directly above the CognosPlugin class definition, for example:

```
@Exportable  
@Availability(AvailabilityLevel.MULTIUSER)  
@Distributed  
class CognosPlugin implements IStartablePlugin, InitializablePlugin {  
    ...
```

4. Add the following variable declarations in the CognosPlugin class

```
private static var _gatewayEndPointUrl : String  
private static var _dispatcherEndPointUrl : String
```

5. Do the following:

- a. Remove the getHostNameAndPort method and replace with the following methods:

```
public static function getGatewayEndPointUrl() : String {  
    return _gatewayEndPointUrl  
}  
  
public static function getDispatcherEndPointUrl() : String {  
    return _dispatcherEndPointUrl  
}
```

- b. Add the following statements in the start method’s second if statement, which starts with the code if (\_state != StartablePluginState.Started):

```
_gatewayEndPointUrl = _params.get("gatewayEndPointUrl") as String  
_dispatcherEndPointUrl = _params.get("dispatcherEndPointUrl") as String
```

#### Step 2: Update and Modify ‘CognosReportPage.pcf’

You must modify PCF file CognosReportPage.pcf

1. Open PolicyCenter Studio.

2. Find and open CognosReportPage.pcf for editing.

3. Select the TemplatePanel widget in the PCF page. This action opens the Properties pane at the bottom of the screen.

4. Copy the following <iframe> code and replace the existing text string in the value field:

```
<iframe id="cognosFrame" src="<%printContent(gw.plugin.cognos.CognosPlugin.getSetCognosCookiesUrl(),  
false)%>?gwGatewayEndPointUrl=<%printContent(gw.plugin.cognos.CognosPlugin.getGatewayEndPointUrl(),  
false)%>&gwDispatcherEndPointUrl=<%printContent(gw.plugin.cognos.CognosPlugin.getDispatcherEndPoint  
Url(),
```

```
false)%>&gwNamespace=<%printContent(com.guidewire.pl.system.dependency.PLDependencies.getPluginConfig().getPluginDef("CognosPlugin").getParams().get("namespace"), false)%>&gwUser=<%printContent(com.guidewire.pl.system.dependency.PLDependencies.getCommonDependencies().getCurrentUser().getCredential().getUserName().toLowerCase(), false)%>&gwPassword=<%printContent(java.net.URLEncoder.encode(com.guidewire.pl.system.dependency.PLDependencies.getCommonDependencies().getCurrentUser().getCredential().getPassword()), false)%>" style="width:1400px;height:1050px"></iframe>
```

## 5. Restart Studio.

### Step 4: Update and Modify Plugin CognosPlugin

You must add additional plugin parameters to CognosPlugin.

#### 1. Open the following plugin:

Plugins → gw → api → startable → IStartablePlugin → CognosPlugin

#### 2. Add the following parameters to the plugin:

Name	Example value
dispatcherEndPointUrl	http://hostname1:9300/p2pd/servlet/dispatch
gatewayEndPointUrl	http://hostname1:80/ibmcognos/cgi-bin/cognos.cgi

**Note:** You must set additional plugin parameters to complete the plugin configuration. See the *PolicyCenter 8.0.2 Reporting Guide* for details. See especially, “Step 4: Configure the Cognos Plugin”.

### Step 4: Restart the Application Server

Restart the application server. It is also useful to restart Studio before you start the application server as this often highlights errors with copy/paste or typing errors.

After you restart the application server, continue with the installation instructions outlined in the *PolicyCenter Reporting Guide*.

## Base PCF File Changes

All links below require the ReleaseNotes\_files directory on your local disk.

### PolicyCenter release 7.0.2 to 7.0.3

- To view a report of the changes in the base PCF files in the modules/pc directory, [click here](#).
- To view a report of the changes in the base PCF files in the modules/p1 directory, [click here](#).

## Base Resources Changes

All links below require the ReleaseNotes\_files directory on your local disk.

### PolicyCenter release 7.0.2 to 7.0.3

- To view a report of the changes in the base resources in the modules/bc directory, [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 Resource Portal.

## 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 meant to be a comprehensive list.

ID	Description
<b>Application Integration</b>	
PL-17945	Guidewire has changed the type of the URL field in the suite-config XSD definition file from xs:anyURI to xs:string.
<b>Assignment</b>	
PL-13175	Delegated the UserRoleAssignment entity out to a UserRoleAssignmentDelegate and removed the UserRoleAssignment entity at the platform level. All Guidewire applications now own their own UserRoleAssignment entity and use entity delegation to UserRoleAssignmentDelegate.
PL-18196	Fixed assignGroupByRoundRobin to manage GroupAssignmentState with proper set of keys including intended GroupType to assign a group properly.
PL-18397	Fixed an issue that occurred during the creation of an exposure at the end of FNOL wizard if you were using assignGroupByRoundRobin assignment.
<b>Batch Processes</b>	
PL-13912	Guidewire has added the ability to set an env attribute on the <ProcessSchedule> element in scheduler-config.xml to specify the schedule to run in a certain environment. As a consequence, you can now have different results for batch processing based on environment. By default, Guidewire does not set the env attribute.
PL-15657	Added Last Run Status column to the Processes table of the Batch Process Info page. This column allows a user to see if the last run of this batch task completed successfully or failed.  The possible values are: <ul style="list-style-type: none"><li>• Completed – The last run has completed successfully.</li><li>• Not available – This task has not ran yet or running right now.</li><li>• Failed/Interrupted –The last run has failed or was interrupted. (See the history of the process for more details.)</li></ul>
<b>Clustering</b>	
PL-10224	In addition to improvements to JGroups implementation of UNICAST protocol, this fix also introduced a new merge protocol, MERGE4. This is not set in the default protocol stack. (MERGEFAST is set.)
PL-17295	The valid range for multicast address is between 224.0.0.0 and 239.255.255.255. This change verifies the validity of the multicast address before attempting to join the cluster. If the address is invalid, PolicyCenter throws the following exception:  "Invalid multicast address " + multicastAddress + ". Valid range is between 224.0.0.0 and 239.255.255.255."
PL-17297	Guidewire has upgraded JGroups to the latest version, which is 2.12.1.
PL-17903	Guidewire now requires a unique server ID for each node in a cluster. The application verifies its server ID for uniqueness across all other cluster members as the node starts.
<b>Code Utilities</b>	
PL-18054	If you have used the \${x} substitution in document templates, then you need to revert those uses back to the <%= ... %> syntax.
<b>Cognos Integration</b>	
PL-18122	Modified the Guidewire reporting LDAP server to perform a case-insensitive compare of the user name in the Cognos LDAP authentication request to the users in PolicyCenter database. (Guidewire normalizes user names to all lower case in its communications with Cognos.)  The issue still exists that Cognos is not able to distinguish between Guidewire users that differ only in case even if these users are distinguished on Guidewire. Do not create Guidewire LDAP users that differ only in case.
<b>Configuration Upgrade</b>	
PL-18021	Fixed an issue with how the Upgrade tool automatically handled rule upgrades.
<b>Data Model</b>	

ID	Description
PC-12086	Provided expanded descriptions for the PolicyPeriodStatus typelist.
PC-12285	The UserRoleAssignment entity now implements a new UserRoleAssignmentDelegate.
PC-14890	The description of the EffectiveDate and ExpirationDate fields on the PolicyLine entity have been clarified in the <i>Data Dictionary</i> .
PC-15361	PolicyCenter denorm columns have been changed to take advantage of SearchColumns. The SearchColumns are a new implementation of denorm columns that are used explicitly for linguistic searches. The SearchColumn values are not stored in memory; therefore, they return null if accessed by means other than a query. To access an actual value in a SearchColumn, you must use the foreign key to the PrimaryAddress to the corresponding column. Note that this implementation removes the denorm columns City and PostalCode from Contact as they are no longer needed.
PC-15946	The CoverageSymbolGroup foreign key has been removed from the Coverage entity and replaced with an enhancement property. Before upgrading, this column should be nulled out in every Coverage table. A version check has been added to make sure these have been nulled out. This should only affect business auto coverages.
<b>Database</b>	
PL-18012	Previously, Guidewire prohibited you from connecting a development server to a production database, or connecting a production server to a development database. Guidewire now permits this kind of connection (merely generating a warning) if using the H2 database.
PL-18142	Fixed an issue in the Leap Year Days calculation that caused an infinite loop if times in the start and end of the DateRange did not match.
<b>Database Upgrade</b>	
PL-17732	Fixed performance issues with the upgrade process.
PL-17849	Guidewire has added a DateFromDatetime method to the BeforeUpgradeDBFunction utility class, which is available only in the context of upgrade queries. It is available only in restrictions, not in the query column list. Certain Guidewire applications need this for use in an upgrade trigger. In a BeforeUpgrade query, you can use this function in a SELECT statement, and it can take a nested function as an argument.
<b>Entities/Metadata</b>	
PL-10608	Guidewire now flags as an error in Studio—and as a warning at server start-up—if you attempt to set a non-queryable path on the Entity Path field in the Entity Names editor.
PL-17718	Fixed an issue in which using a forward slash (/) in a typecode definition prevented the application server from starting but Studio did not display an error flag for this condition.
PL-18071	Added support for denormalizing columns across tables to enhance the performance of search queries.
<b>Financials</b>	
PC-12893	In the default configuration, PolicyCenter now ignores leap days when prorating premiums for non-full-term policies. In previous releases, PolicyCenter would rate a half-term policy spanning a leap day differently than a half-term policy not spanning a leap day.  To preserve the original behavior, set the IgnoreLeapDaysForEffDatedCalc field to false in config.xml. This setting controls whether leap days are ignored in platform-level calculations when scaling effective-dated entities fields marked as scalable.  This was previously a known issue.
<b>Geocoding/Proximity Search</b>	
PL-17675	Fixed an issue in which the Bing Geocode plugin implementation was not visible in Studio.
PL-17886	Guidewire has deprecated the MapPoint implementation. Microsoft announced plans to retire the MapPoint web service effective November 18, 2011. If you currently use geocoding features and the MapPoint plugin, you must migrate from MapPoint to Bing Maps. Otherwise, geocoding features in the application will not function.
<b>Gosu</b>	
PL-16935	Fixed an issue in which PolicyCenter threw a Null Pointer Exception if you attempted to change Gosu code in Studio and it was connected to the application server. (The issue was specific to functions in code blocks that were called from other PCF functions.)

ID	Description
PL-17478	Modified the Gosu Tester parser so that it treats scriptability modifiers in the same way that Gosu classes and enhancements do. This means you should no longer receive the following error message while running code in the Gosu Tester:  'The property [or method], "XXXX", is not visible under the parser's visibility constraints.'
PL-17489	Fixed an issue in which certain corner cases of conditional variable declarations would fail with a Java byte-code VerifyError if the server was running in debug mode. This issue did not affect servers running in production mode.
PL-18033	Fixed an issue that caused incorrect behavior if you called reverse() on an implicitly-declared List in Gosu. (The issue did not occur with an explicitly declared List.) The issue was that the reverse sort on an implicitly-declared List occurred in-place, meaning that the sort occurred on the original list elements. The sort now occurs on a copy of the list, which is the desired behavior.
PL-18299	Added a preload mechanism to support pre-compilation of Gosu classes, as well as other primary classes in the system. The intent is to make the system more responsive the first time requests are made. To support this, Guidewire has added a Studio Other Resources → preload.txt file in which you can add a list of actions to take. The file contains static no-argument method calls, as well as the names of Gosu types to compile to byte-code or the Java types to load.  Guidewire also added a new logging category of Server.PreLoad that provides DEBUG level logging of all actions during server pre-loading of Gosu classes.
<b>GX Tools</b>	
PL-12128	Added the ability for a web service to use types from a different WSDL. For example, <pre>&lt;xs:import namespace="http://example.com/gw/api/test/TheirAPI"     schemaLocation="../../wsdl/local/gw/api/test/TheirAPI.wsdl" /&gt;</pre>
PL-18297	Restored missing eachException() methods on the Guidewire XML Modeler. Guidewire inadvertently removed these methods in a previous 7.x release.
<b>History</b>	
PL-17720	Guidewire no longer marks the HistoryType typelist as final. Therefore, it is now possible to add new values to this typelist. However, Guidewire made this change to allow the removal of the retrieved HistoryType in ClaimCenter. Guidewire recommends that you continue to add custom history types to the CustomHistoryType typelist, rather than the HistoryType typelist.
<b>Integration</b>	
PC-15581	The ClaimToPolicySystemNotificationAPI web service was added to this version of PolicyCenter. This enables integration with ClaimCenter 6.
PC-15680	Added a PolicyCenter 7 version of ProducerAPI, named ProducerAPIv7. However, this is an RPC and not WSI web service, therefore naming collision prevents the file from being named ProducerAPI. Given the naming collision and that WSI and not RPC is the standard for PolicyCenter 7 web services, this is a deprecated API. It will not be supported in the long-term. It does not contain any changes from the PolicyCenter 4 version of ProducerAPI.
<b>Internationalization</b>	
PC-15035	Replaced hard-coded strings with display keys on the Inference tab in Administration → Policy Form Patterns.
<b>Jobs</b>	
PC-7754	Validation warnings do not prevent manual quoting or issuance of a policy; therefore, they should not prevent quoting or issuance of an automatic renewal job. In the default configuration, PolicyCenter no longer raises a validation exception if validation warnings occur during an automatic renewal job. Automated renewals can now proceed without human intervention if validation warnings are present. Other job types are unaffected.
PC-15119	Fixed an issue where the is00SJob method of a job returned incorrect results once the job is completed. This was a known issue in PolicyCenter 7.0.2.
PC-15431	In the default configuration, Personal Auto submissions using the Side-by-Side feature now correctly display three offerings for comparison: Basic, Standard, and Premium. In previous releases, certain paths through the submission wizard resulted in all three Side-by-Side quotes displaying the values from the Basic offering.
PC-15662	Policies are now correctly quoted again when the cancellation effective date is changed. In previous releases, the original, prorated cancellation quote did not change when the cancellation effective date was changed.

<b>ID</b>	<b>Description</b>
PC-15861	Fixed an issue in Side-by-Side quoting where a validation warning blocked quote. The first time a validation warning appears, PolicyCenter prevents quote. Previously, when you clicked <b>Quote All</b> for the second time, PolicyCenter did not quote. Now when you click <b>Quote All</b> the second time, PolicyCenter quotes all versions. This is consistent with how PolicyCenter handles validation warnings in single quote mode.
<b>Line of business</b>	
PC-13922	Previously when writing policies in the workers' compensation line, it was possible to create locations assigned to jurisdictions, and then delete the jurisdictions leaving the locations in place. This is no longer possible. Now, when a jurisdiction is deleted, any locations associated with it are deleted at the same time, and PolicyCenter a message to the user.
<b>Logging</b>	
PL-18234	Improved logging of <code>ConcurrentDataChangeException</code> . Now, the log message shows both of the following: <ul style="list-style-type: none"> <li>The username of the current user, who received the <code>ConcurrentDataChangeException</code></li> <li>The previous user, who made the initial data modification</li> </ul> The log writes the current user's name before the previous user's name in the message.
<b>Messaging</b>	
PL-18211	Fixed an issue in which messages could potentially be sent in incorrect send order if a bundle commit contained multiple event root entities.
<b>Performance</b>	
PC-15382	Improved performance of product availability. For more information, see PL-18299.
<b>Persistence</b>	
PL-17656	Guidewire has introduced a new class <code>com.guidewire.external.Type</code> to use for Java API methods that require an instance of <code>com.guidewire.external.Type</code> . You can obtain an instance of this class using the <code>com.guidewire.external.Type#of</code> method by passing in the class representing the desired type. Methods that require an instance of <code>com.guidewire.external.Type</code> need to document what kinds of types are expected.
<b>Plugins</b>	
PL-17162	Guidewire has added interface <code>gw.plugin.SharedBundlePlugin</code> . Any plugin that implements this interface shares the current bundle rather than creating a new one. For example, any plugin that implements <code>IPreUpdateHandler</code> now needs to add <code>gw.plugin.SharedBundlePlugin</code> to its list of implemented interfaces.
<b>Producer management</b>	
PC-15702	Added a <code>ProducerOfRecord</code> property to <code>PolicyPeriod</code> . <code>ProducerOfRecord</code> is a virtual field pointing to <code>PolicyPeriod.EffectiveDatedFields.ProducerCode.Organization</code> . The <code>PolicyPeriod.Producer</code> field, which points to the same location, is deprecated. On the <code>PolicyInfo</code> screen for a submission, the <code>Organization</code> field is initialized to be the saved <code>ProducerOfRecord</code> on the <code>PolicyPeriod</code> . As recalculate on refresh is required, this field is initialized to the saved value on every post on change. Therefore, updates to the Producer of Record cannot be saved before initializing to the saved value. On the <code>Organization</code> widget, the producer variable has an <code>assignProducer</code> method. This method checks if the field was updated by the user, in which case the field is not initialized to the saved value on refresh.
<b>Queries</b>	
PL-17676	Guidewire has deprecated the <code>RawQuery</code> property on <code>gw.api.database.Query</code> . Instead, create a <code>gw.api.filters.StandardQueryFilter</code> and use its <code>filterQuery</code> method or the <code>IQueryBeanResult.addFilter</code> method to apply the filter to a query.
PL-17719	
<b>Rating</b>	
PC-13868	When accessing the rate table definition detail page for a particular definition, the related information is loaded in the view. For performance, PolicyCenter stores the related information just in case the user wishes to cancel changes. Previously, if you left a rate table definition in edit mode, and rate books were updated to include or exclude this definition within this time, canceling the rate table definition edit did not refresh the <code>Usage</code> tab to reflect rate book changes.
PC-15259	When no row matches a rate table search, PolicyCenter now consistently returns an error regardless of whether the table is in memory or in the database. In previous releases, PolicyCenter would behave differently with in-memory tables than with tables in the database. This was previously a known issue.

ID	Description
PC-15287	When querying a rate book containing multiple rate tables, if a query that is valid for one rate table is run against another rate table for which the query is not valid, PolicyCenter now throws an exception. In previous releases, the invalid query would incorrectly return a value.
PC-15363	You can now create value providers and match operators. In previous releases, the files needed for creating value providers and match operators were not accessible. PolicyCenter now marks the following files as read only, enabling you to examine their content and model your own value providers and match operators: <ul style="list-style-type: none"> <li>• gw.rating.rtm.valueprovider.RateTableCellValueProvider</li> <li>• gw.rating.rtm.matchop.MatchOperationFactory</li> <li>• gw.rating.rtm.validation.MatchOpValidator</li> </ul>
<b>Reinsurance</b>	
PC-15418	PolicyCenter APIs have been changed so that needed reinsurance information is passed to a claims processing system. In previous releases, claims processing systems were unable to retrieve data from PolicyCenter about Per Risk and Annual Aggregate reinsurance agreements.
<b>Revisioning</b>	
PC-15458	The CoverageSymbolGroup foreign key has been removed from the Coverage entity and replaced with an enhancement property. Before upgrading, this column should be nulled out in every Coverage table. A version check has been added to make sure these have been nulled out. This should only affect Business Auto coverages.  Additionally, it is possible, though unlikely, that a carrier has configured their entity extensions to have a similar issue to the one described above with Coverage. If that has occurred, an error will be reported during the upgrade process and the data will need to be fixed before continuing.
<b>Rules Infrastructure</b>	
PL-17893	Guidewire has added a new Server Tools → Info Page that shows the safe persisting order of the Preupdate rules. This page lists the order in which PolicyCenter runs the Preupdate rules for their root entities.
<b>Security</b>	
PL-10126	Guidewire has added a new Credentials plugin to the base configuration. The CredentialsPlugin is similar to the DBAuthenticationPlugin and consists of the following three components: <ul style="list-style-type: none"> <li>• Interface CredentialsPlugin with the following sole method:  <pre>public UsernamePasswordPairBase retrieveUsernameAndPassword(String key)</pre> </li> <li>• Class CredentialsUtil with the following sole static method:  <pre>public static UsernamePasswordPairBase getCredentialsFromPlugin(String key)</pre> </li> <li>• Class CredentialsPlugin.gs provides a default implementation of the plugin in the base configuration.</li> </ul> Code can call the method CredentialsUtil.getCredentialsFromPlugin, passing in a key to specify for which application the credentials are sought. The static method returns the username/password pair for the desired key. The default implementation currently reads the username/password pair from a Credentials.xml file: <ul style="list-style-type: none"> <li>• If the code encounters the token \${username} as the value of the username value in the properties file or the token \${password} as the value of the password value in the properties file, then the code uses the Credentials plugin to obtain the real username and password.</li> <li>• If the Credentials plugin is not enabled or these tokens are not present, the code simply uses the values that are present in the properties file (for backwards compatibility).</li> </ul> However, you can substitute any mechanism for retrieving username/password pairs in place of the default mechanism. This can include retrieving username/password pairs from a directory service or from encrypted fields in a database.
<b>Server Tools</b>	
PL-14011	The (Server Tools) Management Beans → CurrentUserSessions information now displays a sorted list of users. If a user is logged in more than once, PolicyCenter appends the number of sessions to the user name in parentheses. For example: <pre>psmith(2),bsmith</pre> The intent is to preserves all information that is currently available, while providing a view that is easier to scan because of the sorting and the non-repetition of names.
<b>Studio IDE - Debugger</b>	

<b>ID</b>	<b>Description</b>
PL-17929	Fixed an issue that caused a Null Pointer Exception that caused the interface to freeze under some circumstances after connecting Studio to the application server or starting a debugging session on an already-running server.
PL-18106	Fixed an issue in which some actions in Studio, such as setting breakpoints, would occasionally block for a short while waiting for a server response, but did not provide any visible feedback.  Guidewire now displays a progress dialog to indicate the delay.
PL-18110	Previously, if the connection between Studio and the application server was lost (for example, if the server session was terminated for some reason), Studio did not terminate the debug session. As a consequence, after server restart, it was possible for a user to believe that debugging was still ongoing, even though, in reality, the debugger needed to be reset.  The new behavior is that Studio properly terminates the debugger session if it loses connection to the server and notifies the user. The user needs to manually reactivate the debugger after connection to the application server is restored.
PL-18112	Previously, if there was a web service error while activating the Server debugger, it was possible for the activation to fail, but for Studio to show the debugger as active, even it was not. Studio now notifies you of the failure and properly puts the debugger into an inactive state.  Guidewire has also put into place additional recovery mechanisms from certain types of web services errors, making it less likely for the debugger connection to fail in the first place.
PL-18116	Fixed an issue that caused Studio to hang if you were debugging the server and the Gosu Tester debugger was also active. It also occurred if you have an active Gosu Tester debug session, then opened the Guidewire Studio connection dialog and immediately canceled. In that case Studio terminated the Gosu Tester debug session as well.
PL-18275	Fixed a problem with the remote tester debugger hanging if you disconnected it from the server during an active debug session.
PL-18298	Fixed a problem in which it was possible for the Studio tester window to freeze up if the user was using the tester debugger while connected to a remote server and modified some code in Studio and saved it.
<b>Studio IDE - Other</b>	
PL-16798	Modified the Rule Condition parser to accept a statement list, instead of a simple expression only. However, the statement list must contain a return statement. For example:  <pre>uses java.util.HashSet uses gw.lang.reflect.IType  var o = new HashSet&lt;IType&gt;() {A, B, C, ...} return o.contains(typeof(...))</pre>
PL-17386	Added the ability to inspect inner class values in Gosu while debugging in Studio.
PL-17606	Fixed an issue in which Studio incorrectly handled creating a View entity extension.
PL-18107	Studio now shows a broken server connection state (icon) if it loses a server connection, for example, if the user terminates the server process. <ul style="list-style-type: none"> <li>• It detects lost connections very quickly if debugging as there is immediate feedback from the debug API.</li> <li>• It detects lost connections more slowly (but, with not more than 1 minute of delay), if not debugging, as Studio queries the server for status at one minute intervals.</li> <li>• If Studio detects the server has restarted, Studio reconnects automatically.</li> </ul>
PL-18109	Previously, if Studio was connected to a server that was not responding for a period of time, multiple threads would pile up in the Studio process waiting for the server to respond.  Guidewire has improved its handling of restoring Studio's server connection to a server that was previously connected, but became unavailable for a period of time, and then became available again.
PL-18198	Fixed an issue in which, if you set breakpoints in a Studio session, Studio forgot the breakpoints between sessions if you activated the Server debugger after starting the application server, and then shut down and restarted Studio.
PL-18227	Fixed an issue in which PolicyCenter stored the initial value of a newly created script parameter in an incorrect location, thus invalidating the application installation checksum. PolicyCenter correctly stored subsequent entries in the correct location.
<b>Studio IDE - PCF Editor</b>	

ID	Description
PL-11489	Guidewire has added a new PCF Verification option ( <b>Tools</b> → <b>Options</b> → <b>Verification Options</b> ) that you use to enable/disable whether PolicyCenter limits the number of included sections in second-pass PCF verification. (Second-pass verification can cause performance issues with combinations of modal PCF files.) For details, see “Setting Verification Options” in the <i>PolicyCenter Configuration Guide</i> .
PL-17576	<p>Fixed an issue that caused Studio to throw an <code>AssertionError</code> under certain circumstances. This generally occurred if you attempted to load a file (a PCF file, for example) modified outside of Studio into Studio with a stale version of that file already open in Studio. Studio would throw this error upon your return to Studio.</p> <p>This could happen:</p> <ul style="list-style-type: none"> <li>If you opened a file in Studio, shifted focus to another application such as IntelliJ and modified and saved the file, then shifted focus back to Studio.</li> <li>If you opened a file in Studio and shifted focus away from Studio, then (through a source control management system), reloaded a modified version of the file into the local file system and shifted focus back to Studio.</li> </ul> <p>In each case, there was a mismatch between the file stored locally and the version of the file contained in Studio's memory as Studio did not save the open file properly on losing focus.</p>
<b>Studio - PolicyCenter</b>	
PC-14598	Error handing has been improved such that PolicyCenter now displays a meaningful error message such as, <b>An offering in product name is corrupted</b> . In previous releases, creating a product model with an invalid Offering data file would cause a generic null pointer exception during server or Studio startup.
PC-15034	Previously, if a required field was missing from a system table row in Studio, an error would be reported during Studio verification, but the server would still be allowed to start with a warning. Now, the server will not start if a required field is missing from a row in a system table.
<b>Upgrade</b>	
PC-14968	In PolicyCenter 7.0.1, the <code>A11MonopolyStates</code> typecode was removed instead of retired. In this version, the <code>A11MonopolyStates</code> typecode is added back as a retired value. This was a known issue in PolicyCenter 7.0.2.
PC-15275	Improved the speed of database upgrade trigger for <code>AutoNumberSequences</code> on <code>CPBBlanket</code> in the <code>DucatiInitializeCPBBlanketAutoNumberSeqVersionTrigger</code> .
PC-15322	An <code>IllegalArgumentException</code> could occur after upgrading to PolicyCenter 7.0.2 when withdrawing and then viewing a submission. This exception was caused by a change in the way data is stored for account syncable fields. This issue has been fixed by adding an upgrade trigger that sets account syncable, denormalized fields.
<b>User tools</b>	
PC-14190	Fixed an issue where externally managed system tables ( <code>ExternallyManaged = true</code> ) were not synchronized. Now both internally and externally managed system tables are synchronized on server startup and when the user clicks <b>Synchronize System Tables</b> in Studio.
PC-14987	Caching of <code>cClaimSearchCriteria</code> has been disabled to resolve an issue that caused an error when a user edited a work order on a renewal job. The error was caused by editing a policy in an automated workflow resulting in a stale cache.
PC-15061	Fixed an issue in <code>CCCPPolicyLineMapper.gs</code> that threw a null pointer exception in the <code>blanketCovTermExists</code> method. Modified this method to account for <code>CCClassificationCovTerm</code> types with code value of null.
PC-15139	Fixed an issue that caused an exception on the <b>Risk Analysis</b> screen when trying to quote on a policy change.
PC-15182	The <code>PermGen</code> default memory allocation in the application server has been increased to 192 MBytes. In previous release, the default <code>PermGen</code> memory allocation of 128 MBytes was insufficient and could have caused <code>OutOfMemory</code> errors.
PC-15412	When the user changes an incorrect answer, PolicyCenter creates a job history event to record the change. Previously, if the original answer was null, PolicyCenter threw a null pointer exception.
PC-15647	Product module synchronization has been changed so that the checksum stored in the database is removed only if it is invalid. In previous releases, when there were product model warnings, the batch server in a clustered environment would start, but non-batch servers would time out while waiting for the batch server to update the checksum in the database.
PC-15681	Changed a variable name from <code>documentCreationInfo</code> to <code>DocumentCreationInfo</code> (beginning with a capital D). This was changed to avoid a possible error in which Java code might not be able to look up this variable by name because of case-sensitivity on the Java side.

<b>ID</b>	<b>Description</b>
PC-15715	<p>You may have to make a change to <code>logging.properties</code> if you have configured a custom logging level for the <code>BillingSystemPlugin</code> logger category. Previously you would have entered:</p> <pre>log4j.category.BillingIntegration=DEBUG</pre> <p>Now you enter:</p> <pre>log4j.category.Plugin.IBillingSystemPlugin=DEBUG</pre> <p>Of course, you can change DEBUG to WARN, INFO, or another value.</p>
PC-15718	<p>Fixed an issue where PolicyCenter threw an exception in a policy change if you made changes prior to selecting <b>Actions</b> → <b>Edit</b> → <b>Effective Date</b>. Now, PolicyCenter discards all unsaved changes when you select this.</p> <p>You can configure the policy change functionality to save a draft if these are your requirements.</p>
<b>Utilities</b>	
PL-17920	Guidewire has removed the following JAR files from the application build:
	<ul style="list-style-type: none"> <li>• activation.jar</li> <li>• activation-1.1.1.jar</li> </ul>
PL-17922	Guidewire has removed the following JAR files from the Guidewire application builds because the standard Java EE install contains these files:
	<ul style="list-style-type: none"> <li>• mailapi.jar</li> <li>• smtp.jar</li> </ul>
<b>Web - Other</b>	
PL-17861	Added a <code>disablePostOnEnter</code> attribute to the Input Group widget.
<b>Web - UI/Runtime</b>	
PL-18233	Fixed an issue that failed to highlight in yellow a problematic <code>RadioButtonCell</code> field even if the data value for the <code>RadioButtonCell</code> contained an error.
<b>Web Services - WSI</b>	
PL-12215	When you place a WSDL from a web service publisher anywhere in the Gosu class path, Guidewire generates Gosu types from the WSDL elements. A problem in which these generated Gosu types could not be used successfully has been fixed.
PL-16577	To authenticate using HTTP authentication on WebLogic, add the following inside the <code>security-configuration</code> tag of WebLogic's <code>config.xml</code> :
	<pre>&lt;enforce-valid-basic-auth-credentials&gt;false&lt;/enforce-valid-basic-auth-credentials&gt;</pre>
PL-17601	Restored the functionality to extend an interface by a Gosu type. The WSDL was incorrectly overriding the final methods of the <code>Object</code> class ( <code>wait</code> , <code>notify</code> , and similar methods). Eventually, when these WSDLs were transformed into Java classes, the compilation of the Java class failed because the final methods on <code>Object</code> could not be overridden. The fix was to not override these methods in WSDL generation.
PL-17902	Guidewire has added a check for the existence of the generated WSDL directories.
<b>Workflow</b>	
PL-9668	<p>Added new configuration parameter <code>WorkflowLogDebug</code>, which takes a Boolean value:</p> <ul style="list-style-type: none"> <li>• If set to <code>true</code>, PolicyCenter outputs the ordinary verbose system workflow log messages from the Guidewire server to the workflow log.</li> <li>• If set to <code>false</code>, PolicyCenter does not output any of the ordinary system messages.</li> </ul> <p>The setting of this parameter does not have any effect on calls to log workflow messages made by customers. Therefore, all customer log messages are output. If customers are experiencing too many workflow messages being written to the <code>pc_workflowlog</code> table, Guidewire recommends that you set this parameter to <code>false</code>.</p>
PL-17556	<p>It is now possible to run the following writer batch processes from either the (Server Tools) <b>Batch Process Info</b> or (Server Tools) <b>Work Queue Info</b> pages:</p> <ul style="list-style-type: none"> <li>• Activity Escalation</li> <li>• Group Exception</li> <li>• User Exception</li> </ul>
<b>XML Element (and XSD types)</b>	
PL-17934	Fixed an issue in which the XML subsystem did not parse complex types correctly if the complex types within XSD files made circular references to each other.

## Known Issues and Limitations

This section describes known issues with this release of Guidewire PolicyCenter.

- PolicyCenter Known Issues
- Studio/Platform Known Issues

**Note:** For maintenance releases, Guidewire often defers fixing configuration issues if they 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

#### Studio: “Verify Changed Resources” is not implemented in the product model (PC-6928)

**Issue** – The Studio Tools → Verify → Verify Changed Resources command does not work with the PolicyCenter product model.

**Workaround** – Use one of the other verification commands instead. See “Validating Studio Resources” in the *PolicyCenter Configuration Guide* for details.

#### 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.

#### 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. 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 original answer is still saved along with the other answers.

**Workaround** – The answers to these *irrelevant* questions 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)
        }
    }
}
```

#### 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 screen of the agent
- 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.

#### [Attachment inclusions are not copied across terms \(PC-14561\)](#)

**Issue** – RI attachment inclusions are not copied across terms, such as during a renewal or a rewrite.

**Workaround** – Guidewire is aware of this issue.

#### [Contact.AccountHolderCount incorrectly counts the number of accounts \(PC-14736\)](#)

**Issue** – Contact.AccountHolderCount incorrectly counts the number of accounts a particular contact is the account holder on.

**Workaround** – Query the database to count the accounts on which this contact has the account holder role.

#### [Contacts created from the Contact tab are not autosaved \(PC-14761\)](#)

**Issue** – Contacts created from the Contacts tab are not getting autosaved. The **UnsavedWork** light correctly goes on, but when the user clicks elsewhere, the work is lost.

**Workaround** – Set the **autosaveable** attribute on the **NewContact.pcf** to **true**.

#### [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.

#### [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 Are 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.

## AccountAPI.transferPolicies does not throw an exception if archived Policy is transferred to another account (PC-15460)

**Issue** – The AccountAPI.transferPolicies API does not throw an exception if an attempt is made to transfer an archived policy to another account (a prohibited action).

**Workaround** – Check that the policy that you wish to transfer does not have any archived terms before requesting the transfer be made.

## IndustryCodeInput no longer supports 'VALUE' in outputConversion (PC-15603)

**Issue** – In the CreateAccountDV.pcf file, the IndustryCodeInput no longer supports VALUE in the outputConversion property. For example, you cannot enter the following for the outputConversion property:

```
VALUE == null ? "Hello" : "World"
```

**Workaround** – Guidewire is aware of this issue.

## View Progress button in Archive Info throws exception (PC-15843)

**Issue** – If a user with access to the Server Tools administrative section attempts to View Progress of archiving, PolicyCenter may throw a null pointer exception.

**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.

## Class Not Found exception when trying to delete a coverage term option (PC-15902)

**Issue** – When editing the product model in Studio, removing coverage may result in a Class Not Found exception.

**Workaround** – Guidewire is aware of this issue.

## Archived policies are not displayed in the Policies tab for AccountContact file (PC-15917)

**Issue** – If a user visits screens of the account contact file, archived policies and their locations may not appear in the lists. This is because PolicyCenter deleted the records in the active database used to identify to these entities when the policy was archived.

**Workaround** – You can configure PolicyCenter to retain the information about the archived policies and locations in a way that it is available to the searches in the affected screens.

## NullPointerException on Qualification screen (PC-15964)

**Issue** – In some situations, if a user is blocked by an incorrect answer to a pre-qualification question, then change the answer, PolicyCenter may throw a null pointer exception when the user tries to progress to the next screen.

**Workaround** – Guidewire is aware of this issue.

## Studio/Platform Known Issues

### First time you click the arrow of the typekey input, the drop-down menu does not open (PL-10134)

**Issue** – The drop-down menu does not open on the first click of the arrow on a typekey input. Instead, the help text opens.

**Workaround** – Turn off help text on focus by setting `InputHelpTextOnFocus` to `false` in `config.xml`. After you do that, the help text shows only if you mouse over the input, and it does not interfere with opening a drop-down menu.

### XML API upgrade feature missing from documentation (PL-10257)

**Issue** – The *Integration Guide* describes a new set of XML APIs based on the `XmlElement` class. (Legacy APIs are based on the `XmlNode` class.) You can continue to use the legacy APIs. However, the *Integration Guide* omits mentioning an additional upgrade-specific feature.

**Workaround** – For backwards compatibility only, you can import an XML schema into the Gosu type system using the legacy XML system by following these instructions:

1. Copy:

`PolicyCenter/modules/pc/config/registry/compatibility-xsd.xml`

To:

`PolicyCenter/modules/configuration/config/registry/compatibility-xsd.xml`

2. Add an entry for your schema. Set the value of the `namespace` attribute to the Gosu package name of the schema. For example, if the schema is in the package location `my.package` and is called `myschema.xsd`, set the value of `namespace` to `my.package.myschema`.

### Studio Rules do not use correct capitalization for root object's name (PL-10740)

**Issue** – Rule set root objects are not named with first letter lower-cased.

**Workaround** – The Rules engine issues a warning if the correct case for objects is not being used.

### Countries configured in zone-config.xml still generate a warning during regen-dictionary even when zone data is loaded for all of these countries (PL-11947)

**Issue** – Countries configured in `zone-config.xml` still generate a warning during regen-dictionary even when zone data is loaded for all of these countries.

**Workaround** – Warning message is created in error and can safely be ignored.

## User interface cannot handle starting multiple instances of a batch process (PL-12372)

**Issue** – The user interface cannot handle starting multiple instances of a batch process.

**Workaround** – To execute multiple instances of a batch process, start them from the command line. Also, to allow multiple instances to be run simultaneously, you must ensure that the `BatchProcess.isExclusive()` method returns `false`.

## Type system refresh after PCF page title change does not update corresponding menu label (PL-13057)

**Issue** – The type system refresh after a PCF page title change does not update the corresponding menu label.

**Workaround** – After updating a page title, restart the server to refresh menu labels and avoid null pointer exceptions due to stale references.

## Length limitation on entity localization table names (PL-13360)

**Issue** – There is a length limitation on entity localization table names.

**Workaround** – Ensure that the localization `tableName` property specified in the entity extension file is less than 16 characters. If the localization table name exceeds the maximum length, the error message indicates that 18 characters are allowed. However, the error message does not account for two additional characters added by the application.

## US-Locations.txt file with the US geodata from GreatData has special characters that cause validation problems with United States Postal Service (USPS) data (PL-13384)

**Issue** – The `US-Locations.txt` file contains information that does not conform to United States Postal Service (USPS) standards for bulk mailings.

**Workaround** – The provided `US-Locations.txt` file is intended only for use in geocoding to identify addresses for a location. You can edit the `US-Locations.txt` file to conform to your particular address standards, and then import that version of the file instead.

## GX models that reference virtual fields and enhancements throw null pointers if null (PL-13560)

**Issue** – The GX models that reference virtual fields and enhancements throw null pointers when these fields and enhancements are null.

**Workaround** – Include null checks and error handling to prevent referenced virtual fields or enhancements that are null from causing null pointer exceptions.

## Sending email with file attachment with unicode filename is not correctly handed over to the mail server (PL-13582)

**Issue** – An email with a file attachment that has a unicode file name is not sent to the mail server correctly.

**Workaround** – Use Latin characters for file names on attached files.

## 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.

## Cannot make a field from a delegate into a localized column (PL-13761)

**Issue** – You cannot make a field from a delegate into a localized column.

**Workaround** – Move the column to be localized off the delegate and onto each of the implementing entities. Then, to make the column appear as though it exists on the delegate, define an enhancement property on the delegate that *delegates* to the appropriate column, depending on the implementing entity.

## Studio test functionality not working correctly (PL-15153)

**Issue** – If you run a test in the Tests folder in the Studio Resources tree, you get the following exception:

```
Using Test Environment Delegate: com.guidewire.testharness.ConfigEnvTestEnvironmentDelegate
gw.internal.gosu.parser.RuntimeExceptionWithNoStackTrace: java.lang.ClassNotFoundException:
    qa.DothisTest
Caused by: java.lang.ClassNotFoundException: qa.DothisTest
at gw.internal.gosu.parser.TypeLoaderAccess.getIntrinsicTypeByFullName(TypeLoaderAccess.java:522)
at gw.internal.gosu.parser.TypeSystemImpl.getByFullName(TypeSystemImpl.java:139)
at gw.lang.reflect.TypeSystem.getByName(TypeSystem.java:116)
at com.guidewire.studio.junit.ui.RunTestCommand.buildTestSuite(RunTestCommand.java:109
...
...
```

**Workaround** – Run the following command before you run a test:

```
.../bin/gwpc dev-deploy
```

## Problem with *regen-java-api* command and JAR files (PL-16351)

**Issue** – If you run the *PolicyCenter/bin/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 into PolicyCenter using an administrative account.
2. Access Internal Tools → Reload.
3. Click Reload Workflow Engine.

## Deploying EAR File on WebSphere 7.0.0.15 Generates Error Message (PL-18613)

**Issue** – The following steps generate an error message if you attempt to deploy a Guidewire-generated EAR file to WebSphere 7.0.0.15.

1. Generate an EAR file using the following command:  

```
gwpc build-websphere-ear
```
2. Deploy the EAR file to WebSphere.

The deployment fails with an error message similar to the following:

```
com.ibm.websphere.management.application.client.AppDeploy
Exception: com.ibm.websphere.management.application.client.AppDeploymentException:
ADMA0207E: EE 5 module ab.war in ear file contains unsupported xmi format bindings file.
```

**Workaround** – Add the following at the top of the web.xml file before attempting to generate the EAR file.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
"http://java.sun.com/dtd/web-app_2_3.dtd">
```



# Guidewire PolicyCenter 7.0.4

## Release Notes

### Release 7.0.4

#### Release Notes Update: 04-June-2012

---

**IMPORTANT** These release notes replace the release notes that were included in the official product release. Please disregard the earlier version of the release notes.

---

**Document Revision:** 20-May-2014

Copyright © 2001-2014 Guidewire Software, Inc. All rights reserved.

Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Reinsurance Management, Guidewire ContactManager, Guidewire Vendor Data Management, Guidewire Client Data Management, Guidewire Rating Management, Guidewire InsuranceSuite, Guidewire ContactCenter, Guidewire Studio, Guidewire Product Designer, Guidewire Live, Guidewire DataHub, Guidewire InfoCenter, Guidewire Standard Reporting, Guidewire ExampleCenter, Guidewire Account Manager Portal, Guidewire Claim Portal, Guidewire Policyholder Portal, ClaimCenter, BillingCenter, PolicyCenter, InsuranceSuite, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks, service marks, or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries.

All other trademarks are the property of their respective owners.

**This material is confidential and proprietary to Guidewire and subject to the confidentiality terms in the applicable license agreement and/or separate nondisclosure agreement.**

Guidewire products are protected by one or more United States patents.

## Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Support
- Issues and Major Changes
- Improvements and General Issues
- Known Issues and Limitations

## Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

---

**IMPORTANT** If you missed 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 7.0.4.

- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 3.0.4 or later or BillingCenter 7.0.1 or later.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 6.0.0 or later.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 7.0.0.

## 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 Resource Portal – <http://guidewire.custhelp.com>

## Issues and Major Changes

This section contains issues and major changes that may affect your installation.

- Changes and Additions to Rating Management
- Base PCF File Changes
- Base Resources Changes

- Changes in This Release Provided in Upgrade Diff Report

## Changes and Additions to Rating Management

In this release, Guidewire Rating Management now has rate routines and parameter sets.

**Rate routines** define the algorithm for calculating the rate for coverages, taxes, and other costs on a policy. You can create, view, and edit rate routines in PolicyCenter. The rate routine can reference Gosu methods which:

- Perform utility functions such as polynomial calculations
- Call out to third-party systems to get factors
- Implement complex rating logic

**Parameter sets** are associated with rate tables and rate routines. Parameter sets determine the information available for use within a rate routine. Parameter sets also determine the information available for setting the default argument source in the rate table definition. The **Rate Table Definition** screen now has an **Argument Sources** tab where you can set the argument source for parameters.

The personal auto and commercial property lines of business have sample data which includes rate books, rate tables, and rate routines.

For more information, see “Rating Management Concepts” in the *PolicyCenter Application Guide*.

## Base PCF File Changes

All links below require the `ReleaseNotes_files` directory on your local disk.

### PolicyCenter release 7.0.3 to 7.0.4

- To view a report of the changes in the base PCF files in the `modules/pc` directory, [click here](#).
- To view a report of the changes in the base PCF files in the `modules/p1` directory, [click here](#).

## Base Resources Changes

All links below require the `ReleaseNotes_files` directory on your local disk.

### PolicyCenter release 7.0.3 to 7.0.4

- To view a report of the changes in the base resources in the `modules/pc` directory, [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 Resource Portal.

## 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 meant to be a comprehensive list.

ID	Description
<b>Administration</b>	
PC-16003	<p>PL-18232 in this release provides functionality so that you can extend any WorkQueue subclass exposed in PolicyCenter. To extend a WorkQueue subclass you must import these com.guidewire specific classes:</p> <ul style="list-style-type: none"> <li>• uses com.guidewire.pl.system.workqueue.WorkQueue</li> <li>• uses com.guidewire.pl.system.workqueue.DistributedWorker</li> <li>• uses com.guidewire.pl.system.workqueue.WorkQueueTargetFinder</li> </ul>
PC-16455	Database INFO level messages are now written to the main log file instead of to a separate file.
<b>Archiving</b>	
PC-16528	In PolicyCenter archiving, the word <b>Restore</b> has been replaced with <b>Retrieve</b> in the user interface buttons and other labels.
<b>Batch Process</b>	
PL-18232	It is now possible to extend any WorkQueue subclass exposed in PolicyCenter.
PL-18404	<p>Removed the BatchServer config parameter from config.xml. Instead, use the JVM argument or registry element <code>isBatchServer</code> to set the batch server.</p> <p>Also, added the ability to promote a non-batch server to be the batch server on the <b>ClusterInfo</b> page. If a non-batch server is found in the cluster, you can click <b>PromoteToBatch</b> for that server to make that particular server the batch server.</p> <p>Also, added the <code>promoteToBatch</code> operation to the Cluster Management Bean.</p>
PL-18637	<p>The default PolicyCenter behavior for query builder result sets is to retrieve all entries from the database into the application server. If the result set is quite large, this can cause problems. Now, you can use the following method to set the query retrieve chunk size.</p> <pre>public final void setChunkingByID(IQueryResult queryResult, int chunkingSize)</pre>
<b>Clustering</b>	
PL-11554	<p>Modified how Guidewire applications discover and maintain the batch server in the clustered environment. Prior to this change, the batch server was discovered with the help of a message exchange through the JGroups cluster communication channel. Users starting up their nodes at approximately the same time could end up with 2+ batch servers in the cluster, if cluster was not completely formed yet.</p> <p>This changes uses the underlying database to discover and to keep track of the batch server currently active in the cluster.</p>
PL-19141	Improved the usability of (Server Tools) <b>Cluster Info</b> page. The page now shows Server ID and host name information. Guidewire also changed the menu slightly.
<b>Cognos Integration</b>	
PL-19226	The <code>pc cognos</code> command now builds an application-specific Cognos integration ZIP file, with a name of the form <code>pc-cognos-reporting-integration.zip</code> . Each zip file contains application-specific integration scripts ( <code>pc-config</code> and <code>pc-import</code> ). These scripts no longer require the namespace/application parameter.
<b>Command Line Tools</b>	
PL-13663	Removed hard-coded Java heap settings that could cause out-of-memory issues when running certain tools.
PL-18225	Added WSI (document literal) webservices support to Administration command-line tools.
PL-18856	Fixed an issue that involved regenerating the Data Dictionary. Running <code>gwpc regen-dictionary</code> did not actually produce the Data Dictionary files.

ID	Description
PL-18892	<p>Guidewire has added the following optional argument to the data dictionary generation tool:</p> <pre>maxSPVInclusions</pre> <p>This value of this parameter defines the depth for second pass verification that limits the number of shared sections that are included the verification of PCF types using verify-all. For example:</p> <pre>gwpc regen-dictionary -DmaxSPVInclusions=1000</pre> <p>For this case, the second pass compilation of PCF files stops after 1000 permutation of modal PCF files.</p> <p><b>Note</b></p> <ul style="list-style-type: none"> <li>If you do not include this option, the dictionary generation tool behaves as before.</li> <li>Only use positive integer values for the maxSPVInclusions property.</li> </ul>
<b>Configuration Upgrade</b>	
PL-12804	Fixed an issue in which the upgrade tool did not handle rules upgrade properly.
<b>Consistency Checker</b>	
PL-16199	One-to-one relationships can have at most one entity on each side of the relationship. Under certain conditions, Guidewire is unable to create a unique index to enforce this. In those cases, Guidewire now creates a consistency check to verify the integrity of the data.
<b>Contact Domain</b>	
PL-18245	Fixed an issue with IgnoreProperty used in contact-sync-config.xml that caused it to take affect on all subtypes of Contact, rather than only those specified. This functionality now works properly and allows fields to be ignored on contact synchronization only for the specified subtypes.
<b>Contacts and Accounts</b>	
PC-15899	<p>Previously, in CreateAccountDV.pcf file, the IndustryCodeInput did not support VALUE in the outputConversion property. For example, you could not enter the following for the outputConversion property:</p> <pre>VALUE == null ? "Hello" : "World"</pre> <p>In this version, the outputConversion attribute on the IndustryCodeInput no longer generates a verification error. The outputConversion for all usage of IndustryCodeInput is preset, and the value entered into the PCF attribute is not read by PolicyCenter.</p> <p>This was previously a known issue.</p>
PC-16074	The AccountHolder consistency check no longer takes into account retired AccountHolder roles when verifying that only one AccountHolder is defined for each account.
PC-16075	Added a Contact foreign key to MessageHistory. This is now similar to the Contact foreign key on Message.
<b>Database</b>	
PL-1541	PolicyCenter now explicitly names primary key constraints using the following conventions: <ul style="list-style-type: none"> <li>Oracle – Primary key constraint on the table PX_CONTACT will be named as PK_CONTACT.</li> <li>All other databases – Primary key constraint on the table PX_CONTACT will be named as PX_CONTACT_PK.</li> </ul>
PC-16860	There was a problem with setting the appscale attribute for MoneyDataType. The feature worked for currencyamount columns, but not for money columns. The setting, specified in datatypes.xml, was being ignored when a column of type money was retrieved. This setting is now working. The appscale attribute controls the number of digits shown to the right of the decimal point for money or currency in single currency mode. It is similar to the scale attribute. It must be smaller than the scale setting and overrides the scale attribute if defined.
PL-20127	
PL-16199	One-to-one relationships can have at most one entity on each side of the relationship. Under certain conditions, Guidewire is unable to create a unique index to enforce this. In those cases, Guidewire now creates a consistency check to verify the integrity of the data.
PL-17234	PolicyCenter no longer creates shadow tables (tables whose names starts with XXt_ and XXtt_) for new databases if you start the application server in dev mode (JVM parameter -Dgw.server.mode=dev). PolicyCenter now creates shadow tables only if you set the server.running.tests system property to true (explicitly or programmatically).
PL-17286	Guidewire is providing a newer version of the Oracle JDBC driver in the Guidewire application distribution. If you are using an external connection pool instead of the Guidewire-bundled connection pool, then you must copy the new driver from the distribution to the location where the application server data source will pick up the Oracle JDBC driver.

<b>ID</b>	<b>Description</b>
PL-17618	PolicyCenter now logs the reason for generating a certain step during a database upgrade to the server console log (at the DEBUG level). This change also adds this information to the (Server Tools) Upgrade Info page as well.
PL-18637	The default PolicyCenter behavior for query builder result sets is to retrieve all entries from the database into the application server. If the result set is quite large, this can cause problems. Now, you can use the following method to set the query retrieve chunk size. <pre>public final void setChunkingByID(IQueryResult queryResult, int chunkSize)</pre>
PL-19421	Fixed an issue on SQL Server that caused a stack trace overflow if both of the following were true: <ul style="list-style-type: none"> <li>• There was a connection problem during application start-up.</li> <li>• Configuration parameter MigrateToLargeIDsAndDatetime2 was set to false.</li> </ul>
<b>Desktop</b>	
PC-16262	<p>Added a new system permission called <b>View My Accounts</b>. In the default configuration, the Producer, Producer Code - Basics, and Superuser roles have this permission. Accessing the <b>Desktop &gt; My Accounts</b> screen requires this permission.</p> <p>The <b>View My Accounts</b>, <b>View My Submissions</b>, and <b>View My Renewals</b> permissions have been removed from the Audit Examiner, Audit Supervisor, and Premium Auditor roles.</p> <p>The <b>View My Queues</b> permission has been removed from the Premium Auditor role.</p> <p>On the Desktop tab, Audit Examiners and Audit Supervisors only see <b>Activities</b>, <b>Other Work Orders</b>, and <b>Queues</b>. Premium Auditors only see <b>Activities</b> and <b>Other Work Orders</b>.</p> <p>Roles that you created are not automatically updated. You must determine whether to make changes to the permissions included in these roles.</p>
PC-16284	On the Desktop tab, the code for the search filters has been made available in Gosu files which you can access in Studio. You can now add, remove, and edit these search filters.
PC-16286	<p>On <b>My Desktop &gt; My Submissions</b>, <b>My Renewals</b>, and <b>My Other Work Orders</b>, added the following menu items:</p> <ul style="list-style-type: none"> <li><b>Open with activity for me</b> - Display open submissions and issuance work orders for which the current user is assigned to an open activity.</li> <li><b>Open with activity for me due within 7 days</b> - Display open submissions and issuance work orders for which the current user is assigned to an open activity that is due within 7 days.</li> <li><b>Open bound menu item</b> - This displays the current user's open issuance work orders.</li> </ul> <p>In addition, search filters expected to return the most results are only available with the <b>View producer style desktop details</b> permission. This is to limit the search results for users involved in many different work orders.</p> <p>For example, in the default configuration, the Underwriter role previously had a filter that displayed all renewals to which they were assigned. Now, the Underwriter has a filter that displays all renewals to which they are assigned an open activity.</p>
<b>Document Management</b>	
PL-18360	A new version of the Guidewire Document Assistant ActiveX control will be downloaded to the client browser.
PL-18432	Improved performance for retrieval of extremely large documents through the ActiveX Guidewire Document Assistant.
PL-18653	Guidewire has upgraded the BFO PDF library to version 2.11.20. Refer to the following web site for more details on the changes in this version:  <a href="http://bfo.com/viewtext.jsp?url=products/pdf/docs/CHANGELOG.txt&amp;title=Big+Faceless+PDF+Library+ChangeLog">http://bfo.com/viewtext.jsp?url=products/pdf/docs/CHANGELOG.txt&amp;title=Big+Faceless+PDF+Library+ChangeLog</a>
PL-18704	Microsoft Word field forms (text fill-in field) are now restored correctly while generating a document from an Microsoft Word Template.
<b>Entities/Metadata</b>	
PL-14982	Modified the Studio Tools → Verify... feature so that it now reports errors/warnings during verification of Entity Names.
PL-18162	Guidewire has added a validator that prevents you from attempting to denormalize a localized column. Guidewire does not support localized columns for search denorm columns.

ID	Description
PL-18266	<p>Modified the entity type loader to dynamically generate interfaces for the following:</p> <ul style="list-style-type: none"> <li>Entities that do not have a class</li> <li>Entities that are extended with a delegate or Java interface that the compile-time backing class does not extend</li> </ul> <p>The type loader now uses this runtime interface to create array instances. The runtime <code>impl</code> class now implements this dynamically-generated interface as well.</p>
PL-18469	<p>Added a new attribute to entity types called <code>typelistTableName</code>. Use this attribute for non-final entities to specify the name of the corresponding subtype typelist table. If not specified, then PolicyCenter uses the name of the entity as the subtype typelist table name. This capability is useful if an entity name is too long to become a valid typelist table name.</p>
PL-20127	<p>Corrected an issue with setting the <code>appscale</code> attribute for <code>MoneyDataType</code>. The feature worked for <code>currencyamount</code> columns, but not for <code>money</code> columns. The setting, specified in <code>datatypes.xml</code>, was being ignored if a column of type <code>money</code> was retrieved. This setting is now working.</p>
<b>Geocoding/Proximity Search</b>	
PL-18619	<p>Proximity search works around some bad execution plans by disabling index fast full scan and hash join if executing on Oracle. New configuration parameters <code>DisableIndexFastFullScanForProximitySearch</code> and <code>DisableHashJoinForProximitySearch</code> control the workaround. The default value for these parameters is <code>false</code>. These parameters have no effect on databases other than Oracle.</p>
<b>Global Cache</b>	
PL-18322	<p>Fixed an issue with method <code>ProximitySearchQueryUtils.filterWithinRadiusLatLong</code> that did not properly return all the requested points. This issue affected radius searches with a smaller radius value than the number of desired results returned.</p>
<b>Internal Tools/Server Tools Pages</b>	
PL-18017	<p>Guidewire has made the Internal Tools tab available in test mode.</p>
PL-18839	<p>Guidewire has significantly improved the performance of the GW Profiler.</p>
PL-18845	<p>Modified the (Server Tools) Batch Process Info page so that you can now download detailed records for any particular batch process for a specific date range. Clicking Download now opens a page in which you can specify the date range to download for a given batch process.</p>
<b>Jobs</b>	
PC-13051	<p>Fixed an issue with underwriting issues when rewriting a policy with out-of-sequence changes to a new account. Previously, the user received a message that there were unapproved underwriting issues in a future period even though all issues had been automatically approved.</p>
PC-15326	<p>The alternate billing account can now be assigned and removed in rewrite and renewal jobs. This fix required changes to <code>PolicyPeriodBillingEnhancement</code> and the <code>NewTermProcess</code> class. The changes must be merged into your configuration if you have extended these objects.</p>
PC-15719	<p>In <code>ReferenceDatePlugin</code>, replaced entity invocations of <code>getReferenceDateForCurrentJob</code> (on <code>PolicyPeriod</code> and <code>PolicyLine</code>) with local implementations that directly access the <code>ReferenceDatePlugin</code> API. This provides a performance improvement.</p>
PC-15901	<p>Previously, the conversion on renewal process did not create a <code>basedOn</code> <code>PolicyPeriod</code>. With this version, the <code>startConversionRenewal</code> method in <code>gw.webservice.pc.pc700.job.PolicyRenewalAPI</code> creates a <code>basedOn</code> policy period with a unique <code>PolicyPeriodStatus</code> (<code>LegacyConversion</code>).</p>
PC-15909	<p>Previously, automatic renewal was being blocked by bind-level validation warnings. Now the automatic renewal is not blocked by bind-level validation warnings.</p>
PC-16186	<p>The <code>RateAsOfDate</code> field was not set after introducing a new <code>requestQuote()</code> method with a new signature. Added a new callback method <code>#runPreQuote()</code> which populates the field consistently.</p>

<b>ID</b>	<b>Description</b>
PC-16189	<p>In the default configuration, <b>Quote All</b> in the <b>Side by Side Quoting</b> screen does not produce quotes when there are validation warnings. This is to prevent PolicyCenter from generating invalid quotes. Guidewire expects that the default implementation is suitable for most implementations.</p> <p>However, in some limited cases, your implementation may need to <b>Quote All</b> even though there are validation warnings. In this version, you can now change this behavior by adding a parameter to a configuration file. However, this change will necessitate other changes to the PolicyCenter product configuration files or Gosu code.</p> <p>To change <b>Quote All</b> in the <b>Side by Side Quoting</b> screen to not block quote when there are validation warnings, set the <b>SideBySide</b> parameter in <b>config.xml</b> to false. In <b>gw.job.sxs.SideBySideProcess</b>, the call to the <b>requestQuote</b> method has the <b>SideBySide</b> parameter as an argument.</p>
PC-16378	Fixed an issue where an underwriting issue message appeared even after all underwriting messages had been approved in an out-of-sequence change.
PC-16391	Fixed an issue where side-by-side quoting did not alert the user that a quote was invalid. PolicyCenter now displays a message to the user.
<b>Lines of Business</b>	
PC-16152	Removed an unnecessary Row visibility test from the <b>CPBuildingsAndLocationsLV</b> that was recalculating Location Buildings for every Location Building.
PC-16198	In a personal auto policy, the primary location is now set to the garage location of the vehicle with the lowest <b>VehicleNumber</b> (not VIN). Previously the primary location was set to the garage location of the vehicle with a <b>VehicleNumber</b> of 1, but the vehicle numbers do not always start at 1.
<b>Localization</b>	
PC-16543	Fixed an issue where the user could not quote when the <b>MoneyDataType</b> property set <b>appscale=0</b> in <b>datatypeypes.xml</b> .
PL-18490	Modified <b>localization.xml</b> to let the <b>thousandsSymbol</b> attribute of the <b>&lt;NumberFormat&gt;</b> element type accept a non-breaking space character as the thousands separator. For example:
	<b>&lt;NumberFormat thousandsSymbol="&amp;#160;"&gt;</b>
<b>Locations</b>	
PC-16009	Fixed an issue which prevented removing an account location that has just been added if a policy location was also added.
<b>Logging</b>	
PL-14722	Added the following new configuration parameters that configure application logging:
	<ul style="list-style-type: none"> <li>• <b>LoggerCategorySource</b></li> <li>• <b>LoggersShowLog4j</b></li> </ul>
PL-18745	<p>Added support for two new Log4j MDC (Mapped Diagnostic Contexts) keys to include information about the current user in log messages. To use them, include a sequence conforming to the following string in your <b>ConversionPattern</b> in <b>logging.properties</b>:</p> <p style="padding-left: 20px;">%-&lt;LL&gt;. &lt;HH&gt;X{user   userName   userID}</p> <p>The parameters have the following meanings:</p> <ul style="list-style-type: none"> <li>• <b>&lt;LL&gt;</b> – The minimum size of the field. If the actual value is shorter, the user name gets padded with spaces on the right.</li> <li>• <b>&lt;HH&gt;</b> – The maximum size of the field. If the actual value is longer, the user name gets truncated from the left.</li> <li>• <b>user</b> – Prints the user's internal numeric ID number (for example, 4231341234) (This was available in previous releases and remains unchanged for compatibility.)</li> <li>• <b>userName</b> – Prints the user's real-world name (for example, John Smith).</li> <li>• <b>userID</b> – Prints the user's username in the system (for example, jsmith).</li> </ul> <p>For example:</p> <p style="padding-left: 20px;">%-16.16X{userName}</p>

ID	Description
PL-20907	<p>Guidewire updated the logging API between the following releases:</p> <ul style="list-style-type: none"> <li>• BillingCenter 7.0.1 and 7.0.2</li> <li>• ClaimCenter 7.0.1 and 7.0.2</li> <li>• PolicyCenter 7.0.3 and 7.0.4</li> </ul> <p>Guidewire has created the following Knowledge Base article that describes the changes made to the logging API:</p> <p style="padding-left: 2em;"><i>Changes to the logging API during upgrade to BillingCenter 7.0.2, ClaimCenter 7.0.2, and PolicyCenter 7.0.4 or later versions</i></p> <p>Review this article if you are performing an upgrade, in order to update your configuration to the new logging API. If necessary, contact Guidewire Support for a copy.</p>
<b>Messaging</b>	
PL-18678	<p>The MessageSenderRunnable.run method now includes the messageID in the log. The log entry will look similar to the following:</p> <pre>MW.MessageSenderRunnable.run (dest destination id): Entering run() for messageId message id</pre>
PL-19136	Fixed an issue in which a race condition in a non-clustered server could lead to inconsistent results and/or runtime exceptions when modifying entities within a message transport.
PL-19149	After locking out a user who has reached the maximum allowed login failures, PolicyCenter no longer ignores generated events. Instead, PolicyCenter executes the transaction with a service token having a super user. This means that PolicyCenter now generates UserChanged events properly.
<b>Miscellaneous</b>	
PL-3760	<p>Guidewire has made the following modification to config.xml:</p> <ul style="list-style-type: none"> <li>• Removed element &lt;security sessiontimeoutsecs="10800"/&gt;</li> <li>• Added configuration parameter &lt;param name="SessionTimeoutSecs" value="10800"/&gt; as its replacement</li> </ul> <p>PolicyCenter performs an automatic upgrade of config.xml for this change.</p>
PL-15462	<p>Guidewire applications contain several configuration parameters that accept a comma-separated list of values.</p> <p>Now, PolicyCenter parses a comma-separated list of values according to CSV format and trims each value automatically. This permits you to use spaces (and tabs, new lines) for more readability while specifying such a value.</p> <p>For example:</p> <ul style="list-style-type: none"> <li>• A value that before the fix must be without spaces – AUTO, PR, GL, TRAV</li> <li>• The same value after the fix can use spaces – AUTO, PR, GL, TRAV</li> </ul> <p>The following is a more technical description of the format:</p> <ul style="list-style-type: none"> <li>• Fields must be separated with commas.</li> <li>• Leading and trailing spaces are ignored—unless the field is delimited with double-quotes. In that case the white spaces are preserved.</li> <li>• Embedded comma – You must delimit the field with double-quotes, with the comma inside the double quotes.</li> <li>• Embedded double-quotes – You must double embed double-quote characters and you must delimit the field with double-quotes.</li> <li>• Embedded line-breaks – You must surround the field with double-quotes.</li> <li>• Always Delimiting - You can always delimit a field with double-quotes. If not strictly needed, the delimiters will be parsed and discarded by the reading applications.</li> </ul>
PL-15914	Guidewire has added Save and Cancel buttons on the (Server Tools) Management Beans → Guidewire Managed Bean Properties page that you can use to save or cancel a change to a bean property value. The buttons become active after you edit an editable property.
PL-18245	Fixed issue with IgnoreProperty used in contact-sync-config.xml that caused it to take affect on all subtypes of Contact, rather than only those specified. This functionality now works properly and allows fields to be ignored on contact synchronization only for the specified subtypes.
PL-19311	Guidewire has upgraded the Joda-Time third-party library version to 2.0.
<b>Page Configuration</b>	

<b>ID</b>	<b>Description</b>
PL-15548	Printing a second-level list view no longer results in missing items or items to print more than once.
PL-18303	Adding multiple entries in list views for custom entity types no longer causes errors.
PL-19602	Added the ability to retain the scroll position after clicking a pull-right menu item. This fix adds the <code>retainScrollPosition</code> property to the <code>MenuItem</code> widgets.
<b>Performance</b>	
PC-15382	In PolicyCenter, there is now an example <code>preload.txt</code> file based on the default configuration. However, you should regenerate this file in your environment, as you will likely have your own Gosu code that need to be pre-loaded. In addition, the example file may pre-load classes that you do not want or need. For more information, see the "Preloading Gosu Classes" section in the <i>Configuration Guide</i> . See also PL-18511.
PL-18511	Added in a preload mechanism to support pre-compilation of Gosu classes, as well as other primary classes in the system. The intent is to make the system more responsive the first time requests are made. To support this, Guidewire has added a Studio Other Resources → <code>preload.txt</code> file that in which you can add a list actions to take. The file contains static no-argument method calls, as well as the names of Gosu types to compile to bytecode or the Java types to load.  Guidewire also added a new logging category of <code>Server.PreLoad</code> that provides DEBUG level logging of all actions during server preloading of Gosu classes.
<b>Persistence</b>	
PL-18162	Guidewire has added a validator that prevents you from attempting to denormalize a localized column. Guidewire does not support localized columns for search denorm columns.
<b>PL Services</b>	
PL-11554	Modified how Guidewire applications discover and maintain the batch server in the clustered environment. Prior to this change, the batch server was discovered with the help of a message exchange through the JGroups cluster communication channel. Users starting up their nodes at approximately the same time could end up with 2+ batch servers in the cluster if cluster was not completely formed yet.  This changes uses the underlying database to discovery and keep track of the batch server currently active in the cluster.
PL-18232	It is now possible to extend any <code>WorkQueue</code> subclass exposed in PolicyCenter.
<b>Product Integrations</b>	
PC-15105	In previous versions of PolicyCenter, PolicyCenter did not send an event to create the internal carrier organization as a producer in BillingCenter. Now every time PolicyCenter starts, PolicyCenter checks to see if the organization is a producer. If so, PolicyCenter sends a message asking BillingCenter to check whether it already has a producer for the internal carrier organization. If not, then PolicyCenter sends a message asking BillingCenter to create the producer for the internal carrier organization.
PC-16011	Made the following changes in the BillingCenter integration: <ol style="list-style-type: none"><li>1. Before sending charges to BillingCenter, set the <code>Written Date</code> field for Transactions.</li><li>2. Send a value for <code>policyperiod.TermConfirmed</code> to BillingCenter for every new BillingCenter policy period.</li></ol>
PC-16371	PolicyCenter provides the ability to export the PolicyCenter product model to the ClaimCenter Line of Business model. This process was not working properly for coverages created in ClaimCenter. This export process is now fixed. It reads in all existing ClaimCenter Line of Business elements that were created natively in ClaimCenter, adds the PolicyCenter model elements to them, and then exports everything back to ClaimCenter.
<b>Product Model</b>	
PC-15902	In previous versions, when editing the product model in Studio, removing coverage may have resulted in a Class Not Found exception. This issue is fixed in this version. This was previously a known issue.
PC-15856	It is now possible to establish grandfathering based upon coverages on converted terms. The changes that resolved PC-15901 added this capability.
PC-15964	Previously, in some situations, if a user a blocked by an incorrect answer to a pre-qualification question, then change the answer, PolicyCenter threw a null pointer exception when the user tried to progress to the next screen. This is fixed in this version. This was previously a known issue.
<b>Queries</b>	
PL-17541	Modified <code>GWDBFunctionEnhancement.gsx</code> and added support for <code>DBFunction</code> calls of <code>MIN</code> and <code>MAX</code> on <code>java.util.Date</code> .

ID	Description
Rating	
PC-16682	<p>The Cost entity and the CostData class now support flat costs. A flat cost represents a set amount that either exists or does not independent of length of term. For example, a paperwork fee might exist whenever the insured adds an Additional Named Insured to the policy. If a cost is a flat cost, PolicyCenter does not adjust it based on percentage of term length. In other words, PolicyCenter does not prorate the cost.</p> <p>Qualities of flat costs include: (1) If added midterm, the flat cost full amount is charged. (2) If removed midterm, the flat cost full amount is still charged unless it is removed on the same effective date that it was added. (3) If you lengthen or shorten the term later, the amount does not change. (4) For a pro rata cancel, flat charges remain a full charge. However, if a policy change adds a flat cost effective after submission and a pro rata cancel occurs effective at or before the policy change effective date, PolicyCenter reverses the flat cost. (5) Upon flat cancel, a flat charge is refunded. (6) A policy change involving a flat cost typically generates a pair of transactions. The first transaction reverses the charge for the old date range. The second transaction reapplys the charge over the new date range. (7) If you put a flat charge on the policy, take it off at a later effective date, and add it again at another later effective date then it is charged twice.</p> <p>Both Cost and CostData now have a ProrationMethod property. The property contains a ProrationMethod typekey. In the default configuration, it contains two values: proratabydays (the previous pro rata behavior) and f1at (a flat cost). The default implementation of the method CostData.getAmount() checks the value of this typekey. If the value is proratabydays, the method uses the old prorated behavior. If it is f1at, this method returns the full amount. If it is some other value, it calls the protected method computeExtendedAmount. If you add additional values to the ProrationMethod typelist, modify computeExtendedAmount to implement custom logic for additional values.</p> <p>PolicyCenter 7.0.4 does not use flat cost behavior in any built-in line of business. To create new lines of business that use flat costs, or change existing lines of business, the primary requirement is to create CostData objects with the ProrationMethod property set to flat. You must never change the ProrationMethod value for an existing Cost object. Only set ProrationMethod to f1at on a newly-generated CostData object.</p> <p><b>IMPORTANT:</b> To maximize backward compatibility in this maintenance release, there is no change to the behavior of the Cost entity properties Prorated and Proration. These properties operate strictly on the date range of the cost, so they ignore ProrationMethod property and behave as if the proration method is always proratabydays. These two properties are for display purposes only and thus are used exclusively in PCF files in the default configuration. If you use flat costs, you can change your PCF files to check ProrationMethod to avoid displaying confusing Proration values on flat costs.</p>
PC-15388	On the Rate Table Content panel on the Rate Table screen in a rate book, the 0 after decimal point disappeared in read mode. For example, if the factor was a Double type and you entered 8.0 or 8.00, the zeros after decimal disappeared when you clicked <b>Update</b> .
PC-16066	When exporting a rate table to Microsoft Excel, PolicyCenter exports to .xlsx format. Previously, PolicyCenter exported to .xls format.
PC-16141	When viewing the Rate Table screen in a Rate Book, the Rate Table Content panel displayed 0E-10 instead of 0 if you added a rate factor of zero for a double type.
PC-16153	Added information about the user who last updated the following: <ul style="list-style-type: none"> <li>• For Rate Book – Next to the status change date and time for Rate Book also show the username.</li> <li>• For Rate Table Definition – In the Basics tab at the bottom, add a last edit date and user name.</li> <li>• For Rate Routine – Below the Parameters in the Parameter Set show the last edit date and the user name.</li> </ul>
PC-16321	<p>The Data Type named Double is now named Decimal. This label appears on the Parameter Details tab of the New Rate Table Definition screen and other places.</p> <p>On the Factors tab, for Data Type Decimal, there is now a Decimal Places field to specify the number of decimal places. The Physical Column field now specifies the precision and scale of the decimal number. For example, Dec1 (15, 7).</p>
PC-16322	Fixed an issue where PolicyCenter did not display the decimal values in rate table factors. PolicyCenter now displays the decimal values.
PC-16472	Fixed an issue where PolicyCenter did not display an error if the rate table included in a rate book contained values that did not match the rate table definition. For example, a rate table has a decimal factor with precision and scale of 10, 15. In the rate table, the user enters an alphanumeric value for the factor. Previously, PolicyCenter did not display a warning. Now PolicyCenter displays a warning.

<b>ID</b>	<b>Description</b>
PC-16616	The default physical table when creating new rate tables uses DefaultRateFactorRow.etc. DefaultRateFactorRow uses a precision of 15 and a scale of 7 for decimal typed columns. Therefore, 8 digits left of the decimal point are available when using the default physical table for decimal parameters and factors.
PC-16653	Rename the Rate Book Version field to <b>Rate Book Edition</b> . Reorder the Rate Book and Rate Table Definition lists: The Name is before the code and the hyperlinks are on the Name. Change the Rate Book Version button to <b>Create New Edition</b> .
<b>Reinsurance</b>	
PC-15965	The gw.plugin.reinsurance.ReinsuranceConfigPlugin class contained a timestamp that could not be localized. This caused an exception when starting PolicyCenter if the TimeFormat in localization.xml differed from the en_US format.
<b>Search</b>	
PC-15619	Account search now requires more information to meet the minimum search criteria. When searching by person name, a first and last name is required. Additionally, last name must either be an exact match, or the location information must be provided. Location information can be either city and state or zip code. Company names require either the first five characters, or a search for an exact name. It is also possible to search just a work phone number, tax id, account number, producer, or producer code.
PC-15620	Contact search now requires more information to meet the minimum search criteria. When searching by person name, a first and last name is required. Additionally, last name must either be an exact match, or the location information must be provided. Location information can be either city and state or zip code. Company names require either the first five characters, or a search for an exact name. It is also possible to search just a work phone number or tax id.
PC-15621	Minimum search criteria for policies have been made more stringent; requiring users enter more information before a search is possible. When searching by person name, a first and last name is required. Additionally, last name must either be an exact match, or the location information must be provided. Location information can be either city and state or zip code. Company names require either the first five characters, or a search for an exact name. It is also possible to search by just producer, producer code, account number, primary insured work phone number, or job number.
<b>Studio IDE</b>	
PL-13352	Fixed an issue in which Studio ignored the formatting settings for String literals set in Studio Tools → Options → Colors and Fonts.
PL-15219	Fixed an issue with PCF TemplateInputWidget in which autocomplete inserted a value that appeared to be correct, but which failed validation.
PL-15887	Fixed an incorrect message that Studio showed in the Verify Result pane if verifying a valid entity name in the Entity Names editor.
PL-18203	Fixed an issue in which the Studio debugger step-over functionality stopped on Guidewire internal code in between executing Rule Condition and Rule Action code.
PL-18671	Guidewire has modified Studio behavior in regards to readonly mode: <ul style="list-style-type: none"> <li>Studio now displays a padlock button on the status bar that is visible only if it is in readonly mode. If you click the button, Studio displays a modal message box indicating the reason(s) why it is in readonly mode.</li> <li>Studio disables the Save button any time that it is in readonly mode.</li> <li>Studio changes the Save button tooltip in readonly mode to display the reason that save is not active in this mode. This is the same message that Studio shows if you click the padlock icon on the status bar.</li> </ul>
PL-18905	Fixed an issue in which it was possible to edit configuration files even if Studio was in readonly mode.
<b>Utilities</b>	
PL-15733	Guidewire has updated the Google Guava library to release 10.
<b>Web</b>	
PL-15548	Printing a second-level list view no longer results in missing items or items to print more than once.
PL-18303	Removed errors caused while adding multiple entries in list views for custom entity types.

ID	Description
PL-19454	Fixed an issue that occurred while using Internet Explorer 9 in which clicking a drop-down list did not render the list properly.
PL-19602	Added the ability to retain the scroll position after clicking a pull-right menu item. This fix adds the <code>retainScrollPosition</code> property to the <code>MenuItem</code> widgets.
<b>Web Services</b>	
PL-18361	PolicyCenter now provides the ability to implement a pre-existing WSDL in Gosu for WSI web service.
PL-18450	PolicyCenter now consumes MTOM-enabled WS-I web services. MTOM is the W3C Message Transmission Optimization Mechanism that efficiently sends binary data to and from Web services. You do not need to do anything to take advantage of this new feature.
PL-18589	Added a logging category for WSI webservices to log each request, <code>XML.Request</code> . Each request will be logged at the debug level. This include the connecting address and user, if available, as well as the request <code>qname</code> (which is unique for each operation).
<b>Work Queues</b>	
PL-18675	Fixed several issues that involved the (Server Tools) <code>Work Queue Info</code> page.
PL-19003	You can now start or stop work queues from <code>IMaintenanceToolsAPI</code> .
PL-19004	Added the attribute <code>orphansfirst</code> to a worker thereby making it possible to specify that PolicyCenter is to process orphans before new items on this worker.
PL-19059	The (Server Tools) <code>Work Queue Info</code> page now reports the last time that PolicyCenter processed a work item, showing either the last success time or the last exception time. If it was an exception, the <code>Info</code> page also reports the number of consecutive work items that resulted in an exception. The page shows two other dates as well. The last notification time is the last time the worker woke up. If it found work, PolicyCenter updates the last batch found time.

## Known Issues and Limitations

This section describes known issues with this release of Guidewire PolicyCenter.

- PolicyCenter Known Issues
- Studio/Platform Known Issues

**Note:** For maintenance releases, Guidewire often defers fixing configuration issues if they 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

#### Studio: “Verify Changed Resources” is not implemented in the product model (PC-6928)

**Issue** – The Studio Tools → Verify → Verify Changed Resources command does not work with the PolicyCenter product model.

**Workaround** – Use one of the other verification commands instead. See “Validating Studio Resources” in the *PolicyCenter Configuration Guide* for details.

#### Provide a way to filter out added and removed subtentities 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.

## 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. 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 original answer is still saved along with the other answers.

**Workaround** – The answers to these *irrelevant* questions 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)
        }
    }
}
```

## 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 screen of the agent
- 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.

## Attachment inclusions are not copied across terms (PC-14561)

**Issue** – RI attachment inclusions are not copied across terms, such as during a renewal or a rewrite.

**Workaround** – Guidewire is aware of this issue.

### Contact.AccountHolderCount incorrectly counts the number of accounts (PC-14736)

**Issue** – Contact.AccountHolderCount incorrectly counts the number of accounts a particular contact is the account holder on.

**Workaround** – Query the database to count the accounts on which this contact has the account holder role.

### Contacts created from the Contact tab are not autosaved (PC-14761)

**Issue** – Contacts created from the Contacts tab are not getting autosaved. The **UnsavedWork** alert correctly goes on, but when the user clicks elsewhere, the work is lost.

**Workaround** – Set the **autosaveable** attribute on the **NewContact.pcf** to **true**.

### 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.

### 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 Are 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.

### View Progress button in Archive Info throws exception (PC-15843)

**Issue** – If a user with access to the **Server Tools** administrative section attempts to **View Progress** of archiving, PolicyCenter may throw a null pointer exception.

**Workaround** – Guidewire is aware of this issue.

## Renewal processing for renewal conversion assumes BasedOn will be 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 new `gw.webservice.pc.pc700.job.startConversionRenewal` method. For more information, see PC-15901.

## Archived policies are not displayed in the Policies tab for AccountContact file (PC-15917)

**Issue** – On the **Account File Contacts** screen, archived policies are not appearing in the **Associated Policies** and **Associated Work Orders** tabs. On the **Account File Locations** screen, archived policies are not appearing in the **Policy Terms** and **Work Orders** list views. This is because PolicyCenter deleted the records in the active database used to identify to these entities when the policy was archived.

**Workaround** – You can configure PolicyCenter to retain the information about the archived policies in a way that it is available to the searches in the affected screens.

## System table verification in product model raises error for localized field (PC-16274)

**Issue** – When verifying a system table with a localized field, Studio reports an error similar to:

Unknown field “`Name_L10N-ARRAY`”. The file may have been edited outside of Studio. Fix the error directly in XML.

**Workaround** – Do not switch the Studio locale in the middle of editing localized fields in system tables.

## 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 cancellation, 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 **Non Renew** or **Not Take** the renewal. 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.

## Null pointer when questions are answered out of order (PC-16531)

**Issue** – In certain situations, where questions have been configured to be filtered by the responses to other questions, PolicyCenter throws a null pointer exception after the user enters an answer that makes a previously visible question disappear.

**Workaround** – Guidewire is aware of this issue.

## Production server error related to forms after upgrading to PolicyCenter 7 (PC-16651)

**Issue** – After upgrading from PolicyCenter 4 or earlier to PolicyCenter 7.0.0 or later, PolicyCenter throws an error when starting a production server:

```
java.lang.IllegalArgumentException: Unknown product model class: ProductFormProductPattern
```

In PolicyCenter 7.0.0, form definition was moved from the product model in Studio to the PolicyCenter user interface. Therefore, form definition is now stored in the database. This problem is caused because there is not a database upgrade trigger that removes the forms-related product model rows.

**Workaround** – Contact Guidewire Support for guidance on manually removing the affected database rows.

## Invalid quote on sample data policy after upgrade (PC-16847)

**Issue** – The PCRatingPlugin generates an invalid quote on a policy in the sample data. This issue occurs if you have sample data in a PolicyCenter 7.0.3 test environment and upgrade that test environment to PolicyCenter 7.0.4. The structure of the rating data changed between PolicyCenter 7.0.3 and 7.0.4 and causes this problem.

**Workaround** – After upgrading to PolicyCenter 7.0.4, update the sample data as follows:

1. Start PolicyCenter 7.0.4.

2. Load the large sample data set. The sample data now contains a mix of PolicyCenter 7.0.3 and PolicyCenter 7.0.4 rating data.

3. In Studio, connect to the PolicyCenter 7.0.4 server as su.

4. Save the following code as `sampleDataRemoval.gs`.

```
uses gw.transaction.Transaction
uses gw.api.database.Query

function findEntity<T extends KeyableBean>() : List<T>{
    var q = Query.make(T)
    q.startsWith("PublicID", "pc:", false /*ignoreCase*/)
    return q.select().toList()
}

Transaction.runWithNewBundle(\ bundle -> {
    findEntity<RateBook>().each(\ rb -> bundle.add(rb).remove())
    findEntity<RateTableDefinition>().each(\ rt -> bundle.add(rt).remove())
    findEntity<RateTableMatchOpDefinition>().each(\ rb -> bundle.add(rb).remove())
    findEntity<RateFactorRow>().each(\ rb -> bundle.add(rb).remove())
    findEntity<CoverageRateFactor>().each(\ rb -> bundle.add(rb).remove())
    findEntity<CalcRoutineDefinition>().each(\ rb -> bundle.add(rb).remove())
}, "su")
```

5. Run the `sampleDataRemoval.gs` script in the Gosu tester in Studio. This script deletes all rating data such as rate books, rate table definitions, and rate routines. The script does not delete parameter sets.

6. Save the following code as `reloadSampleData.gs`.

```
uses gw.transaction.Transaction
uses gw.sampledata.small.SmallSampleRatingData
uses gw.sampledata.tiny.TinySampleRatingData

Transaction.runWithNewBundle(\ bundle -> {
    var tinyData = new TinySampleRatingData()
    tinyData.load()
    var sampleData = new SmallSampleRatingData()
    sampleData.load()
}, "su")
```

7. Run this script in Gosu tester in Studio. This script reloads the PolicyCenter 7.0.4 rating data.

## Term number is set to null while creating a new branch for preemption (PC-16868)

**Issue** – The term number is incorrectly set to null on a policy change that has been preempted by another policy change.

**Workaround** – Guidewire is aware of this issue.

## Term number is set to null, when effective date is edited on Policy change (PC-16869)

**Issue** – The term number is set to null if you edit the effective date on a policy change.

**Workaround** – Guidewire is aware of this issue.

## Studio/Platform Known Issues

### First time you click the arrow of the typekey input, the drop-down menu does not open (PL-10134)

**Issue** – The drop-down menu does not open on the first click of the arrow on a typekey input. Instead, the help text opens.

**Workaround** – Turn off help text on focus by setting `InputHelpTextOnFocus` to `false` in `config.xml`. After you do that, the help text shows only if you mouse over the input, and it does not interfere with opening a drop-down menu.

### XML API upgrade feature missing from documentation (PL-10257)

**Issue** – The *Integration Guide* describes a new set of XML APIs based on the `XmlElement` class. (Legacy APIs are based on the `XmlNode` class.) You can continue to use the legacy APIs. However, the *Integration Guide* omits mentioning an additional upgrade-specific feature.

**Workaround** – For backwards compatibility only, you can import an XML schema into the Gosu type system using the legacy XML system by following these instructions:

1. Copy:

```
PolicyCenter/modules/pc/config/registry/compatibility-xsd.xml
```

To:

```
PolicyCenter/modules/configuration/config/registry/compatibility-xsd.xml
```

2. Add an entry for your schema. Set the value of the `namespace` attribute to the Gosu package name of the schema. For example, if the schema is in the package location `my.package` and is called `myschema.xsd`, set the value of `namespace` to `my.package.myschema`.

### Studio Rules do not use correct capitalization for root object's name (PL-10740)

**Issue** – Rule set root objects are expected to begin with an upper-case letter and issues warning if a rule set root object does not.

**Workaround** – You can safely ignore the warning message.

### Countries configured in zone-config.xml still generate a warning during regen-dictionary even when zone data is loaded for all of these countries (PL-11947)

**Issue** – Countries configured in `zone-config.xml` still generate a warning during `regen-dictionary` even when zone data is loaded for all of these countries.

**Workaround** – Warning message is created in error and can safely be ignored.

### User interface cannot handle starting multiple instances of a batch process (PL-12372)

**Issue** – The user interface cannot handle starting multiple instances of a batch process.

**Workaround** – To execute multiple instances of a batch process, start them from the command line. Also, to allow multiple instances to be run simultaneously, you must ensure that the `BatchProcess.isExclusive()` method returns `false`.

## Type system refresh after PCF page title change does not update corresponding menu label (PL-13057)

**Issue** – The type system refresh after a PCF page title change does not update the corresponding menu label.

**Workaround** – After updating a page title, restart the server to refresh menu labels and avoid null pointer exceptions due to stale references.

## Length limitation on entity localization table names (PL-13360)

**Issue** – Entity localization table names have a shorter, 16-character limit than other entity tables. If the localization table name exceeds the 16-chara limit, the error message indicates incorrectly that 18 characters are allowed.

**Workaround** – Ensure that the localization tabLeName property specified in the entity extension file is 16 characters or less.

## US-Locations.txt file with the US geodata from GreatData has special characters that cause validation problems with United States Postal Service (USPS) data (PL-13384)

**Issue** – The US-Locations.txt file contains information that does not conform to United States Postal Service (USPS) standards for bulk mailings.

**Workaround** – The provided US-Locations.txt file is intended only for use in geocoding to identify addresses for a location. You can edit the US-Locations.txt file to conform to your particular address standards, and then import that version of the file instead.

## GX models that reference virtual fields and enhancements throw null pointers if null (PL-13560)

**Issue** – GX models that reference virtual fields and enhancements throw null pointers when these fields and enhancements are null. While object path expressions are generally null safe, they are not null safe when the expression navigates to a virtual field or enhancement.

**Workaround** – Include null checks and error handling to prevent referenced virtual fields or enhancements that are null from causing null pointer exceptions. For example, the following object path navigates to a virtual field.

```
anActivity.ActivityStatus
```

Use the null-safe period operator instead of the standard operator, as the following example shows:

```
anActivity?.ActivityStatus
```

## Sending email with file attachment with unicode filename is not correctly handed over to the mail server (PL-13582)

**Issue** – An email with a file attachment that has a unicode file name is not sent to the mail server correctly.

**Workaround** – Use Latin characters for file names on attached files.

## 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.

## Cannot make a field from a delegate into a localized column (PL-13761)

**Issue** – You cannot make a field from a delegate into a localized column.

**Workaround** – Move the column to be localized off the delegate and onto each of the implementing entities. Then, to make the column appear as though it exists on the delegate, define an enhancement property on the delegate that *delegates* to the appropriate column, depending on the implementing entity.

### Problem with *regen-java-api* command and JAR files (PL-16351)

**Issue** – If you run the PolicyCenter/bin/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 into PolicyCenter using an administrative account.
2. Access Internal Tools → Reload.
3. Click Reload Workflow Engine.

# Guidewire PolicyCenter 7.0.5 Release Notes

## Release 7.0.5

**Document Revision:** 20-May-2014

Copyright © 2001-2014 Guidewire Software, Inc. All rights reserved.

Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Reinsurance Management, Guidewire ContactManager, Guidewire Vendor Data Management, Guidewire Client Data Management, Guidewire Rating Management, Guidewire InsuranceSuite, Guidewire ContactCenter, Guidewire Studio, Guidewire Product Designer, Guidewire Live, Guidewire DataHub, Guidewire InfoCenter, Guidewire Standard Reporting, Guidewire ExampleCenter, Guidewire Account Manager Portal, Guidewire Claim Portal, Guidewire Policyholder Portal, ClaimCenter, BillingCenter, PolicyCenter, InsuranceSuite, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks, service marks, or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries.

All other trademarks are the property of their respective owners.

**This material is confidential and proprietary to Guidewire and subject to the confidentiality terms in the applicable license agreement and/or separate nondisclosure agreement.**

Guidewire products are protected by one or more United States patents.

## Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Support

- Issues and Major Changes
- Improvements and General Issues
- Known Issues and Limitations

## Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

---

**IMPORTANT** If you missed 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 7.0.5.

- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 3.0.4 or later or BillingCenter 7.0.1 or later.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 6.0.0 or later.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 7.0.0.

## 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 Resource Portal – <http://guidewire.custhelp.com>

## Issues and Major Changes

This section contains issues and major changes that may affect your installation.

- Changes and Additions to Rating Management
- Generic Schedules
- Localization Improvements
- Logging API Changes
- Base PCF File Changes
- Base Resources Changes
- Changes in This Release Provided in Upgrade Diff Report

## Changes and Additions to Rating Management

In this release, Guidewire Rating Management now includes the following new features:

- **Rating worksheets** – Show the actual values that the rate routine used to calculate the rate for a quoted policy or job. You can use rating worksheets to debug rate routines, validate that rates are as expected, or get detail showing why a coverage was rated as it was.
- **Impact testing** – See the impact that changing the rate book has on policy premium. Choose the policies on which to generate test policy periods by product, jurisdiction, producer code, postal code, effective date, expiration date, and as of date. Export the coverage and cost comparisons for each policy to Excel.
- **Multiple argument source sets** – The rate table definition can include more than one set of default argument sources. When you specify a rate table in a rate routine step, you can choose the argument source set.

For more information, see “Rating Management Concepts” in the *PolicyCenter Application Guide*.

## Generic Schedules

This release introduces generic schedules. *Schedules* are lists of items that define, extend, restrict, or exclude coverages or coverage terms. *Generic schedules* are instances of schedules that are implemented by using the generic schedule data model in PolicyCenter. Although it is possible to define schedules by individually implementing dedicated entities that are defined for each schedule type, Guidewire recommends that you add schedules to product lines using the generic schedule data model.

For more information, see “Generic Schedules” in the *PolicyCenter Application Guide*.

## Localization Improvements

More than 30 cases of hard-coded strings within PolicyCenter internal code, exported Gosu code, and PCF files were replaced with display keys to support localization.

## Logging API Changes

This release contains changes to the logging API. These changes are detailed in “Changes to the Logging API” in the *Upgrade Guide*.

## Base PCF File Changes

All links below require the `ReleaseNotes_files` directory on your local disk.

### PolicyCenter release 7.0.4 to 7.0.5

- To view a report of the changes in the base PCF files in the `modules/pc` directory, [click here](#).
- To view a report of the changes in the base PCF files in the `modules/p1` directory, [click here](#).

## Base Resources Changes

All links below require the `ReleaseNotes_files` directory on your local disk.

### PolicyCenter release 7.0.4 to 7.0.5

- To view a report of the changes in the base resources in the `modules/pc` directory, [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 Resource Portal.

## 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 meant to be a comprehensive list.

ID	Description
<b>Activities</b>	
PL-19637	Fixed an issue that caused the Activity Escalation work queue writer to create duplicate work items.
<b>Archiving</b>	
PC-14486	Fixed an issue where activities on the Desktop for jobs without an archived period appeared unavailable (dimmed).
PC-17491	Removed a <code>singleWhere</code> assertion from the default implementation of <code>PCArchivingPlugin</code> .
PL-19035	Added a <b>Download</b> button to the (Server Tools) Archive Info page to more easily enable the collection and sending of archiving error information.
PL-20041	Fixed an issue in which the archiving schema ( <code>archiving.XSD</code> ) did not match the archiving output XML.
<b>Consistency Checker</b>	
PL-16205	If you request a run of the database consistency checks using the <code>&lt;database&gt;</code> element attribute <code>checker="true"</code> (in <code>config.xml</code> ), PolicyCenter now prints a warning message in the log warning that this feature is deprecated. The message also instructs you to the <a href="#">Info Page</a> or command line to run the consistency checks as a batch job.
<b>Database</b>	
PL-18749	With this release, whenever a database upgrade begins, the upgrade process marks the database with the time the upgrade started and the host and batch server from which it began. At the time the database upgrade succeeds, the upgrade process removes the marker. If a second database upgrade begins before the first upgrade finishes, the second upgrade process detects the marker from the first upgrade process and throws an exception.  A second upgrade process can begin if the first upgrade fails and you attempt a restart without first restoring the database. Alternatively, a second upgrade process can begin if a second batch server begins a database upgrade before the first upgrade process completes successfully.
<b>Database Upgrade</b>	
PL-20127	Guidewire has corrected an issue with setting the <code>appscale</code> attribute for the <code>Money</code> data type. The <code>appscale</code> attribute worked for <code>currencyamount</code> columns, but not for <code>money</code> columns. The setting, specified in <code>datatypes.xml</code> , was being ignored when a column of type <code>money</code> was retrieved. This setting is now working. The <code>appscale</code> attribute controls the number of digits shown to the right of the decimal point for <code>money</code> or <code>currency</code> in single currency mode. It is similar to the <code>scale</code> attribute. The <code>appscale</code> attribute must be smaller than the <code>scale</code> setting. If defined, the <code>appscale</code> attribute overrides the <code>scale</code> attribute.
PL-20200	Guidewire has implemented optimization on the database upgrade. This optimization will improve the performance of new database creation, as well as database changes during minor and major upgrades.
<b>Document Management</b>	
PL-19958	Guidewire template documents using the Microsoft Word formats could not be generated if Microsoft Word was already open. The Guidewire Document Assistant ActiveX controls have changed versions from 2.0.49/2.1.49 (whitelist) to 2.0.52/2.1.52. Customers that manually deploy the control(s) will need to do so again.
PL-20275	Guidewire Document Assistant no longer adds extra section breaks to generated Microsoft Word documents.
PL-20472	Previously, if errors occurred while saving a document, the user was not informed and changes were lost. Now, the user receives an error message.
<b>Entities/Metadata</b>	
PL-19655	Fixed a problem with columns defined as <code>oneToOne</code> that disregarded <code>nullOk = false</code> . Now, PolicyCenter prevents you from committing empty <code>oneToOne</code> columns that do not allow nulls.

ID	Description
PL-20127	Guidewire has corrected an issue with setting the <code>appscale</code> attribute for the Money data type. The <code>appscale</code> attribute worked for currencyamount columns, but not for money columns. The setting, specified in <code>datatypes.xml</code> , was being ignored when a column of type money was retrieved. This setting is now working. The <code>appscale</code> attribute controls the number of digits shown to the right of the decimal point for money or currency in single currency mode. It is similar to the <code>scale</code> attribute. The <code>appscale</code> attribute must be smaller than the <code>scale</code> setting. If defined, the <code>appscale</code> attribute overrides the <code>scale</code> attribute.
<b>Forms</b>	
PC-17283	Fixed an issue in the Commercial Auto line where forms inference did not work.
<b>Geocoding</b>	
PL-1461	Two parameters are added to the plugin registry for the Geocode plugin. The <code>geocodeDirectionsCulture</code> parameter specifies the locale for geocoded addresses and routing instructions returned from a geocoding and routing service. For example, use the locale code <code>ja-JP</code> for addresses and instructions for Japan. The <code>imageryCulture</code> parameter specifies the language for map imagery. For example, use the language code <code>ja</code> for maps labeled in Japanese.
<b>Infrastructure</b>	
PC-16920	Fixed an issue where PolicyCenter threw a stack trace when searching for a producer on the Organization Search screen.
PC-17022	Guidewire has lowered the default maximum heap size for the QuickStart server from 1280 MB to 1024 MB to reduce the possibility of errors during the initialization of the JVM when starting the QuickStart server. The maximum heap size for the QuickStart server is controlled by the value of the <code>com.guidewire.commons.jetty.GWServerJettyServerMain.xmx</code> property in <code>memory.properties</code> .
PC-17446	Fixed an issue with diff in a policy change job that displayed the error <code>Cannot find DisplayKey for</code> in the Policy Review page when removing the schedule item and coverage.
<b>Lines of Business</b>	
PC-17223	The implementation of the <code>canSafelyDeleteBuilding</code> method in the <code>PolicyLineMethods</code> interface has been changed to take advantage of performance tuning that was done on the commercial property line. This change affects CPLine, BOLine and IMLine.  If you want to take advantage of the performance improvements, you must implement the following methods: <ul style="list-style-type: none"><li>• <code>getAllLineBuildingsEver</code> - Returns all <code>LineSpecificBuildings</code> in current and any future slices</li></ul>
Example for CPLine:	
<pre>_line.VersionList.CPLocations.arrays&lt;CPBuildingVersionList&gt;("Buildings").allVersions     Flat&lt;CPBuilding&gt;() as List&lt;LineSpecificBuilding&gt;</pre>	
<ul style="list-style-type: none"><li>• <code>getCannotDeleteBuildingMessage</code> - Returns a message when not able to safely delete current building</li><li>• <code>getCannotDeleteBuildingFutureMessage</code> - Returns a message when not able to safely delete future building</li></ul>	
<b>Note:</b> The <code>mergeDuplicateAdds</code> method in <code>QuoteProcess</code> has been moved after the <code>OOSJob</code> check because it should only be called in case of an out-of-sequence job.	
<b>Localization</b>	
PC-16274	Fixed an issue where system table verification raised an error if a localized field had a value specified for a locale other than default. This was previously a known issue.
PC-16527	Fixed an issue where the server did not start when the <code>appscale</code> or <code>scale</code> value was less than 2 in <code>datatypes.xml</code> . In addition, the <code>PolicyPeriod.DepositCollected</code> default value is now set to 0 instead of 0.0.
PC-16594	Numerous hard-coded strings in PolicyCenter internal code, exported Gosu code, and PCF Files were replaced with display keys to enable localization to non-English-speaking locales.
<b>Logging</b>	
PL-14764	Added the <code>UserInterface.Performance</code> logging category for tracing and debugging issues in the user interface. You can configure the logging category to include the ID of the user.
<b>Messaging</b>	

ID	Description
PL-18715	The Studio Messaging editor contained hard-coded strings for destination names, which made localization of destination names shown in the Admin tab difficult. Now, destination names are display keys. Therefore, destination names are included in the translatable resources of the application. Display keys for destination names are in the Java.MessageDestination level of the display key namespace.
<b>Miscellaneous</b>	
PC-16531	Fixed a previously known issue where PolicyCenter threw a null pointer exception in certain situations where questions had been configured to be filtered by the responses to other questions.
PC-16794	Fixed a problem in the user interface where valid customer configuration would cause the policy period to remain in Quoting status after quoting appeared to be complete. Future attempts to edit the submission would result in the message, "The PolicyPeriod you are trying to change is currently being quoted by another user, and cannot be modified at this time."
PC-16927	Fixed an issue where Policy Search and Policy Search APIs were returning too many PolicyPeriods and matching deleted contacts that were not on the current policy. Policy search now only returns the most recent version of a contact for each term, and matches data from the most recent version.
PC-17133	Fixed an issue where a Conversion on Renewal job did not mark contacts for syncing when bound. Added a call to Account#markContactsForAutoSync to RenewalProcess#unconditionalIssueRenewal to ensure that contacts added to a Policy during a Conversion on Renewal job will be marked to sync with an external address book.
<b>Product Integrations</b>	
PC-16938	Previously, in the integration between PolicyCenter and ContactManager, the ABContactSystemPlugin failed to parse double byte characters. The integration now works correctly with double byte characters.
PC-17397	There was an issue in Billing Center integration with Contact Manager that occurred when sending a new contact to BillingCenter from PolicyCenter when ContactManager was integrated with both applications. Previously, an exception occurred in ContactManager after BillingCenter committed the contact that PolicyCenter sent.
<b>Queries</b>	
PL-19590	Fixed an issue that caused an error in the PolicyCenter user interface by a query that contained a column definition twice.
<b>Rating</b>	
PC-14997	For consistency, changed CPSysTableRatingEngine, GLSysTableRatingEngine, and IMSysTableRatingEngine to be parameterized using their corresponding ProductModel types rather than the entity types. Now the SysTable rating engine uses ProductModel type for all products in the base configuration. This is the preferred model for new lines of business.
PC-16073	When you add a variable in a rate routine, the name must be a valid Gosu variable name and cannot be the same as an in-scope object, such as BaseRate.
PC-16729	In certain corner cases, sorting rate books by effective date resulted in an incorrect order. A small change to a PCF file fixed this issue.
PC-16781	Fixed an issue where small rate tables that were loaded in memory where those rate tables contained an ExactMatchOp matched a null input value.
PC-17023	When you import a rate book from XML by using the Import from XML screen, PolicyCenter now provides the option to display log messages, if any, in addition to writing them to the log.
PC-17024	Guidewire now provides two rate routines that implement the same driver assignment logic in the personal auto line. <ul style="list-style-type: none"> <li>• PA Assign Driver Style 1 permits the end user to choose the type of assignment without controlling the details of how the assignment is made.</li> <li>• PA Assign Driver Style 2 exposes the individual levels of details to the end user, enabling more control of how the assignment is made.</li> </ul>
PC-17025	Guidewire has changed commercial property rating as follows: <ul style="list-style-type: none"> <li>• Moved rating code within CPRatingEngine.gs. Moved the utility functions to the end of the file.</li> <li>• Added multiple argument source sets</li> <li>• Used these source sets in rate routines</li> </ul>

ID	Description
PC-17122	<p>The sample rating for Personal Auto now includes an example of automatic rate capping on renewal. If the cap is active, the capped value is stored as an <code>OverrideTermAmount</code> on the capped cost. To trigger the cap, you must rate the renewal with a newer rate book that changes the rates for Comprehensive and/or Collision.</p> <p>Previously, all override values were considered manual overrides. To distinguish a capped value from a manual override, PolicyCenter now has a new typelist called <code>OverrideSourceType</code> as well as a field of this type, called <code>OverrideSource</code>, on Cost and CostData. The existing enhancement <code>Cost#hasOverride</code> now returns true only if the Cost contains an override value AND the <code>OverrideSourceType</code> value corresponds to a manual override.</p>
PC-17264	Modified the <code>RateRoutinePlugin</code> implementation to delegate to line-of-business-specific classes making it easier to implement a line of business extension.
PC-17324	Fixed an issue with the typekey for the parameter. The <code>Name</code> on the <code>CalcRoutineParamName</code> typelist in Studio becomes the label <code>Code</code> in the <code>Parameter Sets → Parameters</code> tab in PolicyCenter. They are now <code>Name</code> in both places.
PC-17410	Fixed an issue where adding then deleting a parameter set left the <code>Parameter Sets</code> screen in edit mode.
PC-17442	When editing rate routines in the <code>Select a Policy Data Field</code> screen, the fields are now listed in alphabetic order.
<b>Revisioning</b>	
PC-17055	Fixed an issue in <code>PolicyPeriodBaseEnhancement</code> where the wrong policy period status was derived if the system testing clock had been changed. The Java date was being used instead of the Guidewire date utility.
<b>Studio IDE</b>	
PC-17486	<p>Fixed an issue where Studio threw an exception if <code>config.xml</code> contained an env specifier for the database element. The exception was:</p> <pre>[java] java.lang.RuntimeException: java.lang.RuntimeException: No appropriate database found in configuration: env = 'null' [java] at com.guidewire.studio.shell.StudioDependenciesManager.startDependencies (StudioDependenciesManager.java:201)</pre>
PL-19466	Fixed a problem that occurred when you dragged a rule to the top of a rule set. Guidewire Studio incorrectly overwrote the first rule instead of inserting the lower order rule above it. Now Studio inserts the lower order rule correctly above the top rule.
PL-19510	Fixed an issue that caused errors in the Guidewire Studio <code>Find in Path</code> functionality when searching rule resources if the resources contained files that were not proper Gosu rule resources. Proper Gosu rule resources contain Gosu code and end with the file extensions <code>.grs</code> or <code>.gr</code> . Now, <code>Find in Path</code> displays non-Gosu rule resources as errors without affecting search results for valid resources.
PL-19568	Guidewire Studio now supports 64-bit Java Virtual Machines.
<b>Upgrade</b>	
PC-16651	Fixed a previously known issue which occurred after upgrading from PolicyCenter 4 or earlier to PolicyCenter 7.0.0 or later. Previously, PolicyCenter threw an error when starting a production server.
PC-17671	Fixed an issue that caused database upgrades to fail when the <code>WorkFlowWorkItem</code> table was populated. (Upgrade succeeded when the table was empty.) The fix involved changing the <code>WorkFlowWorkItem</code> entity attribute from Final to Extendable.
<b>User Tools</b>	
PC-17047	A policy search no longer matches on a <code>PolicyContactRole</code> that has been deleted.
PC-17137	Fixed an issue where all currently open activities were visible when opening any policy. Now only activities related to the selected policy's account are visible.
<b>Utilities</b>	

ID	Description
PL-11681	<p>The Gosu StringUtil class used many Perl 5, version 3, regular-expression, syntax-compatible functions to search for and replace strings. These functions were implemented by the Apache/Jakarta ORO library, which has been retired. See <a href="http://jakarta.apache.org/oro/">http://jakarta.apache.org/oro/</a> for details.</p> <p>The same functionality for regular-expression, syntax-compatible search and replace of strings is now available in the <code>java.util.regex</code> package and the <code>java.lang.String</code> class.</p> <p>Before this change, you used the following coding pattern:</p> <pre>StringUtil.substitute(inputString, "s/" + regexString + "/" +     replacementString + "/g")</pre> <p>Now, instead, you use the following coding pattern:</p> <pre>var pattern = java.util.regex.Pattern.compile(regexString) var patchedString = pattern.matcher(inputString).replaceAll(replacementString)</pre> <p>Simpler search and replace functions are also available, such as:</p> <pre>java.lang.String.replace(String, String).</pre>
<b>Web – UI/Runtime</b>	
PL-15215	In previous releases, the JAWS screen reader picked up hidden links. Now, hidden links have the <code>aria-hidden</code> HTML attribute to skip hidden links. This fix works with JAWS 13 and IE8 or higher.
PL-19646	Fixed an issue preventing the Guidewire application shortcut keys from working as intended. Previously, the keyboard responded only to the shortcut keys of the browser. Now the keyboard responds to the Guidewire application shortcut keys.
PL-20150	Modified PolicyCenter events processing logic to skip events for removed widgets. For example, if the values of two widgets have been changed in the same form submission, there will be two <code>onChange</code> events in the event queue. If the handler of the first event implicitly removes the widget that is the source of the second event, than the second event will no longer be processed (meaning that the associated <code>onChange</code> handler will not be invoked to avoid errors).
<b>Web Services - WSI</b>	
PL-18729	It is now possible to invoke local web services using the <code>wso.local</code> mechanism from Studio without a running server.
PL-20174	It is now possible to run WSI Web Services on a server other than the batch server in a cluster.
<b>XML Element (and XSD types)</b>	
PL-20076	Fixed an XML parsing bug that caused an <code>xs:choice</code> with a zero-width match to not be handled properly.

## Known Issues and Limitations

This section describes known issues with this release of Guidewire PolicyCenter.

- PolicyCenter Known Issues
- Studio/Platform Known Issues

**Note:** For maintenance releases, Guidewire often defers fixing configuration issues if they 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

#### [Studio: “Verify Changed Resources” is not implemented in the product model \(PC-6928\)](#)

**Issue** – The Studio Tools → Verify → Verify Changed Resources command does not work with the PolicyCenter product model.

**Workaround** – Use one of the other verification commands instead. See “Validating Studio Resources” in the *PolicyCenter Configuration Guide* for details.

## 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.

## 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. 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 original answer is still saved along with the other answers.

**Workaround** – The answers to these *irrelevant* questions 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)
        }
    }
}
```

## 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 screen of the agent
- 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.

### Attachment inclusions are not copied across terms (PC-14561)

**Issue** – RI attachment inclusions are not copied across terms, such as during a renewal or a rewrite.

**Workaround** – Guidewire is aware of this issue.

### Contact.AccountHolderCount incorrectly counts the number of accounts (PC-14736)

**Issue** – Contact.AccountHolderCount incorrectly counts the number of accounts a particular contact is the account holder on.

**Workaround** – Query the database to count the accounts on which this contact has the account holder role.

### Contacts created from the Contact tab are not autosaved (PC-14761)

**Issue** – Contacts created from the Contacts tab are not getting autosaved. The **UnsavedWork** alert correctly goes on, but when the user clicks elsewhere, the work is lost.

**Workaround** – Set the `autosaveable` attribute on the `NewContact.pcf` to `true`.

### 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.

### 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 Are 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.

### View Progress button in Archive Info throws exception (PC-15843)

**Issue** – If a user with access to the Server Tools administrative section attempts to View Progress of archiving, PolicyCenter may throw a null pointer exception.

**Workaround** – Guidewire is aware of this issue.

### Renewal processing for renewal conversion assumes BasedOn will be 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 new `gw.webservice.pc.pc700.job.startConversionRenewal` method.

### Archived policies are not displayed in the Policies tab for AccountContact file (PC-15917)

**Issue** – On the Account File Contacts screen, archived policies are not appearing in the Associated Policies and Associated Work Orders tabs. On the Account File Locations screen, archived policies are not appearing in the Policy Terms and Work Orders list views. This is because PolicyCenter deleted the records in the active database used to identify to these entities when the policy was archived.

**Workaround** – You can configure PolicyCenter to retain the information about the archived policies in a way that it is available to the searches in the affected screens.

### 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 cancellation, 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 **Non Renew** or **Not Take** the renewal. 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.

### Term number is set to null while creating a new branch for preemption (PC-16868, PC-16869)

**Issue** – The term number is incorrectly set to null in the following cases:

- A policy change has been preempted by another policy change.
- You edit the effective date on a policy change.

**Workaround** – To handle the first case, make the following changes to the `handlePreemptions()` method in `JobProcess.gs` to save a copy of the `termNumber` and then set it to the value saved:

```
function handlePreemptions() : ApplyChangesResult {  
    // NEW variable to save the termNumber  
    var termNumberCopy = _branch.TermNumber
```

```

canHandlePreemptions().assertOkay()

var newBranch = _branch.createNewBranchForPreemption()

// give JobProcess subclasses a chance to have their say
processSpecificPreemptionHandling(newBranch)

//New method to copy the saved termNumber to the newBranch
if (newBranch.TermNumber == null) {
    newBranch.TermNumber = termNumberCopy
}

var conflicts = applyChanges(newBranch)
// edit runs after applyChanges so that it affects beans added by applyChanges
newBranch.edit()

return new ApplyChangesResult(newBranch, conflicts, _branch.EditEffectiveDate)
}

```

To handle the second case, make the same change in the `cloneBranchForEditingEffectiveDate()` method in `PolicyChangeProcess.gs`.

### [Exception in impact testing when creating baselines and canceling job \(PC-17592\)](#)

**Issue** – In impact testing, PolicyCenter throws an exception after canceling the job that creates baseline policy periods. This exception happens when the database is the SQL server.

**Note:** 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** and then click **Cancel Job** to cancel creation of baseline policy periods.
3. Click **Create Baselines** again.

PolicyCenter throws an exception.

There is a similar issue when you cancel quoting test periods by clicking the **Cancel Quote for Test Periods** on the **Testing Periods** screen.

**Workaround** – Start and stop the work queues.

1. In PolicyCenter, press **Alt+Shift+T** to access **Server Tools**.
2. Click **Work Queue Info** in the left sidebar.
3. Restart the **ImpactTestingTestRun** and **ImpactTestingPrep** work queues.

### [In the default implementation, PolicyCenter sends the wrong written date to BillingCenter \(PC-17593\)](#)

**Issue** – When PolicyCenter and BillingCenter are integrated, the PolicyCenter default implementation incorrectly sends the `PolicyPeriod.WrittenDate` to the BillingCenter `BillingInstructionInfo`.

**Workaround** – Configure PolicyCenter to send the first `Transaction.WrittenDate` to BillingCenter:

1. In Studio, navigate to **Classes → gw → plugin → bc700** and open **BillingInstructionInfoEnhancement**.
2. In the `startSyncBasicPolicyInfo()` method, change:

```

this.WrittenDate = period.WrittenDate.XmlDateTime
to
this.WrittenDate = period.AllTransactions?.first().WrittenDate.XmlDateTime

```
3. Restart PolicyCenter to deploy your changes.

## 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`.

**Note:** 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.

## When there are multiple slices in a rate, rating worksheets show only one slice (PC-17596)

**Issue** – When there are multiple slices in a rate, rating worksheets shows only one slice. Rating worksheets should show all the slices.

Rating worksheets is a feature of Guidewire Rating Management.

**Workaround** – Guidewire is aware of this issue.

## When only the length of a policy is changed, no rating worksheet is generated (PC-17604)

**Issue** – A policy change transaction that only changes the length of a policy does not produce a rating worksheet. PolicyCenter should show the prior rate worksheet that calculated the original term amount.

Rating worksheets is a feature of Guidewire Rating Management.

**Workaround** – Guidewire is aware of this issue.

## Exception when rerating commercial property after applying rating overrides on a policy change job that adds a new location (PC-17647)

**Issue** – An illegal state exception occurs when the following conditions are met:

1. You quote a policy change job that adds a new building to an existing location on an in-force commercial property policy.
2. After quoting, you override the **Term Amount**.
3. Click **Rerate**.

**Workaround** – Guidewire is aware of this issue.

## Studio/Platform Known Issues

### First time you click the arrow of the typekey input, the drop-down menu does not open (PL-10134)

**Issue** – The drop-down menu does not open on the first click of the arrow on a typekey input. Instead, the help text opens.

**Workaround** – Turn off help text on focus by setting `InputHelpTextOnFocus` to `false` in `config.xml`. After you do that, the help text shows only if you mouse over the input, and it does not interfere with opening a drop-down menu.

#### [Studio Rules do not use correct capitalization for root object's name \(PL-10740\)](#)

**Issue** – Rule set root objects are expected to begin with an upper-case letter and issues warning if a rule set root object does not.

**Workaround** – You can safely ignore the warning message.

#### [User interface cannot handle starting multiple instances of a batch process \(PL-12372\)](#)

**Issue** – The user interface cannot handle starting multiple instances of a batch process.

**Workaround** – To execute multiple instances of a batch process, start them from the command line. Also, to allow multiple instances to be run simultaneously, you must ensure that the `BatchProcess.isExclusive()` method returns `false`.

#### [Type system refresh after PCF page title change does not update corresponding menu label \(PL-13057\)](#)

**Issue** – The type system refresh after a PCF page title change does not update the corresponding menu label.

**Workaround** – After updating a page title, restart the server to refresh menu labels and avoid null pointer exceptions due to stale references.

#### [US-Locations.txt file with the US geodata from GreatData has special characters that cause validation problems with United States Postal Service \(USPS\) data \(PL-13384\)](#)

**Issue** – The `US-Locations.txt` file contains information that does not conform to United States Postal Service (USPS) standards for bulk mailings.

**Workaround** – The provided `US-Locations.txt` file is intended only for use in geocoding to identify addresses for a location. You can edit the `US-Locations.txt` file to conform to your particular address standards, and then import that version of the file instead.

#### [GX models that reference virtual fields and enhancements throw null pointers if null \(PL-13560\)](#)

**Issue** – GX models that reference virtual fields and enhancements throw null pointers when these fields and enhancements are null. While object path expressions are generally null safe, they are not null safe when the expression navigates to a virtual field or enhancement.

**Workaround** – Include null checks and error handling to prevent referenced virtual fields or enhancements that are null from causing null pointer exceptions. For example, the following object path navigates to a virtual field.

```
anActivity.ActivityStatus
```

Use the null-safe period operator instead of the standard operator, as the following example shows:

```
anActivity?.ActivityStatus
```

#### [Email with file attachment with unicode filename not correctly handed over to the mail server \(PL-13582\)](#)

**Issue** – An email with a file attachment that has a unicode file name is not sent to the mail server correctly.

**Workaround** – Use Latin characters for file names on attached files.

## 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.

## Cannot make a field from a delegate into a localized column (PL-13761)

**Issue** – You cannot make a field from a delegate into a localized column.

**Workaround** – Move the column to be localized off the delegate and onto each of the implementing entities. Then, to make the column appear as though it exists on the delegate, define an enhancement property on the delegate that *delegates* to the appropriate column, depending on the implementing entity.

## Problem with 'regen-java-api' command and JAR files (PL-16351)

**Issue** – If you run the `PolicyCenter/bin/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 into PolicyCenter using an administrative account.
2. Access Internal Tools → Reload.
3. Click Reload Workflow Engine.

## Build command 'verify-types' fails for 64-bit JVM (PL-19768)

**Issue** – Executing the `verify-types` command results in the following error on 64-bit versions of Java:

`java.lang.OutOfMemoryError`

**Workaround** – Use a 32-bit version of Java.

## Studio does not properly render tab characters (PL-20531)

**Issue** – Studio does not properly render selected text and caret positions in files containing true tab characters. This typically occurs with code that you paste into Studio from another IDE such as Eclipse.

**Workaround** – Start Studio with the following parameter:

`studio -Dstudio_disable_improved_font_rendering=true`

## Specifying env attribute in <database> generates run time exception (PL-20815)

**Issue** – If you do not define a default env specifier for the <database> element in config.xml, Studio generates the following exception and fails to start:

```
[java] java.lang.RuntimeException: java.lang.RuntimeException: No appropriate database found in
configuration: env = 'null'
[java] at com.guidewire.studio.shell.StudioDependenciesManager
.startDependencies(StudioDependenciesManager.java:201)
```

**Workaround** – Do one of the following:

- Provide a default env value in the database registry section of config.xml.
- Set a default env value by adding the following to the command to start Studio:  
`studio -Dgw.cc.env`

# Guidewire PolicyCenter 7.0.6 Release Notes

## Release 7.0.6

**Document Revision:** 20-May-2014

Copyright © 2001-2014 Guidewire Software, Inc. All rights reserved.

Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Reinsurance Management, Guidewire ContactManager, Guidewire Vendor Data Management, Guidewire Client Data Management, Guidewire Rating Management, Guidewire InsuranceSuite, Guidewire ContactCenter, Guidewire Studio, Guidewire Product Designer, Guidewire Live, Guidewire DataHub, Guidewire InfoCenter, Guidewire Standard Reporting, Guidewire ExampleCenter, Guidewire Account Manager Portal, Guidewire Claim Portal, Guidewire Policyholder Portal, ClaimCenter, BillingCenter, PolicyCenter, InsuranceSuite, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks, service marks, or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries.

All other trademarks are the property of their respective owners.

**This material is confidential and proprietary to Guidewire and subject to the confidentiality terms in the applicable license agreement and/or separate nondisclosure agreement.**

Guidewire products are protected by one or more United States patents.

## Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Support

- Issues and Major Changes
- Improvements and General Issues
- Known Issues and Limitations

## Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

---

**IMPORTANT** If you missed 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 7.0.6.

- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 3.0.4 or later or BillingCenter 7.0.1 or later.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 6.0.0 or later.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 7.0.0.

## 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 Resource Portal – <http://guidewire.custhelp.com>

## Issues and Major Changes

This section contains issues and major changes that may affect your installation.

- Quote Purging
- Free-text Search
- Rating Management
- Base PCF File Changes
- Base Resources Changes
- Changes in This Release Provided in Upgrade Diff Report

### Quote Purging

Quote purging is new in this release.

Over time, the PolicyCenter database accumulates quotes from jobs not resulting in bound policies and alternate policy periods created through multi-version quoting and side-by-side quoting. As time passes, these jobs and policy periods have little business value, increase database storage requirements, and slow response time. Quote purging removes these jobs and policy periods from the database.

Quote purging also removes orphaned policy periods, which are policy periods not associated with a job. Preempted jobs result in orphaned policy periods.

Quote purging provides batch processes to remove from the database these jobs, policy periods, and other objects. Quote purging is not an end user feature and is only accessed through **Server Tools**.

For more information, see “Quote Purging” in the *Application Guide*.

### Batch Processes to Run Prior to Running the Purge Batch Process

A policy period cannot be purged if it is associated with one or more workflows. To more effectively purge and prune policy periods, clean up workflows before purging or pruning by running the following batch processes:

- Purge Workflow
- Purge Workflow Logs
- Workflow

### Purging Test Tool

The **Purging Test** tool is provided to help you see the effect of purging or pruning on one job. Use this tool during development to see the effect of the Purge batch process on a selected job. This tool also has a link to run the Purge batch process.

To access this tool, press ALT+SHIFT+T and select **Internal Tools** → **Purging Test**.

### Flushing other work queues

A policy period cannot be purged if it is associated with one or more workflows. To more effectively purge and prune policy periods, clean up workflows before purging or pruning.

- In **Flush other work queues (Purge workflows, etc.)**, click **Run**.

This command cleans up workflows by running the following batch processes:

- Purge Workflow
- Purge Workflow Logs
- Workflow

### To purge or prune a job

1. Enter the work order number of the job you wish to purge.
2. Click **Find/Refresh Job** to fetch the job from the database.

PolicyCenter displays a **Purge or Prune Job** section. This section displays the following information about the job:

Field	Description
Job Number	The number of the job to purge or prune.
Subtype	The job subtype.
Close Date	The job close date.
DoNotPurge flag (Policy)	The value of the DoNotPurge flag on Policy.
Policy Period(s)	The number of policy periods associated with this job. The <b>Prune -- Purge Policy Period(s)</b> section displays more information about each policy period.
Coverage End	The coverage end date of the policy.

You can select to purge or prune the selected job.

3. To purge or prune the job without checking to see if the job is eligible for purging or pruning, select **Skip configurable Purging checks** or **Skip configurable Pruning checks**.

If you did not choose to skip checks, then the job is purged or pruned based on the current configuration, including the `PurgePlugin` implementation.

If you choose to skip checks, then the job is purged or pruned regardless of the **Subtype**, **Close Date**, or **Coverage End**.

If you choose to skip checks, a job can be purged or pruned only if the following are true:

- The job's `CloseDate` is not null.
- The `DoNotPurge` flag is not set on the policy associated with the job.
- The job has no open activities.
- The job is not archived.

Additionally, if you choose to skip checks, the following criteria also apply. For purging, each policy period on the job must meet this criteria. For pruning, the policy period being pruned must meet this criteria.

- The `PolicyPeriod` is not bound.
- The `DoNotPurge` flag on the `PolicyPeriod` is not set.
- The `PolicyPeriod` has no workflows.

4. Click **Purge Job** or **Prune Job** to purge or prune the selected job.

PolicyCenter does not run the Purge batch process, but performs equivalent actions to purge or prune the selected job. The purge or prune accesses the configuration parameters, and runs the `PurgePlugin` implementation.

### To prune policy periods

After selecting a job, the policy periods attached to the selected job appear in the **Prune -- Purge Policy Period(s)** section. This section displays the following information about each policy period:

Field	Description
Policy Period	The version number of the policy period, and whether it is the selected policy period.
Period Start	The policy period start date.
Period End	The policy period end date.
Period Status	The policy period status.
DoNoPurge (period)	The value of the <code>DoNotPurge</code> flag.
Flip DoNotPurge bit	A button to toggle the <code>DoNotPurge</code> flag on this policy period.

1. To prune policy periods without checking to see if the policy period is eligible for pruning, select **Skip checks for purging policy period**.

If you choose to skip checks, then the policy period is pruned regardless of the **Subtype**, **Close Date**, or **Coverage End**.

If you did not choose to skip checks, then the policy period is pruned based on the current configuration, including the `PurgePlugin` implementation.

2. Click **Purge** in the row of the policy period you wish to prune.

The **Purge** button is enabled for unselected policy period versions.

PolicyCenter does not run the Purge batch process, but performs equivalent actions to prune the selected policy period. The purge or prune accesses the configuration parameters, and runs the `PurgePlugin` implementation.

### To run the purge batch process

In Run Purge Batch Process, click **Run**. This is the same as running the batch process from the **Server Tools → Batch Process Info** screen. The batch process purges jobs and prunes policy periods in the PolicyCenter database.

## Free-text Search

PolicyCenter includes two types of searches: basic and advanced. Basic search is new in this release.

Basic search is a free-text search for quick access against very large databases. Free-text search also provides exact and inexact matching. Inexact matching returns results that partially match, are synonyms, and sound-like the search criteria. In PolicyCenter, free-text search uses an integration with the full-text search engine Solr. PolicyCenter includes basic search for policies and submissions.

Advanced search uses database search, which directly searches the PolicyCenter database. PolicyCenter includes advanced search for policies, jobs, accounts, producer codes, activities, and contacts. For large data sets, advanced search can take longer than basic search.

For more information about basic search, see “Basic Search” in the *Application Guide*.

## Rating Management

This topic contains additional information about Guidewire Rating Management.

### Variables Available in a Rate Routine

In a rate routine step, PolicyCenter filters the list of available variables to choose from when another parameter contains one of the types. For example, an employee may contain a `ClassCode` variable, which in turn contains a `Code` variable. Normally, there would be options for `Employee.ClassCode` and `Employee.ClassCode.Code`. However, if another parameter is added of type `ClassCode`, then only `Employee.ClassCode` appears in the list. This significantly reduces the list of types to choose from for certain high level entities, like `PolicyLine`. If any of those sub-variables are desired, then the `Employee.ClassCode` variable can be passed in as a parameter to be used explicitly.

If you have parameters such as `Employee` and `ClassCode`, where the `ClassCode` variable is not the same as the one inside `Employee`, then the `ClassCode` fields on `Employee` are hidden and not usable. You can explicitly pass the `ClassCode` from `Employee` as a parameter, which can then be used directly.

## Base PCF File Changes

All links below require the `ReleaseNotes_files` directory on your local disk.

### PolicyCenter release 7.0.5 to 7.0.6

- To view a report of the changes in the base PCF files in the `modules/pc` directory, [click here](#).
- To view a report of the changes in the base PCF files in the `modules/p1` directory, [click here](#).

## Base Resources Changes

All links below require the `ReleaseNotes_files` directory on your local disk.

### PolicyCenter release 7.0.5 to 7.0.6

- To view a report of the changes in the base resources in the `modules/pc` directory, [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 Resource Portal.

## 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

ID	Description
<b>Archiving</b>	
PC-15843	Fixed an issue where the <b>View Progress</b> button in <b>Server Tools</b> → <b>Info Pages</b> → <b>Archive Info</b> threw a null pointer exception.
PC-15850	Made changes to the <b>PolicyFile</b> and <b>JobWizards</b> PCF files to make them safe for archived policy terms.
PC-17527	Now an archived policy period will not be fetched. This avoids consistency check failures in the database.
PC-17677	Modified the <b>PCHibernateSourcePlugin</b> to avoid memory issues using a <b>HashMap</b> . The <b>DatabasePurgeAction</b> type has been replaced by the simpler <b>PCDeleteBuilder</b> type.
PC-18032	PolicyCenter has two new batch processes: <ul style="list-style-type: none"> <li><b>PurgeWorkflows</b> – This workflow has a <b>WorkflowPurgeDaysOld</b> configuration parameter.</li> <li><b>PurgeMessageHistory</b> – This workflow has a <b>KeepCompletedMessagesForDays</b> configuration parameter.</li> </ul>
PC-18813	Added a new <b>DisableDomainGraphSupport</b> configuration parameter. All PolicyCenter environments are expected to have valid data models even if they are not currently using a feature that requires domain graph support. These features are archiving and quote purging. In rare situations, you may have an upgrade that does not yet have a valid data model. If there are compelling business reasons not to correct the data model during the upgrade, you can disable domain graph support using the <b>DisableDomainGraphSupport</b> parameter. If you set this parameter to true, certain features, including archiving and quote purging, will fail ungracefully. Guidewire strongly recommends that you resolve domain graph issues at the earliest opportunity to minimize the upgrade cost of the changes required to achieve a valid data model. Default: false
<b>Financials</b>	
PC-18297	Fixed issue where you could not withdraw a job with an overridden quote.
<b>Forms</b>	
PC-17832	Made performance improvements to form inference.
<b>Integration</b>	
PC-17232	Fixed an issue with the integration between PolicyCenter and BillingCenter. A scheduled cancellation can now be rescinded even if the cancellation has been preempted.
PC-17943	Previously in the PolicyCenter to BillingCenter integration, PolicyCenter would incorrectly set the value of <b>BillingInstructionInfo.WrittenDate</b> with <b>PolicyPeriod.WrittenDate</b> before sending it over to BillingCenter. This has been corrected to set the value of <b>BillingInstructionInfo.WrittenDate</b> with <b>WrittenDate</b> of the first transaction on the policy period.
PC-18138	Contacts in PolicyCenter and ContactManager now synchronize correctly after a contact merge in ContactManager.
<b>Internationalization</b>	

ID	Description
PC-17847	Fixed an issue where Studio could not synchronize the product model. This occurred in double-byte versions of Studio.
PC-18120	Fixed an issue with entering rate table factor rows with rate tables containing decimal parameters or factors in a locale that uses a decimal delimiter other than a period.
PC-18216	Replaced a number of hard-coded strings in PolicyCenter internal code, exported Gosu code, and PCF Files with display keys to enable localization to non-English-speaking locales.
PC-18388	Fixed localization specific encoding in the ABContactSystemPlugin plugin.
<b>Jobs</b>	
PC-14525	Duplicate work items are no longer added to the queue in the following work queues: <ul style="list-style-type: none"><li>• ArchivePolicyTermWorkQueue</li><li>• AuditTaskMonitorWorkQueue</li><li>• BoundPolicyExceptionWorkQueue</li><li>• ClosedPolicyExceptionWorkQueue</li><li>• ImpactTestingExportWorkQueue</li><li>• ImpactTestingTestPrepWorkQueue</li><li>• ImpactTestingTestRunWorkQueue</li><li>• JobExpirationWorkQueue</li><li>• OpenPolicyExceptionWorkQueue</li><li>• OverduePremiumReportWorkQueue</li><li>• PolicyHoldJobEvalWorkQueue</li><li>• RestorePolicyTermWorkQueue</li><li>• RetireActivityWorkQueue</li></ul>
PC-16868	While creating a new branch for preemption, the term number will not be set to null. The term number will retain the value from the based on period.
PC-16869	When the effective date is edited in a policy change, the term number will not be set to null. The term number will retain the value from the based on period.
PC-17258	The javadoc of account.createConversonRenewalWithBasedOn now includes a description of what occurs if the policy number already exists in the system.
PC-18015	Previously, when switching the billing method away from direct bill then back to direct bill, the down payment, installment, and total columns were not repopulated. This change adds a call to repopulate the data when switching back to the direct bill billing method.
PC-18187	When doing a side-by-side quote, if a Personal Auto policy period contained more than one vehicle, the line level coverage costs only displayed the cost for one of the vehicles. With the fix the sum for each line level coverage cost is displayed.
PC-18252	Fixed an issue where PolicyCenter threw an exception when entering the Side-by-side Quoting screen from a mid-term policy change with a quoted initial period. This was caused by items in the page's wizard being out of sync with items being created as part of side-by-side. The fix ensures that the bundles are correctly kept in sync.
PC-18352	In Studio, the Entity Names → PolicyAddress definition incorrectly had two names with bean path PolicyAddress.AddressLine1. The second occurrence was changed to PolicyAddress.AddressLine2.
<b>Line of Business</b>	
PC-18208	In PolicyCenter 7.0.5, the Coverable property was added on every cost type but was not always instantiated. To correct this issue, the OwningCoverable property was added and points to the correct coverable. The Coverable property is defined as OwningCoverable.
PC-18348	All costs in Commercial Auto contain a nullable foreign key to vehicle. However, BusinessVehicleCovCostData did not populate the foreign key to vehicle. This caused some costs sub-types, which are associated with a vehicle, to have a Vehicle property with a null value. To correct this issue, SysTable Rating and the BusinessVehicleCovCostData were updated.
PC-18632	Corrected an issue in Worker's Compensation that was created in PC 7.0.5. If Rating Overrides were done, pressing Clear All should reset the overrides. But as a result of the issue, a not null error occurred in the database and the overrides were not cleared. This fix corrects this issue.
<b>Miscellaneous</b>	

ID	Description
PC-17426	Fixed an issue where objects which implement ScheduleItem and which fill in the NamedInsured foreign key were being created incorrectly. This occurred if there were out-of-sequence changes that involved changing the Primary Named Insured and adding Schedules in lines of business such Businessowners or General Liability.
PC-17645	Fixed an issue where the PolicyCenter ready message appeared before the user could log in.
PC-17646	The Activity forward page only allows users with sufficient permissions to view the details of an activity. PolicyCenter now gracefully handles the case where the user has insufficient permissions.
PC-17897	Previously, there was no producer code security when viewing notes, activities and documents. Now, notes, activities, and documents are filtered based on producer code of service when producer code security is enabled.
PC-18177	PolicyCenter now pre-loads all files before users can log in. Previously, users could experience delays when accessing parts of the system that had not been loaded.
PC-18456	Fixed an issue which caused a locked branch error on Contact Merge.
PC-18466	Fixed an issue with the entity runtime interface, which is used to calculate the runtime type of entity arrays. Previously, there was a problem with the runtime interface if the entity implemented a delegate entity that in turn implemented a Gosu interface.
PC-18662	Fixed an issue where PolicyCenter threw a class cast exception when accessing a custom Java method requiring external entity conversion.
<b>Policy File</b>	
PC-16986	A future dated policy causes a null pointer exception if you start a submission in the future and bind it, then go to the submission work order and back to the policy file. This was caused because policy forwards from submission or any job wizard page did not include the effective date.
<b>Product Model</b>	
PC-18009	When setting direct coverage term values, the values are now rounded based on the scale of the database column assigned for the coverage term. For example, suppose that xxxCov.eti definition has 2 columns: DirectTerm1 (scale = 6) and DirectTerm2 (scale = 0). Further, suppose that there is coverage defined in the product model with two direct coverage terms, and TermA is assigned to the column DirectTerm1 and TermB is assigned to DirectTerm2. In PolicyCenter, when the user enters the values for those coverage terms, TermA is rounded to 6 positions after the decimal, and TermB is rounded to zero decimal positions. The rounding method for both is HALF_EVEN.
<b>Rating</b>	
PC-16815	When importing values into a rate table, if the scale or precision exceed the scale or precision supported by the target rate table, PolicyCenter rounds the values and displays a warning message.
PC-17116	Fixed an issue that caused a null constraint violation. Rate Routine now has a similar Language list view similar to the Rate Book screen.
PC-17489	Previously, the CalcStepDefinition entity had three fields of type text with unspecified size. These fields were Description, Notes, and SectionComment.  This change defines Description as a shorttext field, Notes as a mediumtext field, and SectionComment as a mediumtext field.
PC-17533	Previously, the sample data in the PA_COVERAGE_BASE_RATE table did not include entries for the following vehicle coverages: <ul style="list-style-type: none"><li>• PACollision_MA_MI_Limited</li><li>• PAExcessElectronicsCov</li><li>• PATapeDiscMediaCov</li></ul> In the default configuration with sample data loaded, policies with these coverages could encounter an exception. With this change, sample data is populated for these coverages.
PC-17570	Previously, if the Impact Testing batch process was still running when user tried to create another batch job, PolicyCenter threw an evaluation exception.
PC-17592	In Impact Testing, if you canceled the creation of either baseline or test periods, and then quickly pressed the Create button again, you got an exception. Impact Testing now displays a message that the cancel request is being processed. You must wait to start a new creation job.

ID	Description
PC-17595	Made improvements to performance of quoting when Rating Management is enabled. When exporting values from a rate table into Excel, if the precision exceeds 15, PolicyCenter displays an error message that the export values exceed the precision supported by Excel.
PC-17596	Previously, when there were multiple slices in a rate, the rating worksheet showed only one slice. Rating worksheets now show all the slices.
PC-17604	Previously, a policy change transaction that only changed the length of a policy did not produce a rate worksheet. PolicyCenter now shows the rate worksheet used to calculate the original term amount.
PC-17647	An illegal state exception occurred when rerating commercial property after applying rating overrides on a policy change job that adds a new location.
PC-17649	Previously, clicking <b>Validate</b> on a rate routine did not flag referencing a local variable before defining it. Now, PolicyCenter displays a validation message if a local variable is referenced before it is defined.
PC-17678	Previously, you could not edit newly added parameters when the rate routine parameter set was used in a rate table definition without specifying a default argument source. Now you can edit these parameters.
PC-17682	Previously you could edit or delete a rate routine parameter even if it was used as a function call argument in a rate routine. Now, you cannot edit or delete such a parameter.
PC-17685	During rate table import, an invalid row caused import to stop at the error. Import will now continue past invalid rows allowing valid rows to be successfully imported.
PC-17692	Fixed an issue that occurred if you made a policy change which removed a vehicle when rating worksheets were enabled. Previously, the removed vehicle would not show up in the rating worksheets, even for the date range where it was still on the policy.
PC-17706	When doing a database query, a comparison between BigDecimal (Double) and Integer is now possible.
PC-17717	The <b>CostData.gs</b> file contains a check against amounts computed using a release before PolicyCenter 7.0.2 or 4.0.6. This check assures that on a policy change, if the item being rated had not changed, the new prorated amounts were the same as the computed amounts from the prior release. This avoids the amounts being slightly off due to rounding, for example.  This check could potentially slow down the overall rating response time, even if external rating is used.  This check is now disabled by default. If you are upgrading from a release prior to 7.0.2 or 4.0.6, you can enable this check by updating the <b>CostData.GuardAgainstRoundingChange</b> property.
PC-17718	Other costs were not included in rating worksheets. In the default configuration, examples of other costs include multi-policy discount and cancel short rate penalty. Other costs are now included in rating worksheets.
PC-17754	Fixed an issue where the <b>Impact Testing Results</b> page rendered poorly and displayed an <b>Error Occurred</b> message if the graphed result page was refreshed after the data it was displaying expired or contained all failures. This page now display a <b>No valid data points</b> message when this occurs.
PC-17767	Fixed an issue which caused an exception when doing rate table lookups.
PC-17785	Fixed an issue in the Impact Testing wizard. Previously, if you clicked <b>Cancel Job</b> on the <b>Create Baseline</b> screen, the updated screen did not display the <b>Create Baseline</b> button.
PC-17807	Fixed an issue where impact testing batch job errors were truncated in the log file.
PC-17844	In PolicyCenter 7.0.3 if the <b>RateTableDefinition</b> table was populated but the <b>EntityName</b> column was unspecified, then upgrading to 7.0.5 caused the data in the table to be erased. The upgrade can now happen without losing the data in the table.
PC-17898	Previously, two Parameter Sets with the same name but different lists of parameters could interfere with each other's contents. This appeared in the Rate Routine editor and the Argument Sources tab of the Rate Table Definition editor.
PC-17912	In Impact Testing, when running <b>Quote Test Periods</b> or <b>Requote Test Periods</b> , the drop-down list in the <b>Impact Results</b> page was not updated from the previous results. This issue has been fixed.
PC-18039	Rating sample data is now loaded only if rating licensed.
PC-18309	Fixed an issue with accessing the subfields of a <b>ClassCode</b> entity when editing a rate routine. Formerly, this worked for some <b>ClassCode</b> objects and not others, depending on how they were defined.
PC-18310	Fixed an issue with rate routines that caused a failed lookup. This occurred when doing an exact search for a floating point constant against a decimal field.

ID	Description
PC-18386	Fixed an issue where rate routines that referenced a rate factor table which return a factor of type date failed during execution with a class not found exception.
PC-18452	Previously, the Reset button did not completely clear all fields on the Impact Testing → Choose Policies screen.
<b>Rating Localization</b>	
PC-15050	You can now define variable names containing unicode characters if the name meets the Java identifier character requirements defined in the JDK. See <code>java.lang.Character.isJavaIdentifierPart()</code> .
PC-17951	Previously when editing rate routines, users were required to enter numeric constants using a comma as the grouping separator and a period as the decimal separator regardless of locale. Now, numeric values are interpreted using the current locales number format. Entry and use of constant values must be input and used in a single locale.
PC-18002	When exporting a rate book with multi-byte characters in the filename and rate book name, both the filename and the export file contained incorrect characters. This fix was to accurately render multi-byte characters in the filename and contents.
PC-18107	Date constants added to a rate routine using a non-English US locale caused errors during validation and execution. The fix allows date constants to be successfully defined and used in non-English US locales.
PC-18115	Previously, problems occurred when using non-US English format numeric or date constants as parameters for rate functions. Now, numeric and date constants as rate function parameters work with all locales.
PC-18127	Previously, Rate table lookup in a rate routine using numeric or date constant overrides only worked with locales that used comma group and period decimal delimiter. This now works with any locale.
PC-18152	In a locale that uses a decimal point other than period, you could not specify a modifier value using the locale specific decimal point. An attempt to do this caused validation errors.
PC-18378	Fixed an issue where the number format on Rating Worksheet was not consistent in the French locale. Rating Worksheets now display numbers using the configured locale-specific decimal point symbol.
PC-18493	Fixed an issue where importing a rate table into a rate book using the French locale, PolicyCenter threw a conversion exception. Previously PolicyCenter did not take into account the different number formats (comma instead of period for a decimal place).
<b>Reinsurance</b>	
PC-17954	Added Javadocs for the <code>RIProgramFinder</code> class.

## Platform Improvements and General Issues

ID	Description
<b>Archiving</b>	
PL-20690	A null pointer exception sometimes occurred during restore of archived claims.
PL-21380	The <code>regen-xsd</code> command generated an XSD file with errors. The command has been updated to produce a valid XSD file.
<b>Authentication</b>	
PL-21737	The Gosu implementation of the <code>AuthenticationSourceCreator</code> plugin now supports JBoss.
<b>Batch Processes</b>	
PL-20755	Fixed an issue in which worker threads could unexpectedly quit if any database/JDBC error occurred during a time span longer than the maximum wait period.
<b>Command Line Tools</b>	
PL-19768	Fixed an issue that occurred while attempting to execute the <code>verify-types</code> command on a 64-bit JVM, which caused an Out of Memory error.
PL-20110	The command line utility <code>gwpc regen-soap-api</code> no longer fails on Japanese Windows.
<b>Database</b>	

ID	Description
PL-8729	Guidewire removed the configuration parameter TableEstimatePercent from config.xml. Instead, use the samplingpercentage attribute of the database statistics element in config.sys, as the following sample XML code shows. <pre>&lt;databasestatistics samplingpercentage="20"&gt;</pre>
PL-10468	Fixed an issue on the Database Storage Information page regarding the table display whenever switching from Index Physical Statistics to Tables and Indexes.
PL-10469	With this release, Guidewire applications do not start if the CurrentEncryptionPlugin parameter in config.xml specifies an encryption plugin implementation that does not exist.
PL-14490	The data distribution tables now include a column for the internal major version number of the database schema.
PL-17742	On Oracle, changed database upgrade logic by providing additional optimizer hints to help improve upgrade performance.
PL-17835	Instead of evicting a database connection from the connection pool whenever an exception occurs, Guidewire applications now determine whether an exception was fatal to a connection before marking it for eviction. In addition, whenever the application marks a connection for eviction, the application logs the reason for the eviction. Constraint violations no longer cause evictions.
PL-20546	In data distribution tables, the size of numeric columns increased to hold larger values.
PL-20584	On Oracle, the Guidewire Profiler changed to help optimize the capture of the range of AWR snapshots.
PL-21145	Replaced the String value on DBNullConstraint Exception with display key Java.Database.DBException.NullConstraintViolation to enable localization of the message.
PL-21448	On SQL Server, PolicyCenter now provides additional information on database connections, which can then be captured and analyzed in the Server DMV Snapshot screen.
PL-21554	In previous versions on SQL Server, tuning queries was difficult because DBAs could not determine what part of the application generated the queries. Now, you can enable the new IdentifyQueriesViaComments parameter in config.xml to provide comments with contextual information in certain SQL Select statements sent to the relational database.  The SQL comments are in the format:  <i>ApplicationName:ActionName</i>  <i>ApplicationName</i> is ClaimCenter, PolicyCenter, or BillingCenter.  <i>ActionName</i> is the name of the page configuration file that submitted the SQL Select statement.
PL-22597	In earlier versions of PolicyCenter 7.x, if you defined an entity type that implements a Gosu interface, problems occurred if the interface had a method that returns an array of that entity type. None of the interface features were accessible on the entity that implemented the interface, and class cast exceptions could occur. Now, the interface features are accessible and class cast exceptions do not occur.

**Database Support**

PL-19689	Guidewire has changed the mechanism for creating the database performance reports for Oracle and SQL Server. Instead of clicking on a button and waiting for the report to be generated and packaged in a ZIP file for immediate download, the report is now generated by a background batch process that stores the completed report in a new table named xx_dbperfreport. This change requires a database upgrade to create this new table.
----------	---

**Database Upgrade**

PL-16977	Previously after a database upgrade, PolicyCenter logged differences between the metadata configuration and the upgraded database schema at the INFO logging level. Now, PolicyCenter logs differences at the WARN logging level.
----------	---

ID	Description
PL-17616	You can now directly run SQL commands as part of custom database upgrade triggers. For details, see “Using the IDatamodelUpgrade Plugin” in the <i>PolicyCenter Upgrade Guide</i> .
PL-21310	Fixed an issue that caused database upgrades to fail if the <code>WorkFlowWorkItem</code> table was populated. Upgrade succeeded if the table was empty. The fix involved changing the <code>WorkFlowWorkItem</code> entity attribute from Final to Extendable.
PL-21357	The <b>Upgrade Info</b> page and download now provide more consistent information about the steps that are being performed during an automated DB upgrade.
PL-21387	Corrected an issue that caused a null-pointer exception (NPE) in version triggers during a database upgrade.
PL-21452	Corrected a issue that caused Document Management file downloads to fail if running over SSL in Internet Explorer 8 or earlier.
PL-22007	In earlier versions, PolicyCenter ran the <code>AfterUpgradeVersionTrigger</code> on fresh databases, which could cause problems. With this release, PolicyCenter runs the <code>AfterUpgradeVersionTrigger</code> only on existing databases, not on fresh databases.
<b>Document Management</b>	
PL-18278	<p>Guidewire added a new configuration parameter, <code>AllowActiveXAutoInstall</code>. This parameter controls whether PolicyCenter automatically installs the Guidewire Document Assistant ActiveX control and supporting JScript support files used by the control. If <code>AllowActiveXAutoInstall</code> is set to <code>false</code> and <code>AllowActiveX</code> is <code>true</code>, PolicyCenter does not install the control but uses the control if it has been manually installed. Contact Guidewire Support for information about manually installing the control.</p> <p>In addition, Guidewire added the ability to customize the location of JScript support files used by the control. See “Specifying Location for Guidewire Document Assistant Scripts” in the <i>System Administration Guide</i>.</p> <p>Finally, Guidewire updated the version number of the control from 2.0.49/2.1.49 (white list) to 2.0.54/2.1.54 (white list).</p>
PL-20549	Fixed a race condition in the initialization of hidden frames for PolicyCenter that could cause the supporting JScript support files for the Document Assistant ActiveX control ( <code>TemplateRunner</code> ) to not be downloaded.
<b>Email</b>	
PL-13582	A problem prevented files with Unicode file names attached to email from reaching email clients. Now, email clients that handle attachments with Unicode file names, such as Microsoft Outlook, receive the attachments from PolicyCenter and send them correctly. If your email client does not handle Unicode file names, you must modify the <code>EmailMessageTransport</code> plugin to convert attachment file names to use Latin characters only.
<b>Entities/Metadata</b>	
PL-18068	Fixed a problem with verification of typelist typecodes on server startup. Typelist typecodes that match regardless of case fail validation. For example, the typecodes <code>foobar</code> and <code>FooBar</code> match disregarding case and thus fail validation.
PL-20193	When an element of an owned array changes, the parent changes as well. For <code>effdated</code> owned arrays, the wrong parent changed. Now, the correct parent changes. This issue also affects <code>effdated</code> owned one-to-ones, because they are implemented as owned arrays.
<b>File Export</b>	
PL-21546	Exported CSV files no longer contain UTF-8 BOM (byte order mark) characters at the beginnings of files that do not use UTF-8 encoding.
<b>Global Cache</b>	

ID	Description
PL-20332	A new configuration parameter, <code>GlobalCacheDetailedStats</code> , determines whether to collect detailed statistics for the global cache. Detailed statistics are data that PolicyCenter collects to explain why items are evicted from the cache. Basic statistics, such as miss ratio, are still collected regardless of the value of this parameter.  The <code>GlobalCacheDetailedStats</code> parameter is set to false by default. Set the parameter to true to help tune your cache. At runtime, use the Management Beans page to enable the collection of detailed statistics for the global cache. Whenever the <code>GlobalCacheDetailedStats</code> parameter is disabled, the Evict Information and Type of Cache Misses graphs are not displayed.
<b>Gosu</b>	
PL-20961	In Gosu you can now subclass an inner class that a supertype declares.
PL-22469	When an entity contains an <code>implementsInterface</code> element, that entity type is supposed to implement the interface specified in the <code>implementsInterface</code> element. Previously, methods inherited by the interface could be invoked, but references to the entity could not be assigned to variables of the interface type. This issue fixes the problem so that the entity type truly implements the interface, preserving assignability.
<b>Messaging</b>	
PL-13887	The <code>IMessagingToolsAPI</code> has a new method to retrieve the status of a messaging destination. The method signature is:  <code>getDestinationStatus(destID : int) : String</code>
<b>Plugins</b>	
PL-21409	In earlier versions, the User Authentication Service plugin ( <code>AuthenticationServicePlugin</code> ) threw four exceptions that <code>LoginForm.java</code> caught and displayed. There was no standard mechanism to add custom exceptions to the plugin and the login form. Now, you can modify the User Authentication Service plugin and throw <code>DisplayableLoginException</code> .
<b>Staging Tables</b>	
PL-20303	PolicyCenter offers two new consistency checks: "One-to-one non-null check" and "Edge foreign key non-null check".
<b>Studio IDE</b>	
PL-19308	An issue occurred with Studio because some source control systems require a file or directory in a directory controlled by Studio. Studio interfered with the special, source control files. Now, Studio safely ignores third party files and directories required by some source control systems.
PL-20531	Fixed a problem that could cause Studio to not properly render selected text and caret positions in files containing true tab characters.
PL-21216	Studio now generates a warning message during PCF verification (Verify All) if the number of attempted verification passes exceeds the number set for <b>Section Inclusion Limit</b> in Tools → Verification Options → PCF Verification Options.
PL-21296	Modified the Studio PCF Editor so that the PCF Toolbox now shows subcategories for all Application customized widgets that you can use in widget filtering.
PL-21447	Corrected an issue in which the Studio Gosu editor periodically froze for a brief period of time on large files if you typed quickly.
<b>Web Services</b>	
PL-21158	The Guidewire Profiler did not actually profile WS-I web services. Now whenever you select a WS-I web service to profile, PolicyCenter provides profile information.

## Known Issues and Limitations

This section describes known issues with this release of Guidewire PolicyCenter.

- PolicyCenter Known Issues

- Studio/Platform Known Issues

**Note:** For maintenance releases, Guidewire often defers fixing configuration issues if they 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

### [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.

### [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.

### [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`.

**Note:** 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.

### [In the default implementation, PolicyCenter sends the wrong written date to BillingCenter \(PC-17593\)](#)

**Issue** – When PolicyCenter and BillingCenter are integrated, the PolicyCenter default implementation incorrectly sends the `PolicyPeriod.WrittenDate` to the BillingCenter `BillingInstructionInfo`.

**Workaround** – Configure PolicyCenter to send the first `Transaction.WrittenDate` to BillingCenter:

1. In Studio, navigate to **Classes** → `gw` → **plugin** → `bc700` and open `BillingInstructionInfoEnhancement`.
2. In the `startSyncBasicPolicyInfo()` method, change:

```
this.WrittenDate = period.WrittenDate.XmlDateTime  
to  
this.WrittenDate = period.AllTransactions?.first().WrittenDate.XmlDateTime
```
3. Restart PolicyCenter to deploy your changes.

## Limitations of free-text search (PC-17496)

**Issue** – Free-text search does not support:

- Searching archived policies
- Non-US languages or non-ASCII characters
- Searching unbound cancellation jobs

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 **Non Renew** or **Not Take** the renewal. 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.

## Archived policies are not displayed in the Policies tab for AccountContact file (PC-15917)

**Issue** – On the **Account File Contacts** screen, archived policies are not appearing in the **Associated Policies** and **Associated Work Orders** tabs. On the **Account File Locations** screen, archived policies are not appearing in the **Policy Terms** and **Work Orders** list views. This is because PolicyCenter deleted the records in the active database used to identify to these entities when the policy was archived.

**Workaround** – You can configure PolicyCenter to retain the information about the archived policies in a way that it is available to the searches in the affected screens.

## Renewal processing for renewal conversion assumes BasedOn will be 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 new `gw.webservice.pc.pc700.job.startConversionRenewal` method.

## [BillingCenter integration: New Invoice Streams Are 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 underwriting edited something on the policy. This is an issue with BillingCenter 3.0.1 and later.

**Workaround** – Guidewire is aware of this issue.

## [Contacts created from the Contact tab are not autosaved \(PC-14761\)](#)

**Issue** – Contacts created from the **Contacts** tab are not getting autosaved. The **UnsavedWork** alert correctly goes on, but when the user clicks elsewhere, the work is lost.

**Workaround** – Set the **autosaveable** attribute on the **NewContact.pcf** to **true**.

## [Contact.AccountHolderCount incorrectly counts the number of accounts \(PC-14736\)](#)

**Issue** – **Contact.AccountHolderCount** incorrectly counts the number of accounts a particular contact is the account holder on.

**Workaround** – Query the database to count the accounts on which this contact has the account holder role.

## [Attachment inclusions are not copied across terms \(PC-14561\)](#)

**Issue** – RI attachment inclusions are not copied across terms, such as during a renewal or a rewrite.

**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 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 screen of the agent
- 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. 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 original answer is still saved along with the other answers.

**Workaround** – The answers to these *irrelevant* questions 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.

### Studio: “Verify Changed Resources” is not implemented in the product model (PC-6928)

**Issue** – The Studio Tools → Verify → Verify Changed Resources command does not work with the PolicyCenter product model.

**Workaround** – Use one of the other verification commands instead. See “Validating Studio Resources” in the *PolicyCenter Configuration Guide* for details.

## Studio/Platform Known Issues

### Quote purging does not generate events when data is purged (PL-22577)

**Issue** – If you use quote purging, be aware that purging does not generate events. This can impact external systems that track PolicyCenter entities or data that is subject to purge. If these systems rely on Deleted or Modified events, these systems will be out of sync after the purge occurs.

**Workaround** – Use the `prepareForPurge` method in the `PurgePlugin` plugin interface to track entities or data that is about to be purged.

### Studio Rules do not use correct capitalization for name of root object (PL-10740)

**Issue** – Rule set root objects are expected to begin with an upper-case letter and issues warning if a rule set root object does not.

**Workaround** – You can safely ignore the warning message.

### User interface cannot handle starting multiple instances of a batch process (PL-12372)

**Issue** – The user interface cannot handle starting multiple instances of a batch process.

**Workaround** – To execute multiple instances of a batch process, start them from the command line. Also, to allow multiple instances to be run simultaneously, you must ensure that the `BatchProcess.isExclusive()` method returns `false`.

### ‘US-Locations.txt’ file with the US geodata from GreatData causes validation problems with United States Postal Service (USPS) data (PL-13384)

**Issue** – The `US-Locations.txt` file contains information that does not conform to United States Postal Service (USPS) standards for bulk mailings.

**Workaround** – The provided `US-Locations.txt` file is intended only for use in geocoding to identify addresses for a location. You can edit the `US-Locations.txt` file to conform to your particular address standards, and then import that version of the file instead.

### 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.

### Cannot make a field from a delegate into a localized column (PL-13761)

**Issue** – You cannot make a field from a delegate into a localized column.

**Workaround** – Move the column to be localized off the delegate and onto each of the implementing entities. Then, to make the column appear as though it exists on the delegate, define an enhancement property on the delegate that *delegates* to the appropriate column, depending on the implementing entity.

## Problem with ‘regen-java-api’ command and JAR files (PL-16351)

**Issue** – If you run the `PolicyCenter/bin/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 into PolicyCenter using an administrative account.
2. Access Internal Tools → Reload.
3. Click Reload Workflow Engine.



# Guidewire PolicyCenter 7.0.7 Release Notes

## Release 7.0.7

**Document Revision:** 20-May-2014

Copyright © 2001-2014 Guidewire Software, Inc. All rights reserved.

Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Reinsurance Management, Guidewire ContactManager, Guidewire Vendor Data Management, Guidewire Client Data Management, Guidewire Rating Management, Guidewire InsuranceSuite, Guidewire ContactCenter, Guidewire Studio, Guidewire Product Designer, Guidewire Live, Guidewire DataHub, Guidewire InfoCenter, Guidewire Standard Reporting, Guidewire ExampleCenter, Guidewire Account Manager Portal, Guidewire Claim Portal, Guidewire Policyholder Portal, ClaimCenter, BillingCenter, PolicyCenter, InsuranceSuite, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks, service marks, or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries.

All other trademarks are the property of their respective owners.

**This material is confidential and proprietary to Guidewire and subject to the confidentiality terms in the applicable license agreement and/or separate nondisclosure agreement.**

Guidewire products are protected by one or more United States patents.

## Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Support

- Issues and Major Changes
- Improvements and General Issues
- Known Issues and Limitations

## Release Information

These release notes apply only to this release of Guidewire PolicyCenter.

---

**IMPORTANT** If you missed 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 7.0.7.

- If you are integrating with Guidewire BillingCenter, this version of PolicyCenter requires BillingCenter 3.0.4 or later or BillingCenter 7.0.1 or later.
- If you are integrating with Guidewire ClaimCenter, this version of PolicyCenter requires ClaimCenter 6.0.0 or later.
- If you are integrating with Guidewire ContactManager, this version of PolicyCenter requires ContactManager 7.0.0.

## 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 Resource Portal – <http://guidewire.custhelp.com>

## Issues and Major Changes

This section contains issues and major changes that may affect your installation.

- Base PCF File Changes
- Base Resources Changes
- Changes in This Release Provided in Upgrade Diff Report

### Base PCF File Changes

All links below require the `ReleaseNotes_files` directory on your local disk.

#### PolicyCenter release 7.0.6 to 7.0.7

- To view a report of the changes in the base PCF files in the `modules/pc` directory, [click here](#).
- To view a report of the changes in the base PCF files in the `modules/p1` directory, [click here](#).

## Base Resources Changes

All links below require the `ReleaseNotes_files` directory on your local disk.

### PolicyCenter release 7.0.6 to 7.0.7

- There are no changes to the base resources in the `modules/pc` directory.

## 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 Resource Portal.

## 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

ID	Description
<b>Account</b>	
PC-19113	Fixed an issue where <code>PolicyPeriod.getAllAccountSyncables</code> was no longer configurable. With this fix, you can now add your own <code>AccountSyncables</code> to PolicyCenter by modifying the implementation of <code>IPolicyPeriodPlugin#getAllAccountSyncables</code> . This method returns all <code>AccountSyncables</code> in the given <code>PolicyPeriod</code> including any new <code>AccountSyncables</code> that you create.
PC-19319	Fixed an issue where account level activities did not display on the <code>Account File Summary</code> screen.
PC-19628	You can now modify the <code>SourceAccount</code> , <code>RelationshipType</code> , <code>TargetAccount</code> , and <code>Retired</code> indexes on the <code>AccountAccount</code> entity in the <code>AccountAccount.etx</code> file. You can add additional fields, such as effective and expiration dates, to the index.
<b>Administration</b>	
PC-16316	Fixed an issue where PolicyCenter threw an exception the first time you went to <code>Administration → Actions → New Group</code> and clicked the search arrow for <code>Parent Group</code> , then <code>Search for Group</code> .
<b>Archiving</b>	
PC-17830	Fixed an issue where users could create multiple activities for the same request to retrieve an archived policy period. With this fix, PolicyCenter creates only one activity per user. If multiple users make the same request, PolicyCenter creates one activity per user, so each user can see their own activity.
PC-18594	For improved performance, the <code>Archived</code> property on <code>PolicyTerm</code> is now a derived property and <code>PolicyTerm</code> has a <code>PolicyTermArchiveState</code> property. Previously, <code>PolicyTerm.Archived</code> was <code>true</code> if any policy period in the term had a non-null <code>ArchiveState</code> . In this version, the behavior of this property is the same, but <code>PolicyTerm.Archived</code> derives its value by looking at <code>PolicyTerm.PolicyTermArchiveState</code> .
PC-19711	Fixed an issue where PolicyCenter displayed policy periods as archived when they were not actually archived. This occurred when the type system was reloaded.

ID	Description
PC-20096	If you have archived policy terms in previous versions of PolicyCenter you need to edit the domain XML schema (XSD) for archiving. Prior to this release, versions of the domain schema defined an incorrect type for referencing effective dated entities. The schema in PolicyCenter 7.0.7 defines the correct type for these entities. When you upgrade to this version of PolicyCenter, this type mismatch prevents PolicyCenter from retrieving these policy terms from the archive.
PL-24805	<p>If you did not use policy term archiving in previous releases, no action is required.</p> <p>If you used policy term archiving in previous releases, you must make the following change to any domain XML schema referenced from an XML file. If no XML files reference a particular domain XML schema, you do not need to make changes to that schema.</p> <p>The IArchivingSourcePlugin.storeSchema method creates these schemas. In the default configuration, the schema contains the product major and minor version, the application major and minor version, and your extension version. The schemas have target namespaces such as <code>http://guidewire.com/pc/p04_00300_a11_00604_00139/domain</code> where p04_0300 is the product major and minor version, a11_00604 is the application major and minor version, and 139 is your extension version. Archived XML files reference these namespaces.</p> <ol style="list-style-type: none"> <li>For each domain XML schema that is pointed to by XML files for archived policy terms, do a merge between that schema and the PolicyCenter 7.0.7 schema.</li> <li>Update the older schema with the PolicyCenter 7.0.7 attribute definition for each difference similar to the highlighted text below:</li> </ol> <p>Previous versions of PolicyCenter:</p> <pre>&lt;xs:attribute name="Fixed" type="xs:IDREF" use="required" gw:iType="entity.BANonOwnedBasis"/&gt;</pre> <p>PolicyCenter 7.0.7:</p> <pre>&lt;xs:attribute name="Fixed" type="xs:long" use="required" gw:type="java.lang.Long" gw:iType="entity.BANonOwnedBasis"/&gt;</pre>
<b>Contacts</b>	
PC-18142	Clarified an error message that occurs when PolicyCenter is integrated with ContactManager and the user searches for contacts by phone number only.
PC-18301	Fixed an issue in the AccountAPI interface. Previously the addContactToAccount method created a Company if the contact was a PersonContact. This method now correctly creates a Person.
PC-18556	Fixed an issue in the Web service Account API where adding a new Contact to an Account (addContactToAccount) returned a null public ID.
<b>Data Model</b>	
PC-19673	Resolved an issue with the Gosu type system that led to an infinite loop and a stack overflow exception
PL-24640	when constructing an entity type. The issue occurred if an entity used <implementsInterface> and that interface had a method that returned an array of the same entity.
<b>Infrastructure</b>	
PC-18774	Fixed an issue where PCSolrMessageTransportPlugin and PCSolrMessageTransport were initialized during PolicyCenter start up when free-text search was not enabled. Free-text search uses an integration with the full-text search engine Solr.
PC-18821	Previously, searching for a policy with a blank search field could cause a stack trace. If you created or modified PCF files in the default configuration, check for the following condition. Find PCF files which have SearchItem elements related to PolicySearch. If the afterFailure property contains a cast of the form:  (searchCriteria as PolicySearchCriteria)  Replace this with a protected cast:  if (searchCriteria typeis PolicySearchCriteria) searchCriteria
PC-18874	The Free-text Search tools were moved from System Tools to Internal Tools.
PC-19493	Fixed a issue where PolicyCenter did not send an event to inform basic search to remove the index entry on a purged quote. Previously, the policy period was searchable. However, the policy period did not appear in the display results. This was previously a known issue.
PC-19540	Fixed a potential concurrency issue in free-text search.
PC-20037	Fixed an issue where the Solr plugin sent search queries as a GET. Now the plugin sends search queries as POST.

ID	Description
<b>Jobs</b>	
PC-14967	Fixed an issue where retired typekeys were shown in drop-down lists. With this fix, retired typekeys are no longer shown so they cannot be selected for new submissions. For existing policy periods with retired typekeys, PolicyCenter displays a validation error when validating the policy period.
PC-18550	Fixed an issue where PolicyCenter threw an exception in side-by-side quoting if you reset the middle policy period to the Standard Program offering in a personal auto policy.
PC-18682	Policy searches from ClaimCenter now return the Primary Insured name that corresponds with the date being requested.
PC-18688	Fixed an issue where starting a cancellation on a converted renewal caused PolicyCenter to throw an error.
PC-18836	Fixed an issue where the installment plan listview was not refreshed after selecting an installment plan.
PC-19036	Fixed an issue that occurred when copying a submission. If you set the submission copy's effective date to be before the effective date of the original submission, when you quoted the policy, the EffectiveDatesForRating property returned two dates. With this fix, PolicyCenter only returns the effective date of the copy.
PC-19244	Removed the <b>Handle Preemption</b> button which appeared in a renewal job. Preemption cannot occur in a renewal.  Note: You can use a policy change to update a pending renewal. If you do not apply changes to the renewal when binding the policy change, you can apply them later. In the policy change job, click <b>Apply Change to Renewal</b> .
PC-19306	Updated an error message to reflect that quoting depends on both edit and quote permissions.
PC-19314	Fixed an issue where policy holds on policy changes did not work if you did not specify any coverages.
PC-19413	Previously in side-by-side quoting, PolicyCenter considered two entities different if both entities had a null parent identity column and all other fields matched. This caused unnecessary database updates to synchronize the entities. Now PolicyCenter considers these entities equivalent.
PC-19728	Fixed an issue in a renewal with two in-sequence policy changes. Previously, although there were no out-of-sequence conflicts, the <b>Job Complete</b> screen displayed two out-of-sequence conflicts.
PC-19762	Fixed an issue where PolicyCenter did not display warnings if a coverage was automatically removed during Copy Data.
PC-20181	Added an as-of date selector that appears when copying data between policies. When you select a work order to copy from, PolicyCenter displays the slice on the edit effective date of the work order. For policy terms, PolicyCenter displays the last slice of the policy period. For policy terms, you have an option to specify a date which represents the slice of the policy at that particular time. PolicyCenter displays the entities available from that slice of the policy.
PC-20185	Fixed an issue where PolicyCenter displayed policy conflicts when no conflicts existed. This occurred on a second policy change.
<b>Lines of Business</b>	
PC-18394	Fixed an issue where PolicyCenter displayed an incorrect error message if a user entered a value beyond the integer maximum of 2147483647 in a General Liability Exposure basis field.
PC-20114	Fixed an issue in a business auto submission job. Previously, coverages on a cloned vehicle disappeared if you changed the effective date during a submission.
<b>Localization</b>	
PC-19627 PL-24462	In previous releases, PolicyCenter incorrectly handled Imperial dates that equate to Gregorian Dates in the year 1950, as well as other ranges of old dates. The issue occurred because American military bases used daylight saving time in Japan beginning in 1948, which differed from timekeeping used by the general public. With this release, PolicyCenter handles these Imperial dates correctly.
<b>Miscellaneous</b>	
PC-19460	Fixed a problem in the <code>isOverlap</code> method of <code>EffectiveDateCheckVerifierBase.gs</code> . The method was returning incorrect values in certain conditions.
<b>Product Model</b>	
PC-19597	Fixed an issue where you could not set a default value on product model direct coverage term with a negative scale.

ID	Description
<b>Rating</b>	
PC-17933	Fixed an issue where the rate book was not selected if the user selected an offering.
PC-18404	Fixed an issue with saving a rate table containing Japanese characters in the name. Previously those characters became underscores in the filename.
PC-18795	In a development environment, changes to rate table data were not picked up until server restart. This did not affect a production environment. This issue is now fixed.
PC-19053	For certain use cases such as seasonal rating, you can now configure custom proration methods in the IProrationPlugin. The example code that implements this plugin is ProrationPlugin.gs.
PC-19068	Previously you could export a rate book that referenced a rate table which was not included in the rate book. This could result in an error when importing the rate book if the referenced rate table was deleted. Now when promoting a rate book, PolicyCenter requires that all rate tables are included in the rate book. PolicyCenter also displays an error if you attempt to export a rate book that references a rate table not included in the rate book.
PC-19094	Fixed an issue where rating parameters appeared even if they were not part of the selected parameter set. With this fix, if you switch between different parameter sets, only parameters belonging to the currently selected parameter set are selectable.
PC-19102	When editing a rate routine, you can now include parentheses in conditional expressions, such as the operand of an IF instruction. The parentheses must be correctly balanced.
PC-19201	In a multi-line product, you can now view rating worksheets for all available policy lines on a policy. Previously, you could view rating worksheet for only the first policy line.
PC-19412	Fixed an issue that caused the quote to fail if a policy change was applied to a renewal in a future policy term. PolicyCenter generated an invalid quote and threw an exception because of more than one reference the RatingWorksheet entity.
PC-19633	Fixed an issue where numeric constants were not displayed on the rating worksheet.
PC-19638	Previously if a rate routine referenced a modifier that was set to unavailable in product model, the value of the modifier was 1. This was incorrect. Now, the value of the modifier in the rate routine is 0 if the modifier is a schedule rate modifier and is unavailable. Otherwise, the value is 1.
PC-19837	Changing decimal place precision supported for rate table parameters in the user interface. Added a Decimal Places field to the Rate Table Definition → Parameters → Parameter Details tab. In this field you specify the number of decimal places that the user can enter when entering decimal parameters. This field applies to both values of a range parameter. This field only appears for the Decimal data type.
PC-20232	Impact testing now excludes archived policies.
<b>Reinsurance</b>	
PC-14561	Added a denormalized PolicyTerm column to RIAttachmentInclusion. The DraftRiskVersionList column is now required and always set instead of being set only when the version list is in draft status.
<b>Underwriting Authority</b>	
PC-19050	Neither audit nor cancellation jobs support underwriting issues in the base configuration. The Add UW Issue button is now disabled in audit and cancellation jobs.
<b>Upgrade</b>	
PC-19671 PL-23686	With this release, you can control whether to include updates of database statistics as part of the database upgrade command. A new updatestatistics attribute is added to the <upgrade> element. In the default configuration, the updatestatistics attribute is set to true to preserve the previous behavior. To avoid updating database statistics during the database upgrade command, set the value to false. If you set updatestatistics to false, you must update database statistics at a later time.
PC-19672 PL-23785	Altered the algorithm used to calculate column statistics for database upgrade.
PC-19807	Renamed upgrade trigger <code>DucatiPopulateVehicleForeignKeyOnBusinessVehicleCovCostVersionTrigger</code> to <code>DucatiPopulateBusinessVehicleOnBusinessVehicleCovCostVersionTrigger</code> . Added archive upgrade trigger to <code>DucatiPopulateBusinessVehicleOnBusinessVehicleCovCostVersionTrigger</code> .

## Platform Improvements and General Issues

ID	Description
<b>Archiving</b>	
PL-23757	With this release of PolicyCenter, whenever the server detects non-cyclic errors during server startup, the server writes an additional message to the log for each cyclic path found in the domain graph, specified with dot notation. You control whether the server detects non-cyclic errors during server startup by setting the <code>DisableDomainGraphSupport</code> parameter in <code>config.xml</code> to true. The default value is false.
<b>Build Infrastructure</b>	
PL-15509	<p>In previous versions, Guidewire supported running build scripts only through the <code>gwpc</code> command on Windows. With this version you, can run the EAR and WAR build scripts on Unix by invoking Ant directly, with the following command:</p> <pre>ant -f PolicyCenter/modules/ant/build.xml buildScript</pre> <p>Substitute <code>buildScript</code> with one of the following values:</p> <ul style="list-style-type: none"> <li>• <code>build-jboss-war</code> – Builds a generic WAR file for use with JBoss</li> <li>• <code>build-tomcat-war</code> – Builds a generic WAR file for use with Tomcat</li> <li>• <code>build-weblogic-ear</code> – Builds an EAR file for use with WebLogic</li> <li>• <code>build-websphere-ear</code> – Builds an EAR file for use with WebSphere</li> </ul> <p>Any other build scripts that you can run through the <code>gwpc</code> command must be run on Windows.</p>
<b>Cognos Integration</b>	
PL-22539	With this release, the Cognos plugin no longer removes domain names from the URL parameter to the <code>setCognosCookiesUrl</code> plugin method.
<b>Command line tools</b>	
PL-20378	With this release, the <code>verify-types</code> command-line tool does not stop verification if a failure, such as a programming error, occurs while verifying a specific type. The tool reports all errors, warnings and failures.
<b>Database support</b>	
PL-21842	With this release, the <code>system_tools</code> command has additional options for submitting batch jobs that report database performance.
PL-23523	For Oracle, you can configure new LOB columns to use SecureFile LOBs or compressed SecureFile LOBs instead of the default BasicFile LOBs. You can configure the LOB type for the entire PolicyCenter database or for specific tables only. For more information, see “Configuring Oracle LOB Types” in the <i>Installation Guide</i> .
<b>Database upgrade</b>	
PL-22478	In earlier releases on SQL Server, queries failed if they contained a large number of CASE conditions. Now, PolicyCenter breaks up queries with more than 125 CASE conditions into nested CASE clauses or into separate queries.
PL-22686	With this release, upgrade version checks now include validation SQL in the downloads that you obtain from the <a href="#">Upgrade Info</a> page.
PL-22934	This release fixes an issue that caused a SQL failure during database upgrade.
PL-23044	With this release, the database upgrade now supports table and index partitioning. Specify which tables and indexes you want to have partitioned, and the upgrader creates any new table and new indexes appropriately. Existing tables and indexes remain unchanged, regardless of the discrepancy between the metadata configuration file and the database schema.
PL-23104	With this release, the automated database upgrade supports changing a column on an entity definition from nullable to non-nullable.

ID	Description
PL-23504	<p>With this release, a database upgrade in a development environment records checkpoints of upgrade triggers that complete successfully. You can restart a failed database upgrade, and it resumes with the upgrade trigger that failed. The restart feature helps you test your upgrade with realistically large data sets. You avoid time spent to restore the database and time spent to run upgrade triggers that work successfully.</p> <p>To restart a test database upgrade from a checkpoint reached in an earlier upgrade, you need to roll back manually any database changes that occurred during the upgrade trigger that failed. In addition, assure you resolved the problem that caused the trigger to fail before restarting. A test run of your upgrade is successful only when it runs from start to finish without a restart. Never use the restart feature of database upgrade in a production environment.</p>
<b>Document Management</b>	
PL-13904	A new class, DocTreeNode, simplifies the creation of a tree that is based on a customer defined path.
PL-22562	<p>With this release, you can pass custom variables from document descriptor files to document templates and merge scripts. Custom variables let you add information to produced documents, including to their headers and footers, and they let you add conditional processing logic. First, declare the variable in the XSD for document template descriptors. Second, set the variable to a value in the metadata of the appropriate document descriptor file. Finally, use the variable, with an underscore (_) as a prefix, in the appropriate document template or merge script.</p> <p>For example, to add conditional processing to the production of Word documents:</p> <ol style="list-style-type: none"> <li><b>1. Declare the variable in the XSD for document template descriptors</b> – In Studio, open the file document-template.xsd. Locate this XSD by pressing CTRL+N and entering document-template. Add the variable as an XSD attribute, as the following sample code shows.</li> </ol> <pre>&lt;xsd:attribute name="mail-merge-type" type="xsd:string" use="optional"/&gt;</pre> <p>Then, save the file.</p> <ol style="list-style-type: none"> <li><b>2. Set the variable to a value in the document template descriptor</b> – In Studio, navigate to resources → Other Resources → doctemplates and open the file SampleLetter.gosu.rtf.descriptor. Add the variable as attribute on the &lt;DocumentTemplateDescriptor&gt; element, as the following sample code shows.</li> </ol> <pre>mail-merge-type="catalog"</pre> <p>Then, save the file.</p> <ol style="list-style-type: none"> <li><b>3. Use the variable in the Mail Merge and Close merge script</b> – In Studio, navigate to resources → Other Resources → Web Resources → resources → documentassistant. Open the file MailMergeAndClose.js. Modify the file to conditionalize the following Javascript statement, as the following sample code shows.</li> </ol> <pre>if (_mail-merge-type == "catalog") {     objWord.MailMerge.MainDocumentType = /* wdCatalog */ 3; }</pre>
PL-22645	With this release, users should be able to edit documents whenever IDocumentContentSource uses mode=uri.
PL-22737	In previous versions, the BFO library was broken for some PDF templates. This version includes an updated BFO library that corrects the problem.
<b>Entities/Metadata</b>	
PL-22866	Calling an API method that attempts to modify a read-only bundle now results in an immediate error.
<b>Management plugin</b>	
PL-16543	With this release, the counters NumActiveDBConnections and NumIdleDBConnectionsmanagement are integers instead of strings, so management tools that connect via JMX can plot them now on graphs.
<b>Persistence</b>	
PL-13380	With this release of PolicyCenter, the <i>Data Dictionary</i> has a Database View for data entities. The Database View shows persistent entities, and it includes column, typekey, and foreign key fields associated with each entity. Array and virtual fields are excluded from the Database View.
PL-19786	With this release of PolicyCenter, calls to DiffUtils.java.compareBean reports changes to one-to-one fields.
PL-23941	For workflow-related distributed work queues, the logging level of the message "WDW processing workitem: ..." changed from INFO to DEBUG to reduce noise in the log files.

ID	Description
<b>Web - Other</b>	
PL-18559	Added alt-text, or alternative tooltip text, for the following: <ul style="list-style-type: none"><li>• BooleanRadioInput</li><li>• PicketLink</li><li>• DocumentationCell</li></ul>
<b>Web - UI/Runtime</b>	
PL-23848	PCF input elements now render currency amounts so the scale and appscale parameters are honored. For example, if appscale == 0 and the amount is the integer 1, the input element renders the amount as "1". If appscale == 2 and the amount is the integer 1, the input element renders the amount as "1.00".
<b>Web services – WS-I</b>	
PL-23514	This release introduces a new method, preExecute, for use in the invocation handlers that you write for WS-I web services. In earlier releases, if you wrote an invocation handler, you unavoidably bypassed some important WS-I features. For example, the application: 1) did not enable profiling for method calls; 2) did not check run level annotations, even at the class level; 3) did not check web service permission annotations, even at the class level; (4) did not check for duplicate external transaction IDs if those SOAP headers are present. Now, you can call the preExecute method in your invocation handler to assure the application takes the preceding actions, as needed. See the <i>Integration Guide</i> for more information about invocation handlers and how to use the preExecute method.
PL-23721	To detect duplicate operations from external systems, add the new annotation @WsCheckDuplicateExternalTransaction to your WS-I web service implementation class. See the <i>Integration Guide</i> for details for using the annotation and the SOAP header for the transaction ID.

## Known Issues and Limitations

This section describes known issues with this release of Guidewire PolicyCenter.

- PolicyCenter Known Issues
- Studio/Platform Known Issues

**Note:** For maintenance releases, Guidewire often defers fixing configuration issues if they 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

#### [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.

#### [PolicyCenter throws exception when changing locale on Rating Worksheet screen \(PC-19645\)](#)

**Issue** – PolicyCenter throws an exception if you navigate to the Rating Worksheet screen and then change the locale.

**Workaround** – Close PolicyCenter and log in again. You can reset the locale back to its original setting.

## 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.

## 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.

## In the default implementation, PolicyCenter sends the wrong written date to BillingCenter (PC-17593)

**Issue** – When PolicyCenter and BillingCenter are integrated, the PolicyCenter default implementation incorrectly sends the **PolicyPeriod.WrittenDate** to the BillingCenter **BillingInstructionInfo**.

**Workaround** – Configure PolicyCenter to send the first **Transaction.WrittenDate** to BillingCenter:

1. In Studio, navigate to **Classes** → **gw** → **plugin** → **bc700** and open **BillingInstructionInfoEnhancement**.
2. In the **startSyncBasicPolicyInfo()** method, change:  

```
this.WrittenDate = period.WrittenDate.XmlDateTime  
to  
this.WrittenDate = period.AllTransactions?.first().WrittenDate.XmlDateTime
```
3. Restart PolicyCenter to deploy your changes.

## 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.

#### [Problem with Solr Batch Loader and non-UTF-8 characters sets](#)

**Issue** – If you run the free-text batch load command in an environment in which UTF-8 is not the default character set, the index documents loaded may not contain the correct characters.

**Workaround** – Run the free-text batch load command in an environment where the default character set is UTF-8.

#### [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 cancellation, 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.

#### [Archived policies are not displayed in the Policies tab for AccountContact file \(PC-15917\)](#)

**Issue** – On the **Account File Contacts** screen, archived policies are not appearing in the **Associated Policies** and **Associated Work Orders** tabs. On the **Account File Locations** screen, archived policies are not appearing in the **Policy Terms** and **Work Orders** list views. This is because PolicyCenter deleted the records in the active database used to identify to these entities when the policy was archived.

**Workaround** – You can configure PolicyCenter to retain the information about the archived policies in a way that it is available to the searches in the affected screens.

#### [Renewal processing for renewal conversion assumes BasedOn will be 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 new `gw.webservice.pc.pc700.job.startConversionRenewal` method.

#### [BillingCenter integration: New invoice streams are 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 underwriting edited something on the policy. This is an issue with BillingCenter 3.0.1 and later.

**Workaround** – Guidewire is aware of this issue.

#### [Contacts created from the Contact tab are not autosaved \(PC-14761\)](#)

**Issue** – Contacts created from the **Contacts** tab are not getting autosaved. The `UnsavedWork` alert correctly goes on, but when the user clicks elsewhere, the work is lost.

**Workaround** – Set the `autosaveable` attribute on the `NewContact.pcf` to true.

#### [Contact.AccountHolderCount incorrectly counts the number of accounts \(PC-14736\)](#)

**Issue** – `Contact.AccountHolderCount` incorrectly counts the number of accounts a particular contact is the account holder on.

**Workaround** – Query the database to count the accounts on which this contact has the account holder role.

#### [Attachment inclusions are not copied across terms \(PC-14561\)](#)

**Issue** – RI attachment inclusions are not copied across terms, such as during a renewal or a rewrite.

**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 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 screen of the agent
- 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 subtentities 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.

## Studio: “Verify Changed Resources” is not implemented in the product model (PC-6928)

**Issue** – The Studio Tools → Verify → Verify Changed Resources command does not work with the PolicyCenter product model.

**Workaround** – Use one of the other verification commands instead. See “Validating Studio Resources” in the *PolicyCenter Configuration Guide* for details.

## Studio/Platform Known Issues

### Studio Rules do not use correct capitalization for name of root object (PL-10740)

**Issue** – Rule set root objects are expected to begin with an upper-case letter and issues warning if a rule set root object does not.

**Workaround** – You can safely ignore the warning message.

### User interface cannot handle starting multiple instances of a batch process (PL-12372)

**Issue** – The user interface cannot handle starting multiple instances of a batch process.

**Workaround** – To execute multiple instances of a batch process, start them from the command line. Also, to allow multiple instances to be run simultaneously, you must ensure that the `BatchProcess.isExclusive()` method returns `false`.

### Type system refresh after PCF page title change does not update corresponding menu label (PL-13057)

**Issue** – The type system refresh after a PCF page title change does not update the corresponding menu label

**Workaround** – After updating a page title, restart the server to refresh menu labels and avoid null pointer exceptions due to stale references.

### US-Locations.txt file with the US geodata from GreatData causes validation problems with United States Postal Service (USPS) data (PL-13384)

**Issue** – The `US-Locations.txt` file contains information that does not conform to United States Postal Service (USPS) standards for bulk mailings.

**Workaround** – The provided `US-Locations.txt` file is intended only for use in geocoding to identify addresses for a location. You can edit the `US-Locations.txt` file to conform to your particular address standards, and then import that version of the file instead.

### 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.

## Cannot make a field from a delegate into a localized column (PL-13761)

**Issue** – You cannot make a field from a delegate into a localized column.

**Workaround** – Move the column to be localized off the delegate and onto each of the implementing entities. Then, to make the column appear as though it exists on the delegate, define an enhancement property on the delegate that *delegates* to the appropriate column, depending on the implementing entity.

## Problem with regen-java-api command and JAR files (PL-16351)

**Issue** – If you run the `PolicyCenter/bin/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 into PolicyCenter using an administrative account.
2. Access Internal Tools → Reload.
3. Click Reload Workflow Engine.

## In some languages, web browsers render column headers of list views improperly (PL-18027)

**Issue** – In some languages, web browsers render some column headers of list views improperly if their column widths are specified too narrowly in their PCF definitions. For example, sometimes a numeric column is specified with a variable width of 1%. This narrow setting forces the browser to render the column too narrowly for the text of the translated column heading.

**Workaround** – Edit the PCF file that defines the column and clear the value from the width property. Without a specified value for the column width, browsers render the column widely enough to display the full text of the translated column heading.

## Error on print for CheckBoxGroupInput Widget in Popup (PL-22209)

**Issue** – Problems occur whenever a user clicks a PrintToolbar button on a page that has a CheckBoxGroupInput widget.

**Workaround** – Alter the page that has the problem to remove the CheckBoxGroupInput widget.

## Database upgrade does not handle nullable to non-nullable columns with a default value for sub-types (PL-23104)

**Issue** – For entity definitions, the automatic database upgrade converts nullable columns to non-nullable with a default value successfully. However, this column type conversion is not possible for columns in subtype definitions. PolicyCenter implements non-nullable columns on subtype in the database as nullable because that column must have null values for rows that represents instances of other subtypes.

**Workaround** – Write a version trigger to populate the column with the default value for existing rows for the subtype. After you upgrade, PolicyCenter enforces the column value to be non-nullable with the default value for new rows of the subtype.

#### New transport plugin definitions do not show in the list of valid transport plugins (PL-23317)

**Issue** – Newly implemented transport plugin definitions do not show in the list of valid transport plugins displayed by clicking the light-bulb icon next to the Transport Plugin field in the messaging destination editor.

**Workaround** – Restart Studio.

#### Solr does not work with the JMX implementation in WebSphere (PL-23371)

**Issue** – Solr does not work with the JMX implementation in WebSphere. This is a known JMX issue. See <https://issues.apache.org/jira/browse/CAMEL-5187>.

**Workaround** – Turn off JMX in Solr, or use a different application server, such as Tomcat.

#### Find in Resources fails for resources under Data Model Extensions or Web Resources (PL-23320)

**Issue** – In Studio, the **Find in Resources** option does not work for resources that are in Data Model Extensions or Web Resources.

**Workaround** – To see the full resource path, hover over the resource name in the tab of the resource editor. Then, navigate the resource pane on the left to find the resource in the resource hierarchy.

# Guidewire PolicyCenter 8.0.0

## Release Notes

### Release 8.0.0

**Document Revision:** 20-May-2014

Copyright © 2001-2014 Guidewire Software, Inc. All rights reserved.

Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Reinsurance Management, Guidewire ContactManager, Guidewire Vendor Data Management, Guidewire Client Data Management, Guidewire Rating Management, Guidewire InsuranceSuite, Guidewire ContactCenter, Guidewire Studio, Guidewire Product Designer, Guidewire Live, Guidewire DataHub, Guidewire InfoCenter, Guidewire Standard Reporting, Guidewire ExampleCenter, Guidewire Account Manager Portal, Guidewire Claim Portal, Guidewire Policyholder Portal, ClaimCenter, BillingCenter, PolicyCenter, InsuranceSuite, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks, service marks, or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries.

All other trademarks are the property of their respective owners.

**This material is confidential and proprietary to Guidewire and subject to the confidentiality terms in the applicable license agreement and/or separate nondisclosure agreement.**

Guidewire products are protected by one or more United States patents.

### Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Support

- Issues and Major Changes
- Known Issues and Limitations

## 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.

## 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 Resource Portal – <http://guidewire.custhelp.com>

## Issues and Major Changes

This section contains issues and major changes that may affect your installation.

- Upgrade does not convert Money properties to MonetaryAmount properties in entity extensions
- Base PCF File Changes
- Base Resource Changes
- Changes in This Release Provided in Upgrade Diff Report

### Upgrade does not convert Money properties to MonetaryAmount properties in entity extensions

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 Resource Portal.

## Known Issues and Limitations

This section describes known issues with this release of Guidewire PolicyCenter.

- PolicyCenter Known Issues
- Product Designer Known Issues
- Studio/Platform Known Issues

### 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 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.

## 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 cancellation, 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.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 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_Rules`).

## 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:

```
<Input id="xxx"
      postOnChange="true"
      onChange="someMethod()"
      disablePostOnEnter="doEvaluation()"/>
```
  - New syntax example:

```
<Input id="xxx">
  <PostOnChange onChange="someMethod()" disablePostOnEnter="doEvaluation()"/>
</Input>
```
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.



# Guidewire PolicyCenter 8.0.1 Release Notes

## Release 8.0.1

**Document Revision:** 20-May-2014

Copyright © 2001-2014 Guidewire Software, Inc. All rights reserved.

Guidewire, Guidewire Software, Guidewire ClaimCenter, Guidewire PolicyCenter, Guidewire BillingCenter, Guidewire Reinsurance Management, Guidewire ContactManager, Guidewire Vendor Data Management, Guidewire Client Data Management, Guidewire Rating Management, Guidewire InsuranceSuite, Guidewire ContactCenter, Guidewire Studio, Guidewire Product Designer, Guidewire Live, Guidewire DataHub, Guidewire InfoCenter, Guidewire Standard Reporting, Guidewire ExampleCenter, Guidewire Account Manager Portal, Guidewire Claim Portal, Guidewire Policyholder Portal, ClaimCenter, BillingCenter, PolicyCenter, InsuranceSuite, Gosu, Deliver Insurance Your Way, and the Guidewire logo are trademarks, service marks, or registered trademarks of Guidewire Software, Inc. in the United States and/or other countries.

All other trademarks are the property of their respective owners.

**This material is confidential and proprietary to Guidewire and subject to the confidentiality terms in the applicable license agreement and/or separate nondisclosure agreement.**

Guidewire products are protected by one or more United States patents.

## Overview

These release notes contain the following:

- Release Information
- Installing This Release
- Support

- Issues and Major Changes
- Improvements and General Issues
- Known Issues and Limitations

## 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 7.0.0 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

## Support

For assistance, visit the Guidewire Resource Portal – <http://guidewire.custhelp.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 Requires Committing All Change Lists Before Upgrading PolicyCenter
- Base PCF File Changes
- Base Resource Changes
- Changes in This Release Provided in Upgrade Diff Report

## 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 supports 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 Resource Portal.

## 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
- Product Designer Improvements and General Issues
- Platform Improvements and General Issues

## PolicyCenter Improvements and General Issues

ID	Description
<b>Account</b>	
PC-16669	As of this release, the <code>Lookups.gs</code> class is deprecated. See the class for suggested replacement methods.
PC-22022	The <code>ILocationPlugin</code> was renamed <code>AccountLocationPlugin</code> .
<b>Administration</b>	
PC-22323	Fixed an intermittent issue with the <b>New Producer Code</b> → <b>Roles</b> tab. In some instances, selecting the <b>Name</b> drop-down list resulted in an “Invalid server log” message.
<b>Archiving</b>	
PC-17887	The associated policy transactions card on the <code>AccountFile</code> contacts section now shows data for the <code>PrimaryNamedInsured</code> contacts of an archived policy period.
PC-21557	Fixed an issue in the policy <b>Summary PCF</b> 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.
<b>Contacts</b>	
PC-23017	Contacts in PolicyCenter and ContactManager now synchronize correctly after a contact merge in ContactManager.
<b>Financials</b>	
PC-17321	Added <b>Earned Premium</b> to the policy <b>Summary</b> screen.
<b>Infrastructure</b>	
PC-20234	Fixed an issue where <code>PCSolrMessageTransportPlugin</code> , <code>PCSearchPlugin</code> , and <code>PCSolrMessageTransport</code> 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 <code>InvalidSlice</code> differences when running <b>Consistency Check</b> from <b>Internal Tools</b> > <b>Free-text Search</b> .
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: <ul style="list-style-type: none"> <li>• <code>IPolicyHoldJobEvalPlugin.findJobsToEvaluate</code></li> <li>• <code>PCArchivingUtilInternal.findArchivingCandidates</code></li> <li>• <code>PCArchivingUtilInternal.findRestoreCandidates</code></li> </ul>
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 <code>&lt;br&gt;</code> occurrences from labels in PCF files and replaced with <code>/n</code> . HTML is not allowed in labels.
<b>Integration</b>	
PC-14630	The <code>IPolicyPortalAPI.gs</code> 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.

ID	Description
PC-17810	Fixed an issue with multicurrency and the <code>gw.plugin.billing.impl.StandAloneBillingSummaryPlugin</code> 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 <i>conversion on renewal process</i> did not set the <code>MostRecentTerm</code> on the converted policy. The renewal batch process can only operate on policies containing a <code>PolicyTerm</code> with the <code>MostRecentTerm</code> bit set to true. This issue prevented the policy from being picked up by the Renewal batch process and from being renewed.
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 <b>Unapplied Funds</b> 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: <ul style="list-style-type: none"> <li>• <code>gw.plugin.contact.ab800</code> – Contact Manager integration</li> <li>• <code>gw.contactmapper.ab800</code> – Contact Manager integration supporting classes</li> <li>• <code>gw.plugin.claimsearch.cc800</code> – ClaimCenter integration</li> <li>• <code>gw.plugin.billing.bc800</code> – BillingCenter integration</li> </ul>
PC-22008	Renamed the <code>selectFinalAuditCancellationSchedulePattern</code> method to <code>selectFinalAuditSchedulePatternForCancellation</code> . Renamed the <code>selectFinalAuditExpirationSchedulePattern</code> method to <code>selectFinalAuditSchedulePatternForExpiredPolicy</code> . The old names are deprecated.
PC-22016	Previously, the methods in the billing summary plugin ( <code>IBillingSummaryPlugin</code> ) 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 <code>IBillingSystemPlugin</code> 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 <code>IBillingSystemPlugin</code> plugin interface. <p>The <code>isAccountExist</code> method is now <code>accountExists</code>. The <code>isProduceExist</code> method is now <code>producerExists</code>. The <code>isAccountExist</code> and <code>isProduceExist</code> methods are deprecated.</p> <p>Changed parameter of <code>updatePolicyPeriodTermConfirmed</code> method from <code>Boolean</code> to <code>boolean</code> so that it no longer allows <code>null</code>. The previous method signature is deprecated.</p>
PC-22021	Deprecated <code>ClaimDetail getClaimDetailByClaimNumber(Claim claim)</code> since it referred to <code>ClaimNumber</code> which is not a part of the API signature. Use <code>ClaimDetail getClaimDetail(Claim claim)</code> in its place.
PC-22062	Plugins that communicate with WSI web service APIs are now thread-safe.
PC-22132	Previously, the <i>conversion on renewal process</i> did not set the <code>IssueDate</code> when issuing the renewal job. This prevented subsequent jobs from being taken on the Policy because the default <code>PolicyPlugin</code> checks for <code>Policy.Issued</code> in <code>canStart&lt;job&gt;</code> .
PC-22263	Previously on the <b>Policy Billing</b> screen, the <b>Period</b> drop-down showed both bound and unbound periods even though BillingCenter only has bound periods. Now the <b>Period</b> drop-down only shows bound periods.
PC-22337	To support multicurrency, the <code>IBillingSummaryPlugin</code> plugin interface changed arguments and return values that represent money (for example, <code>BigDecimal</code> types) to the new <code>MonetaryAmount</code> 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 <code>BundleEnhancement.turnOffValidation</code> method has been deprecated.
PC-22694	In <code>IBillingSystemPlugin</code> , removed the <code>CompatibilityMode</code> flag which provided backwards-compatibility with PC 4.0.

<b>ID</b>	<b>Description</b>
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.
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 <b>Currency</b> 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.
<b>Internationalization</b>	
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 <b>Currency Selection for a Producer Code</b> .
PC-21897	Web service interfaces were modified to transmit the following international name and address fields: <ul style="list-style-type: none"><li>• Name – FirstNameKanji, LastNameKanji, Particle, and CompanyNameKanji</li><li>• Address – AddressLine1Kanji, AddressLine2Kanji, CityKanji, CEDEX, and CEDEXBureau</li></ul> The following web services were modified: <ul style="list-style-type: none"><li>• gw.webservice.pc.pc800.account<ul style="list-style-type: none"><li>• AccountAPI.gs, AccountSearchInfo.gs</li><li>• gw.webservice.pc.pc800.community.datamodel<ul style="list-style-type: none"><li>• AddressDTO.gs</li><li>• ProducerCodeDTO.gs</li></ul></li><li>• gw.webservice.pc.pc800.contact<ul style="list-style-type: none"><li>• AddressData.gs</li><li>• AddressDataCopier.gs</li></ul></li><li>• gw.webservice.pc.pc800.gxmodel<ul style="list-style-type: none"><li>• AccountLocationModelEnhancement.gsx</li><li>• ContactModelEnhancement.gsx</li><li>• AddressModel.gs and AccountLocationModel.gs</li></ul></li><li>• gw.plugin.contact.ab800<ul style="list-style-type: none"><li>• ABContactAPISearchCriteriaEnhancement.gs</li><li>• DemoContactSystemPlugin*.gs</li><li>• AddressInfoEnhancement</li><li>• AccountLocationModelEnhancement.gs</li></ul></li><li>• gw.webservice.pc.pc800.ccintegration<ul style="list-style-type: none"><li>• CCContactGenerator.gs</li><li>• CCPolicyGenerator.gs</li><li>• CCAddressEnhancement</li><li>• PolicyLocationInfoEnhancement.gs</li></ul></li></ul></li></ul>
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.
<b>Jobs</b>	
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	<p>The signature of <code>IPolicyTermPlugin.calculatePeriodEnd(Date, TermType)</code> has been changed to <code>calculatePeriodEnd(Date, TermType, PolicyPeriod)</code>.</p> <p>The prior implementation of <code>calculatePeriodEnd</code> 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 <code>PeriodStart</code> of the first <code>PolicyPeriod</code> in the Policy to correct for date drifting.</p> <p>Now the plugin implementation in the base configuration calculates the <code>PeriodEnd</code> as follows:</p> <p>A policy incepting on day <math>X</math> of a month expires/renews on the same day unless:</p> <ul style="list-style-type: none"> <li>• The expiring month has fewer than <math>X</math> days – The period expires on the last day of the month.</li> <li>• 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.</li> </ul> <p>This fix adds a new Boolean <code>shouldPerformDateReconciliation</code> on Job. This Boolean is used by <code>PolicyTermPlugin.calculatePeriodEnd</code> method to determine if the policy period's job should be reconciled. By default this is <code>false</code>, however, <code>SubmissionImpl</code> and <code>RenewalImpl</code> overwrite this to <code>true</code>.</p>
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.
<b>Lines of Business</b>	
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 <code>ViewQuote</code> .
<b>Policy File</b>	
PC-17673	Fixed an issue where selecting a policy number from the <code>QuickJump</code> text box caused a null pointer exception when you navigated to the <code>Reinsurance</code> screen.
PC-21532	In previous releases, <code>PolicyPeriod.PolicyNumber</code> used the string literal <code>Unassigned</code> to indicate no policy number has been assigned to this period. In this release, the code was modified to use the <code>null</code> value.
PC-22567	Fixed an issue where PolicyCenter threw an exception if you tried to set the <code>Base State</code> on the <code>Policy Info</code> page to <code>None</code> . PolicyCenter now displays an error message if the base state is required.
<b>Product Model</b>	
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, <code>Boolean</code> and <code>Other</code> , in the <code>CovTermModelVal</code> typelist. This typelist is used by the <code>Value Type</code> field in the product model. In this release, these values ( <code>Boolean</code> and <code>Other</code> ) have been retired. <code>Boolean</code> coverage terms should instead be modeled as generic coverage terms instead of option coverage terms. Generic coverage terms do not have a <code>Value Type</code> to be set from the <code>CovTermModelVal</code> 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 <code>Value Types</code> of <code>Boolean</code> or <code>Other</code> 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 <code>CovTermModelVal</code> .
<b>Rating</b>	

ID	Description
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	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.
PC-19214	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.
<b>Reinsurance</b>	
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.
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.

ID	Description
<b>Teams</b>	
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.
<b>Search</b>	
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.
<b>Security</b>	
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.
<b>Upgrade</b>	
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.

ID	Description
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 <code>productdesigner.war</code> file, add translations, and then create a new <code>.war</code> file. The unpack and repack operations are automated by new scripts located in the <code>productdesigner/language-config</code> directory. A new <code>language.properties</code> file enables customers to configure available and default languages. The process is documented in a <code>readme.html</code> 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: <ol style="list-style-type: none"> <li>1. Add a pattern to a product.</li> <li>2. Disable the pattern in an offering.</li> <li>3. Revert the changes.</li> <li>4. Re-add the same pattern to the product.</li> </ol> <p>This issue has been fixed. Patterns that are removed and then re-added are now correctly enabled in offerings</p>
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.
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.

ID	Description
BCC-1992	<p>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:</p> <ol style="list-style-type: none"> <li>1. Must have an .html filename suffix.</li> <li>2. Must not contain Javascript.</li> <li>3. Must have absolute or relative image references using the tag &lt;img src="..."&gt;</li> <li>4. Must not have external link references.</li> <li>5. Must have relative Internal links using the tag "&lt;a href="..."&gt;"</li> <li>6. Must have the same file name as the page it describes. For example, the help file for Product Designer page "http://&lt;pd url&gt;/#PolicyLines()" must be named PolicyLines.html</li> <li>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.</li> </ol> <p>For more information, see the <code>readme.html</code> file located in the <code>productdesigner/language-config</code> directory. Also see the previous entry for BCC-1658.</p>
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
<b>Application Server</b>	
PL-27203	JBoss 6 deployment now works properly.
<b>Archiving</b>	
PL-26930	Modified the <code>restoreSeveredTransactionOffsetOnsetOnsetLinks</code> method so it can be called from Gosu.

ID	Description
<b>Batch Processes</b>	
PL-18406	<p>In releases prior to 8.0.0, you created custom batch processes by developing Gosu classes that implement the <code>BatchProcess</code> interface and subclass from the <code>BatchProcessBase</code> class. You configured custom batch processes in the <code>scheduler-config.xml</code> 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.</p> <p>Beginning with release 8.0.0, you also can create custom batch process by developing Gosu classes that subclass from the <code>WorkQueueBase</code> class. You configure this newer type of custom batch process in the <code>work-queue.xml</code> file, with definitions for its worker processes that run on other servers.</p> <p>Note: The methods on <code>WorkQueueBase</code> are called from different threads at different times on different servers. So, do not attempt to use class properties to configure or maintain state.</p>
<b>Bundles and Transactions</b>	
PL-22240	The <code>touch</code> method has been reimplemented. Please refer to the <a href="#">Gosudoc</a> for detailed usage.
<b>Cognos Integration</b>	
PL-21153	Thread usage on the LDAP server is now managed more efficiently.
<b>Command Line Tools</b>	
PL-27378	Removed <code>debug-start</code> from command line option.
PL-28544	The obsolete command line command <code>copy-theme</code> has been removed.
<b>Configuration Upgrade</b>	
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 <code>ClaimContactInput</code> widgets that use <code>postOnChange</code> , or if you modified a base <code>ClaimContactInput</code> widget by adding <code>postOnChange</code> . In these cases, the <code>postOnChange</code> would not get upgraded to the new syntax.
<b>Data Distribution</b>	
PL-25702	The <b>Data Distribution</b> page enables <b>Download comparison Zip</b> file and <b>Download Combined Zip</b> file options only if two executions are selected. In prior versions, the options were erroneously enabled even if there were not two executions selected.
<b>Database Configuration</b>	
PL-28656	Fixed a problem that prevented the <code>DBAuthenticationPlugin</code> from working.
<b>Database Support</b>	
PL-24417	The performance requirement to set the <code>action="delete"</code> attribute for <code>pc_message</code> in <code>database-config.xml</code> has been removed.
PL-28323	Fixed a bug which prevented scheduling of the Database Statistics process.
<b>Database Support – Oracle</b>	
PL-27438	Added support for Oracle Date interval partitioning.
PL-28085	The Oracle fast add column feature is used when adding monetary amount columns to an entity or a subtype.
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 <i>PolicyCenter Installation Guide</i> .
<b>Database Upgrade</b>	
PL-27016	Guidewire has added support for using Oracle's parallel DDL execution feature during upgrade. The <code>createIndexInParallel</code> attribute of the <code>&lt;upgrade&gt;</code> element has been replaced with the new <code>degree-parallel-ddl</code> attribute. See “Configuring Parallel DML and DDL Statement Execution” in the <i>Upgrade Guide</i> .

ID	Description
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 <i>Upgrade Guide</i> .
PL-28449	Fixed a bug that prevented handling of statistics on the ID column properly.
<b>Document Management</b>	
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.
<b>Email</b>	
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.
<b>Entities/Metadata</b>	
PL-19023	The <i>Data Dictionary</i> has been modified to show references for subtypes
PL-24743, PL-27611	<i>Data Dictionary</i> 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 <i>Data Dictionary Data Entity View</i> .
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.
<b>Globalization</b>	
PL-26606	Updated GroupUserSearchDV.pcf to support Japanese kanji fields in the standard way represented elsewhere in the application.
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,
	<pre>&lt;GWLanguge   code="de_DE"   name="German (Germany)"   typecode="de_DE"&gt;   &lt;ExtJsSettings&gt;     &lt;LabelWidth size="220" /&gt;   &lt;/ExtJsSettings&gt; &lt;/GWLanguge&gt;</pre>
<b>Gosu</b>	
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.

ID	Description
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.
<b>IntelliJ IDE – Compiler</b>	
PL-28346	Fixed a compilation error when compiling an entity or its extension if the entity had a subtype.
<b>IntelliJ IDE – Debugger</b>	
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 <b>Enhance Entities Visualization</b> .
<b>IntelliJ IDE – Display Key Editor</b>	
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.
<b>IntelliJ IDE – Entity Editor</b>	
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.
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.
<b>IntelliJ IDE – Gosu Editor</b>	
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 <i>Gosu Reference Guide</i> 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.

ID	Description
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: <pre>for (x in 10) {...}</pre> is illegal and must be upgraded with an interval such as: <pre>for (x in 0.. 10) {...}</pre> using the provided inspection in Studio.
<b>IntelliJ IDE – Line of Business Editor</b>	
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.
PL-26971	Fixed an issue in Studio where removing a typecode from its parent also incorrectly removed it from all of its other parents.
<b>IntelliJ IDE – Other</b>	
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.
<b>IntelliJ IDE – PCF Editor</b>	
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.
<b>IntelliJ IDE – IntelliJ IDE – Plugins, OSGi, Plugins</b>	
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 <i>Integration Guide</i> .
<b>IntelliJ IDE – Refresh</b>	
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.
<b>IntelliJ IDE – Typelist Editor</b>	
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.

ID	Description
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.
<b>IntelliJ IDE – Web Services Editor</b>	
PL-26576	Added a check to the timeout value of web services to insure that it is lower than Studio's maximum integer.
<b>Integration</b>	
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 <code>integrationservice</code> and then do the rest of the configuration in the new file <code>inbound-integration-config.xml</code> . Also, the API details for file and JMS integrations changed. There is a new plugin interface called <code>InboundIntegrationHandlerPlugin</code> . Also, the file integration now supports processing one file at a time, rather than one line at a time. For details, refer to the <i>Integration Guide</i> .
<b>Other – Persistence</b>	
PL-25820	Fixed an issue with whitespace not being trimmed by trimming unicode full-width whitespace.
<b>Profiling</b>	
PL-28172	Fixed an issue where the <code>Purge Profiler Data</code> batch job would throw exceptions when purging web services profiling data.
<b>Queries</b>	
PL-18578	<p>This release adds a new class, <code>gw.api.database.QuerySelectColumns</code>, 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 <code>select</code> method, you pass a list of <code>IQuerySelectColumn</code> objects, which you construct by using static methods on the <code>QuerySelectColumns</code> class. Each <code>IQuerySelectColumn</code> object represents a column in the result.</p> <p>For example, the following Gosu sample code creates a row query, with <code>Address</code> as the primary entity. The result includes only the <code>City</code> column for <code>Address</code> instances that match the query criteria.</p> <pre> 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))}) </pre> <p>The <code>QuerySelectColumns</code> 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 <code>Address</code> instances.</p> <pre> 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 addressQuery.withDistinct(true) // Always run aggregate queries with Distinct set to true. addressResult = addressQuery.select({QuerySelectColumns.dbFunction( DBFunction.Count(Paths.make(Address#City))) }) </pre>

ID	Description
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.
<b>Search</b>	
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.
<b>Web - ListViews</b>	
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.
<b>Web – UI/Runtime</b>	
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.
<b>Work Queues</b>	
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.
<b>XMLElement (and XSD types)</b>	
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
- Product Designer Known Issues
- Studio/Platform Known Issues

**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 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.

#### 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.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 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 subtentities 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 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.

## 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 `@WsPermission` 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.

