Red Hat Application Migration Toolkit 4.0

Getting Started Guide

Simplify Migration of Java Applications

Red Hat Application Migration Toolkit 4.0 Getting Started Guide

Simplify Migration of Java Applications

Legal Notice

Abstract

This guide provides information to help you get started with the Red Hat Application Migration Toolkit.

Table of Contents

CHAPTER 1. INTRODUCTION

1.1. ABOUT THE GETTING STARTED GUIDE

This guide is for engineers, consultants, and others who want to use Red Hat Application Migration Toolkit (RHAMT) to migrate Java applications or other components. It provides an overview of the Red Hat Application Migration Toolkit and how to get started using the tools to plan and execute your migration.

CHAPTER 2. ABOUT RED HAT APPLICATION MIGRATION TOOLKIT

What is Red Hat Application Migration Toolkit?

Red Hat Application Migration Toolkit (RHAMT) is an extensible and customizable rule-based tool that helps simplify migration of Java applications.

RHAMT examines application artifacts, including project source directories and application archives, then produces an HTML report that highlights areas needing changes. RHAMT can be used to migrate Java applications from previous versions of *Red Hat JBoss Enterprise Application Platform* or from other containers, such as *Oracle® WebLogic Server* or *IBM® WebSphere® Application Server*.

How Does Red Hat Application Migration Toolkit Simplify Migration?

Red Hat Application Migration Toolkit looks for common resources and highlights technologies and known trouble spots when migrating applications. The goal is to provide a high-level view into the technologies used by the application and provide a detailed report organizations can use to estimate, document, and migrate enterprise applications to Java EE and Red Hat JBoss Enterprise Application Platform.

2.1. RHAMT FEATURES

Red Hat Application Migration Toolkit (RHAMT) provides a number of capabilities to assist with planning and executing migration projects.

Planning and work estimation

RHAMT assists project managers by detailing the type of work and estimation of effort to complete the tasks. Level of effort is represented in RHAMT reports as story points. Actual estimates will be based on the skills required and the classification of migration work needed.

Identifying migration issues and providing solutions

RHAMT identifies migration issues and highlights specific points in the code where an issue occurs. RHAMT suggests code changes and provides additional resources to help engineers resolve the specific issue.

Detailed reporting

RHAMT produces numerous reports to give both high-level views of the migration effort and details of specific migration tasks. You can view migration issues across all applications, charts and summary information about issues in an application, a breakdown of issues by module in the application, reports of technologies used, and dependencies on other applications and services. You can also examine source files to see the line of code where an issue occurs. See the *CLI Guide* for more information on the available RHAMT reports.

Built-in rules and migration paths

RHAMT comes with a core set of rules to provide migration assistance for several common migration paths. These rules identify the use of proprietary functionality from other application servers or deprecated subsystems from previous versions of JBoss EAP. RHAMT also contains rules to identify common migration issues, such as hard-coded IP addresses and JNDI lookups.

Rule extensibility and customization

RHAMT provides the ability to create powerful and complex rules. You can expand upon the core set of rules provided by RHAMT and create rules to identify additional issues that are important for your migration project. You can also override core rules and create custom rule categories. See the *Rules Development Guide* for more information on customizing RHAMT rules.

Ability to analyze source code or application archives

RHAMT can evaluate application archives or source code, and can evaluate multiple applications together. It can identify archives that are shared across multiple applications, which can help with more accurate effort estimation.

2.2. RHAMT RULES

Red Hat Application Migration Toolkit (RHAMT) contains rule-based migration tools that analyze the APIs, technologies, and architectures used by the applications you plan to migrate. In fact, the RHAMT analysis process is implemented using RHAMT rules. RHAMT uses rules internally to extract files from archives, decompile files, scan and classify file types, analyze XML and other file content, analyze the application code, and build the reports.

RHAMT builds a data model based on the rule execution results and stores component data and relationships in a graph database, which can then be queried and updated as needed by the migration rules and for reporting purposes.

RHAMT rules use the following rule pattern:

```
when(condition)
    perform(action)
otherwise(action)
```

RHAMT provides a comprehensive set of standard migration rules out-of-the-box. Because applications may contain custom libraries or components, RHAMT allows you to write your own rules to identify use of components or software that may not be covered by the existing ruleset. If you plan to write your own custom rules, see the *Rules Development Guide* for detailed instructions.

CHAPTER 3. SUPPORTED CONFIGURATIONS

3.1. SUPPORTED MIGRATION PATHS

RHAMT supports application migration from several platforms to Red Hat JBoss Enterprise Application Platform (JBoss EAP). See the below table for which JBoss EAP version is currently supported by RHAMT for direct migration from your source platform.

Source Platform	Migration to JBoss EAP 6	Migration to JBoss EAP 7
Oracle® WebLogic Server	✔	✔
IBM® WebSphere® Application Server	✔	✔
JBoss EAP 4	✔	✘ [a]
JBoss EAP 5	✔	✔
JBoss EAP 6	N/A	✔

[a] Although RHAMT does not currently provide rules for this migration path, Red Hat Consulting can assist with migration from any source platform.

3.2. SYSTEM REQUIREMENTS

3.2.1. Software

- Java Platform, JRE version 8+

- RHAMT is tested on Linux, Windows, and macOS. Other operating systems with Java 8+ support should work equally well.

3.2.2. Hardware

The following memory and disk space requirements are the minimum needed to run RHAMT. If your application is very large or you need to evaluate multiple applications, you may want to increase these values to improve performance.

For tips on how to optimize performance, see the Optimize RHAMT Performance section of the RHAMT *CLI Guide*.

- A minimum of 4 GB RAM. For better performance, a quad-core processor with 8 GB RAM is recommended. This allows 3 - 4 GB RAM for use by the JVM.

- A minimum of 4 GB of free disk space. A fast disk, especially a solid-state drive (SSD), should improve performance.

CHAPTER 4. ABOUT THE TOOLS

Red Hat Application Migration Toolkit (RHAMT) provides several tools to assist you in the various stages of your migration and modernization efforts. Review the details of each tool to determine which is right for your project.

- CLI

- Web Console

- Eclipse Plugin

4.1. ABOUT THE CLI

The CLI is a command-line tool in the Red Hat Application Migration Toolkit that allows users to assess and prioritize migration and modernization efforts for applications. It provides numerous reports that highlight the analysis results. The CLI is for advanced users with highly customized needs looking for fine-grained control of RHAMT analysis options or to integrate with external automation tools.

For more information on using the CLI, see the RHAMT *CLI Guide*.

4.2. ABOUT THE WEB CONSOLE

The web console for Red Hat Application Migration Toolkit is a web-based system that allows a team of users to assess and prioritize migration and modernization efforts for a large number of applications. It allows you to group applications into projects for analysis and provides numerous reports that highlight the results.

For more information on using the web console, see the RHAMT *Web Console Guide*.

4.3. ABOUT THE ECLIPSE PLUGIN

The Eclipse plugin for Red Hat Application Migration Toolkit provides assistance directly in Eclipse and Red Hat JBoss Developer Studio for developers making changes for a migration or modernization effort. It analyzes your projects using RHAMT, marks migration issues in the source code, provides guidance to fix the issues, and offers automatic code replacement when possible.

For more information on using the Eclipse plugin, see the RHAMT *Eclipse Plugin Guide*.

CHAPTER 5. PLANNING YOUR APPLICATION MIGRATION

5.1. GOALS OF ASSESSING A MIGRATION

Many organizations are faced with the challenges of keeping their existing business operations running, while also trying to innovate. There are typically increased expectations to deliver new functionality faster and to reduce cost.

When assessing a migration or modernization effort, it is important to address the challenges that are specific to your organization. Common challenges that organizations face are limited budgets, lack of in-house skills, perceived risks, no known predictable process, and accurate effort estimation in a timely manner.

In general, the goals when assessing a large-scale middleware migration or modernization effort are to:

- Predict the level of effort and cost.

- Schedule the application migrations and handle conflicts.

- Identify all potential risks at a code, infrastructure, process, or knowledge level.

- Predict the return on investment to make the business case.

- Identify and mitigate risks to the business.

- Minimize disruption to existing business operations.

Red Hat has developed a strategy to minimize risk and make application migration and modernizations more efficient. See Red Hat's Application Migration Approach for more information.

5.2. RED HAT'S APPLICATION MIGRATION APPROACH

Red Hat has defined a strategy to make large-scale application migrations to Red Hat JBoss Middleware quantifiable, less costly, less risky, and easier to complete.

Migrating from proprietary or outdated middleware platforms to Red Hat JBoss Middleware gives you lightweight, modular, and cloud-ready middleware with a supporting application infrastructure to make teams more productive. JBoss Middleware and other Red Hat technologies also provide great opportunities to modernize your application development and delivery, allowing you to innovate more quickly.

5.2.1. Best Practices

Red Hat recommends the following practices when planning and executing a migration or modernization:

- Create a centralized collaboration platform for information sharing.
  Comprehensive and accessible documentation is important so that knowledge can be easily shared and to ensure that effort is not duplicated by solving the same issue twice. It is recommended to document the following items:

  - A step-by-step guide for migrating or modernizing an application from scratch.

  - A collection of solutions to known and encountered issues.

- - Information about the new platform.

  - A record of the changes made to specific pilot projects.

- Reuse, automate, and standardize as much as possible.
  Consider reusing existing components instead of creating new ones. Automate processes related to the application lifecycle, such as builds, configuration, deployment, and tests. Standardize and document software packaging format, configuration management, libraries, and dependencies.

- Use a proven and repeatable methodology.
  The recommendation is to follow a practical approach and make as few changes as possible to get a functionally identical application. See Migration Methodology below for more details on using this methodology.

- Involve Red Hat technical expertise early for the chosen Red Hat Middleware components.
  This is crucial to make your migrations and modernizations low risk, predictable, and efficient. Contact Red Hat Consulting for assistance.

5.2.2. Methodology

Red Hat's recommended methodology is a proven, scalable approach that helps you to incrementally plan and conduct a migration or modernization.

Figure 5.1. Red Hat Migration Methodology
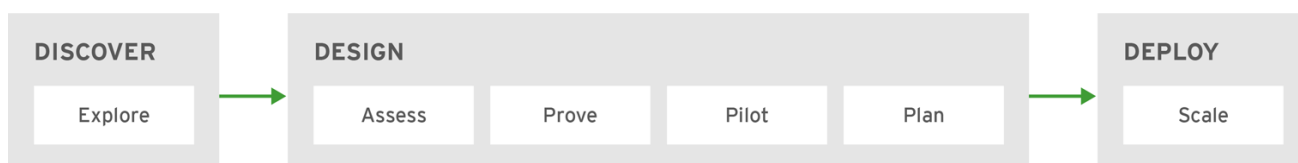
The approach consists of the following phases:

Discover

Explore the technologies and identify concerns, organization requirements and challenges. Discuss options and potential approaches. See Discover Phase for more details.

Design

Identify and mitigate the biggest risks from an application, infrastructure, process, and knowledge perspective. Establish a strategy for the solution. Prove the technology and build the business case. See Design Phase for more details.

Deploy

Scale the modernization or migration iteratively by executing the previously defined strategy according to a well-defined delivery model. See Deploy Phase for more details.

5.2.2.1. Discover Phase

Figure 5.2. Red Hat Migration Methodology: Discover Phase
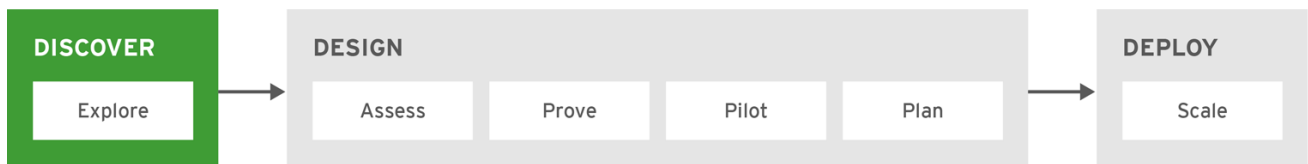
The *Discover* phase is when you gather all stakeholders and decision-makers together to understand the current state and business drivers, and to determine the migration or modernization needs.

In this phase, you explore technologies and discuss potential approaches. Identify the existing pain points, concerns, requirements, and some potential challenges. Define the high-level business priorities and scope the assessment. Determine in what ways you want to modernize your application development and delivery to allow you to innovate more rapidly.

Typically, this can be covered in a day in a workshop with Red Hat experts.

5.2.2.2. Design Phase

Figure 5.3. Red Hat Migration Methodology: Design Phase
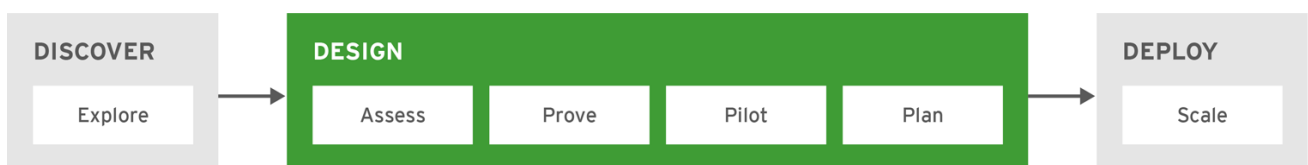
The *Design* phase is when you identify all risks, define a strategy and target architecture, prove the technology, and build the business case. This phase consists of the following steps:

Assess

Examine the existing infrastructure, architecture, technologies, and applications. Identify dependencies, knowledge, processes, and lifecycles. Define the desired infrastructure, architecture, technologies, and applications. Determine the feasibility and potential risks. Draft a plan and provide rough estimates.

Analyzing applications using the Red Hat Application Migration Toolkit web console or CLI helps to determine dependencies, potential risks, and relative effort. See the *Web Console Guide* or *CLI Guide* for information on how to use these tools.

Prove

Solve and document the identified technical risks, for example, high risk items and issues with unknown effort. Build and shape the new platform infrastructure. Refine the estimates based on the outcomes.

Pilot

Choose a small set of representative applications to transform. Finalize the target platform infrastructure as necessary. Update the documentation as the process is fine-tuned. This pilot consists of one or several iterations of the migration execution, which is scaled during the Deploy phase.

Use the Red Hat Application Migration Toolkit Eclipse plugin to accelerate the code migration. See the *Eclipse Plugin Guide* for information on how to use the Eclipse plugin.

Plan

Based on the outcomes of the previous steps, sharpen the estimates and refine the project plan. Define the rollout strategy to be used in the Deploy phase to complete the migration. Prepare and schedule the relevant technical enablement courses .

5.2.2.3. Deploy Phase

Figure 5.4. Red Hat Migration Methodology: Deploy Phase

RHAMT_446947_0617

The *Deploy* phase is when you execute the plan created in the Design phase. In this phase, you scale the overall transformation process to complete the plan and bring all applications to their new production environment.

The plan is executed in iterations to deliver value incrementally. With each iteration, you continuously validate against the plan and document findings to improve the next sprint.

The use of the Red Hat Application Migration Toolkit Eclipse plugin increases speed in each iteration. It can be used with Eclipse or JBoss Developer Studio, and marks migration issues directly in the source code, provides inline hints, and offers code change suggestions. See the *Eclipse Plugin Guide* for information on how to use the Eclipse plugin.

APPENDIX A. REFERENCE MATERIAL

A.1. ADDITIONAL RESOURCES

A.1.1. Get Involved

To help make Red Hat Application Migration Toolkit cover most application constructs and server configurations, including yours, you can help with any of the following items.

- Send an email to jboss-migration-feedback@redhat.com and let us know what RHAMT migration rules should cover.

- Provide example applications to test migration rules.

- Identify application components and problem areas that may be difficult to migrate.

  - Write a short description of these problem migration areas.

  - Write a brief overview describing how to solve the problem migration areas.

- Try Red Hat Application Migration Toolkit on your application. Be sure to report any issues you encounter.

- Contribute to the Red Hat Application Migration Toolkit rules repository.

  - Write a Red Hat Application Migration Toolkit rule to identify or automate a migration process.

  - Create a test for the new rule.

  - Details are provided in the *Rules Development Guide*.

- Contribute to the project source code.

  - Create a core rule.

  - Improve RHAMT performance or efficiency.

  - See the *Core Development Guide* for information about how to configure your environment and set up the project.

Any level of involvement is greatly appreciated!

A.1.2. Important Links

- RHAMT forums: https://developer.jboss.org/en/windup

- RHAMT JIRA issue trackers

  - Core RHAMT: https://issues.jboss.org/browse/WINDUP

  - RHAMT Rules: https://issues.jboss.org/browse/WINDUPRULE

- RHAMT mailing list: jboss-migration-feedback@redhat.com

- RHAMT on Twitter: @JBossWindup

- RHAMT IRC channel: Server FreeNode (`irc.freenode.net`), channel #**windup** (transcripts).

A.1.3. Known RHAMT Issues

You can review known issues for RHAMT here: Open RHAMT issues.

A.1.4. Report Issues with RHAMT

Red Hat Application Migration Toolkit uses JIRA as its issue tracking system. If you encounter an issue executing RHAMT, please file a JIRA Issue.
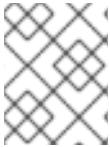
> **NOTE**
>
> If you do not have one already, you must sign up for a JIRA account in order to create a JIRA issue.

A.1.4.1. Create a JIRA Issue

1. Open a browser and navigate to the JIRA Create Issue page.
   If you have not yet logged in, click the **Log In** link at the top right side of the page and enter your credentials.

2. Choose the following options and click the **Next** button.

   - **Project**
     For core RHAMT issues, choose *Red Hat Application Migration Toolkit (WINDUP)*

     For issues with RHAMT rules, choose: *Red Hat Application Migration Toolkit rules (WINDUPRULE).*

   - **Issue Type**: *Bug*

3. On the next screen complete the following fields.

   - **Summary**: Enter a brief description of the problem or issue.

   - **Environment**: Provide the details of your operating system, version of Java, and any other pertinent information.

   - **Description**: Provide a detailed description of the issue. Be sure to include logs and exceptions traces.

   - **Attachment**: If the application or archive causing the issue does not contain sensitive information and you are comfortable sharing it with the RHAMT development team, attach it to the issue using the **browse** button.

4. Click the **Create** button to create the JIRA issue.

Revised on 2018-04-04 12:21:18 EDT