Migrating WebSphere Application to IBM Cloud Private Using Transformation Advisor and Microclimate

Table of Contents

Overview

IBM Transformation Advisor can be used to migrate your WebSphere applications to Liberty running on IBM Cloud Private. The application binary file and other dependency files, such as DB2 or MQ drivers, are needed for the application ICP image build process. There are two ways to get access to those necessary binaries:

- Upload into IBM Transformation Advisor
- Provide the Maven artifact location information

The first approach accelerates a simple application migration when no code changes are needed by allowing Transformation Advisor to quickly move your application into IBM cloud Private.

However, the best practice is to get the binaries from a binary repository rather than a source repository. Transformation Advisor now supports Maven repositories for binary artifacts. Transformation Advisor continues to use GIT for the source files it generates to build your ICP image. Now, you can use Transformation Advisor to define your binary files' Maven repository locations so that the build uses the application binaries directly from your build systems.

For an application that requires code changes, Transformation Advisor provides the code modification recommendations and effort estimation. If you use Eclipse, we recommend you use the WebSphere Application Migration Toolkit (WAMT) Eclipse plugin to help modify the application code. After you build

your new application archive, you will be able to publish your application to your Maven repository and have the result integrated with the Microclimate build.

It's worth mentioning that the repository types we support by default are:

- HTTP server backed Maven repository
- Artifactory repository manager that allows anonymous user access

At the moment of the current release, authentication is not supported. Nexus repository is not supported by default, but you may manually modify the build files such as Jenkinsfile or pom.xml to use Nexus.

Also, the repository type is not limited to Maven repository only. You can manually modify the build files in the migration bundle to pull them from other types of repository. For example, you can change the pom.xml to download from a remote HTTP server, or you can change the Dockersfile to move the downloaded artifacts to some customized file location other than the default.

This document describes how to plugin users' Maven build system into Transformation Advisor by default.

Prerequisites

Prerequisites are listed at the beginning of each step. The following is the complete set of installation prerequisites required to get started.

- An instance of IBM Cloud Private
- An installation of Transformation Advisor in IBM Cloud Private
- An installation of Microclimate in IBM Cloud Private
- The data collector results are loaded into IBM Transformation Advisor with migration recommendations
- Eclipse with the WAMT plugin to make the necessary code modifications. WAMT can be obtained from *https://developer.ibm.com/wasdev/downloads/#asset/tools-WebSphere_Application_Server_Migration_Toolkit*. If you prefer a different IDE, use the detailed analysis report from Transformation Advisor to make your code changes.
- An implementation of Git (GitHub/GitLab) or access to a public implementation (GitHub/GitLab)
    - A user with permissions to create a repository, clone a repository & update a repository in the above implementation
- Maven repositories where the application binary and other dependencies can be downloaded from; make sure the repositories are accessible from Microclimate
- A Liberty server to test the modified applications

Step 1: Modify applications based on migration recommendations

In this Step you will modify your applications based on the migration recommendations provided by Transformation Advisor.

Prerequisites

1. The applications scan result are loaded into IBM Transformation Advisor and migration recommendation are suggested
2. Eclipse with WAMT plugin to make the necessary code modification.
3. A Liberty server to test the modified applications
4. Maven repositories where the newly built application binary and other dependencies can be published. Make sure the repositories are accessible from Microclimate so that the binaries can be downloaded during the Microclimate build.

Tasks

1. Modify application code according to the migration recommendations. It's suggested to use IBM® WebSphere® Application Server Migration Toolkit (WAMT). The Eclipse tool has the same migration advice provided by the binary scanner used by Transformation Advisor except that it operates on the source code to make your updates easier. More information about the tool can be found on the site: *https://developer.ibm.com/wasdev/downloads/#asset/tools-WebSphere_Application_Server_Migration_Toolkit*.
2. Deploy your modified application to a local Liberty server for testing and make sure the application works as expected.
3. You might need several iterations of above steps to get the final version of the modified application.
4. Publish the modified application binary files to the Maven repository. If other dependency files such as DB2 or MQ drivers are needed to run your application, make sure they are available in the same or other accessible Maven repositories.

Step 2: Generates the migration bundle that will download the dependency files from the Maven repository

In this Step you will provide the Maven repository information for application and other required dependency binary files. Transformation Advisor will generate the migration bundle using the Maven information you provide.

Prerequisites

- Step1 is complete. The application and other dependency files are available in the Maven repository Microclimate can access.

Tasks

1. In the Transformation Advisor UI, go to the *Recommendations* page, click *Migration Plan* button for your application. This opens the Migration page.

   The Migration page lists the artifacts that Transformation creates as well as the application binary or other dependencies. For each dependency, enter the dependent file location URL in your Maven repository. An Artifactory Maven repository file URL may look like:

http://<REPO_IP>:8081/artifactory/libs-release-local/<GroupId_PATH1>/<GroupId_PATH2>/.../<GroupId_PATHn>/<ArtifactId>/<Version>/<FileName>.<PackageType>

2. Once all dependency information is provided, Transformation Advisor generates the migration bundle accordingly. Transformation Advisor adds the Maven repositories to pom.xml and generates the build commands to download the dependency binaries from your Maven repositories. Those binary files will be used to build your application docker image.

Step 3: Set up Jenkins pipeline

In this Step you will push the Migration bundle to your GIT repository, create a Microclimate project with a Jenkins pipeline links to it.

Prerequisites

1. Completed step 2
2. An implementation of Git (GitHub/GitLab) or access to a public implementation (GitHub/GitLab)
    - A user with permissions to create a repository, clone a repository & update a repository in the above implementation
3. An instance of IBM Cloud Private
4. An installation of Microclimate in IBM Cloud Private

Tasks

1. In the Application Migration page, click *Deploy Bundle* and fill in the GIT repository and Microclimate URL information.
2. Click the *Set up pipeline* button (TODO: button name probably will be different). Transformation advisor will upload the migration bundle into the GIT repository, then create a Microclimate build project with a Jenkins pipeline linked to your GIT repository.
3. You may need to update some configuration files in Git repository. For example, fill in passwords needed in the server.xml.
4. Transformation generated pom.xml in the migration bundle already has the tasks below:
    - Include repository access to your provided Maven repositories
    - Download application binary and other dependency files
    - Move files to the required location so that they will be copied to the required location in the Docker image build process.
5. You may modify pom.xml to include your application specific build operations. For example, if your dependency files are available in a remote repository other than Maven repository, you may refer to Apache Maven website (*https://maven.apache.org/index.html*) to write your specific downloading tasks.
6. Commit the changed migration bundle files to your GIT repository.

Step 4: Deploy the application to IBM cloud Private

In this Step you will create the application deployment in IBM cloud Private.

Prerequisites

1. Completed step 3
2. A Microclimate project is created and the associated Jenkins pipeline to the GIT repository is available

Tasks

1. Log into Microclimate UI and open the project created in step 3.
2. Click the *Add Deployment* button to trigger a Jenkins pipeline build that will create a Docker image with the provided binary files from provided Maven repositories. An IBM cloud Private deployment will be created at the end of the build process.
3. Log into the IBM Cloud Private console follow the menu to *Workloads -> Deployments* to find all deployments. Find your deployment in the list using the search filter if needed. Verify the deployment is available and open the application to try it out. Note: you may use the same steps to redeploy the application if needed later.