Guidewire Software Upgrade Guide

(The only document you’ll ever need to read to have a successful Guidewire Product upgrade!)

Hari Vadlamudi

Table of Contents

[Getting Started 3](#_Toc471413954)

[Guidewire Software Upgrade Strategy 4](#_Toc471413955)

[Guidewire software upgrade process and Platform requirements 4](#_Toc471413956)

[ClaimCenter 9 Platform Matrix 4](#_Toc471413957)

[AMS Claims Software Upgrade Strategy 5](#_Toc471413958)

[Successful Upgrade Strategy 5](#_Toc471413959)

[AMS Claims Software Upgrade Assessment 6](#_Toc471413960)

[Upgrade Assessment 6](#_Toc471413961)

[AMS Claims Software Upgrade Process 7](#_Toc471413962)

[Agile Approach 7](#_Toc471413963)

# Getting Started

**Why Upgrade?**

Upgrades are critical for every software product. GW product upgrade brings several benefits such as new features, performance improvements, process optimization, security vulnerability fixes, ongoing vendor support, and integrations with the latest applications and DB servers.

Thus, the question is whether or not to upgrade, but rather when and how to do so most effectively. Each version of the application presents performance improvements and process optimization opportunities based on new functionality and technical enhancements.

In Alternative Market Segment, we are using ClaimCenter v7.0.3 and Contact Manger v7.0.3.The latest GW ClaimCenter release available in the market is v 9.x.x. To make most of the upgrade, it is necessary to look into the new features/improvements provided in the target version. Let’s dive in.

ClaimCenter 9 features a number of improvements that support claims management objectives. Here are some of the key features and benefits.

KEY FEATURES

* Business rules for activities - Enables business users with the appropriate permissions to change what activities are created on a claim, when they are created, and to whom they are assigned – all without coding or the need to wait for an IT release cycle.
* Financial configurability - Claims adjuster and claims management both have access to the appropriate financial information at the appropriate level of granularity. These changes result in reduced implementation effort and time to market.
* Expanded cloud deployment
* Straight-through invoice processing - Invoices are automatically associated with the appropriate claim, and the business can decide what invoices should be auto-paid via rules. If an invoice does not meet the criteria, it is flagged for the adjuster to manually approve and pay.
* Vendor documents - Adjuster can use vendor documents capabilities within the claim to view documents associated at the vendor level, such as service-level agreements or pricing.
* Improved subrogation capabilities - Offers better automated identification of potential subrogation opportunities and the ability to select which exposures should be subrogated, which means company guidelines are more consistently followed. The ability to assign a subrogation owner provides improved tracking and gives subrogation team supervisors better insight into team workload and performance.

KEY BENEFITS

* Increased productivity through streamlined and automated processes
* Reduced leakage
* Increased business agility
* Improved visibility and tracking of key claims metrics
* Optimized IT workflows
* Scalability for future growth

# Guidewire Software Upgrade Strategy

## Guidewire software upgrade process and Platform requirements

Guidewire upgrades usually occur at fairly regular intervals, most likely every two to four years and GW supports only three latest versions of the GW products at any one time.

Guidewire uses major release numbers (numbered 7.0.0, 8.0.0, 9.0.0) to roll out important new features, improvements in integration capabilities, and support for newer versions of other technologies in the stack such as application servers and databases.

GW uses maintenance releases such as 7.0.3, 9.0.1, to address bugs, incorporate improvements from ongoing performance testing, and introduce minor enhancements between major releases. These releases typically are made available roughly every three months after a major release for about a year, and then less frequently as needed after that.

## ClaimCenter 9 Platform Matrix

This list enumerates Guidewire's supported infrastructural components. Guidewire performs functionally tests with several third party products to provide various infrastructure options to customers. While selecting third party products, Guidewire looks at the industry adoption rate, life-cycle of each upcoming release, and also takes product end-of-life events into account.

Table 1. ClaimCenter 9 Platform Requirements

|  |  |
| --- | --- |
| Type | Supported Product and Version |
| **Production** | |
| Operating System | AIX 7or higher  Windows 2012 or higher |
| Application Server | IBM WebSphere Application Server or ND 9  Apache Tomcat 8.0.x, x>=23  JBoss Enterprise Application Server (6.4.5)  Oracle Weblogic Server 12c (12.2.x, x>=1) |
| Database Server | Microsoft SQL Server Enterprise 2014 SP1  Oracle 12.1.0.2 Enterprise RDBMS/ Oracle 12.1.0.2 Enterprise RAC |
| JDK | Hotspot 8 JDK 64bit 1.8.0\_x,x>=92 / IBM JDK 8 |
| **Development** | |
| Operating System | Microsoft Windows 7 SP1 (64 bit) / Windows 8 (64 bit) / Windows 10 (64 bit) |
| Application Server | Jetty (Bundled) |
| Database | H2 1.2.147 (Bundled) |
| Studio | JetBrains Intellij 15.0.6 Community Edition (Bundled) / JetBrains  Intellij 15.0.x, x>=6 Ultimate Edition (not included) |
| **End User** | |
| Operating System | Microsoft Windows 7 SP1 (64 bit recommended)/ Windows 8 / Windows 10 |
| Web Browser | IE 11, Microsoft Edge, Chrome, Firefox |

# AMS Claims Software Upgrade Strategy

## Successful Upgrade Strategy

"Smaller, Faster, Smarter"

AMS is relying on advanced and fully featured GW ClaimCenter technology platform to support critical operations in claims. To take the full advantage of the platform and maintain leading-edge, it is necessary to keep the product up to date with all the new features and fixes. In fact, those firms that complete successful upgrades based on a holistic approach and proven accelerators are more likely to maximize returns on technology investments and gain competitive advantage.

It is possible to have smaller or manageable list of changes with every major upgrade if all the intermediate maintenance releases are already applied to the product. With this approach, the time needed for testing shortens and the product can be promoted to market quickly.

Of course, the enhancements brought into the system with every upgrade may impact the claims system functionality. To ensure the functionality, the application needs to be completely tested after every upgrade. Since this is a repetitive process, more focus should be given towards automating this process as much as possible which will lead us to have a smarter upgrade process.

As we move forward, here's how best practices need to change. Instead of "if it's not broke, don't fix it" we need to move to an "if there's a release, perform upgrade assessment no matter what” mentality.

No doubt this is a hassle. But we're now in a "pay now or pay later" world when it comes to upgrades. It's going to be a hassle, either way. Performing the upgrade assessment on every maintenance/major release is a smarter approach as it allows us to take a calculated decision based on several factors such as feature set, cost-benefit analysis, major releases in near future and any issues with dependent products/technologies.

# AMS Claims Software Upgrade Assessment

## Upgrade Assessment

To determine whether the upgrade process is going to be a smooth, less time consuming or rough, long-running process, we need to look at several factors, including:

* Is the new version needs new hardware or software?
* Is the current Prod Application has too many custom features?
* Are there any features available in new version that conflict with custom changes?
* Is it possible to leverage tools/automation approaches for upgrade?
* Are there too many changes between current base and new base versions?
* Any impact on downstream systems or vendor implementations?
* Is the ETL or BI module needs to be changed?
* Is the statutory reporting module needs to be changed?
* Is the Medical Bill Review process needs to be changed?
* Are there any changes in business/process flows?

**Steps to Take:**

* Review current business processes against the OOTB processes
* Review current backlog of functional requirements/enhancements
* Assess business rules against the new versions
* Study and assess available new features
* Determine if it is possible to align current and future state requirements and processes
* Ensure reporting and data warehouse requirements align to the new data model

**Potential challenges along the way:**

* Business processes are not documented well or are out of date
* Business rules are embedded in the UI and not easy to review
* Formal design documents are inadequate, out of date, or non-existent
* Current application is not well-documented
* Refactoring Gscript code to GOSU
* Aligning integrations with new APIs and plug-ins
* Consistency of administration data
* New roles/permissions
* New business tables

**Start now:**

* Identify the core business and IT team members
* Review the Guidewire ClaimCenter process and tools needed for upgrade
* Review the Infrastructure related changed needed to support the new version

# AMS Claims Software Upgrade Process

## Agile Approach

A solid foundation is needed for a super structure. Similarly, while upgrading the GW 7 to GW 9, it is necessary to upgrade the existing production application to the target version without adding new components, process flows and features to establish a base that can be easily regression tested and corrected if needed.

When all the GW 7 features are working on GW 9 version, the new features and process flows are added to the application based on the priority order and the application is promoted to Production in incremental releases. It is easier to test and debug smaller feature set during each iteration. This allows more targeted and rigorous testing of each feature within the overall product.

Upgrade Steps

1. Create a Configuration Backup
2. Update Infrastructure
3. Install and Test GW 9 Base
4. Compare and Merge custom configuration changes with GW9
5. Compare and Merge Data Model Changes
6. Compare and Merge Rules
7. Validate and Resolve Errors in Configuration
8. Export Admin Data
9. Purge unused /address correction/completed workflows and logs
10. Run DB Validations
11. Create a Database Backup
12. Prepare DB for upgrade with all pre-requisites
13. Implement Data Model Upgrade Plugin to perform DB operations during upgrade
14. Disable Scheduler
15. Suspend Message Destinations
16. Configure DB for upgrade
17. Start Server and Perform auto DB Upgrade
18. Run DB Validations
19. Back up the upgraded DB
20. Validate Integrations and Configurations