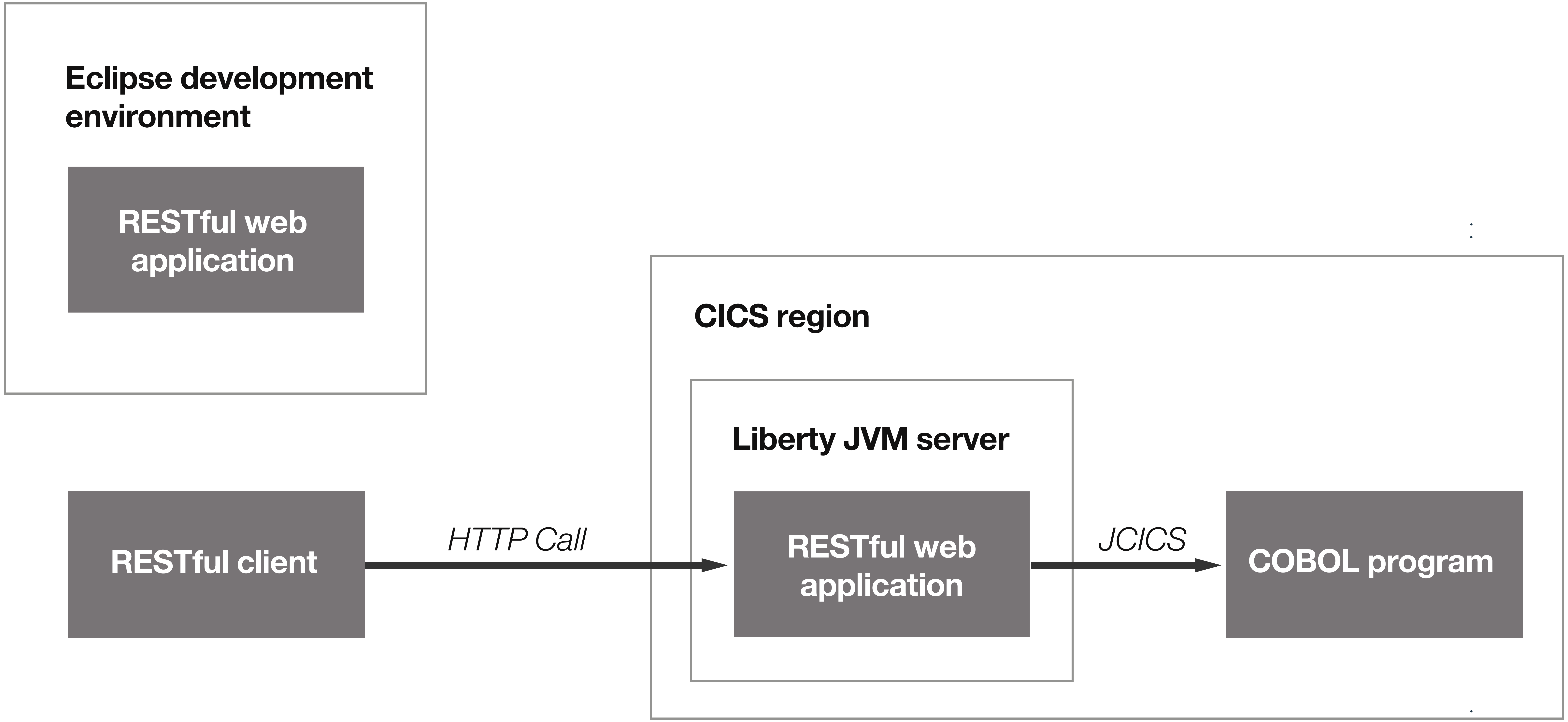
Developing a RESTful Web application for Liberty in CICS

Redbooks

Introduction

Course introduction

What you’ll see in this course



What you’ll need for this course

- Development workstation
- Ability to install IBM Explorer for z/OS
- Access to sample code
- CICS TS for z/OS V5.1 or later

What you’ll learn by the end of this course

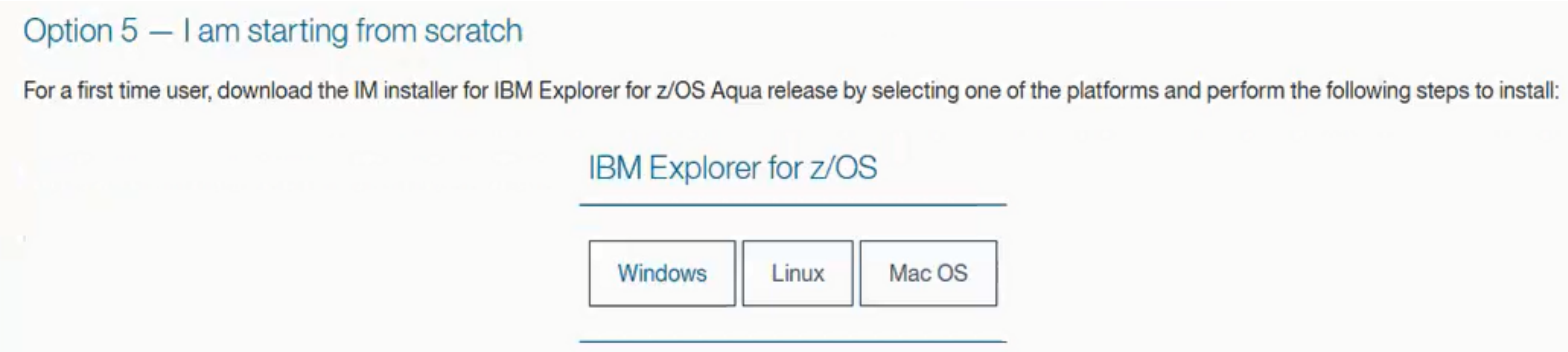
1. Developing a RESTful Java web service
2. Using the CICS Java API
3. Deployment of web applications

Creating the development environment

Installing IBM Explorer for z/OS

Downloading Eclipse

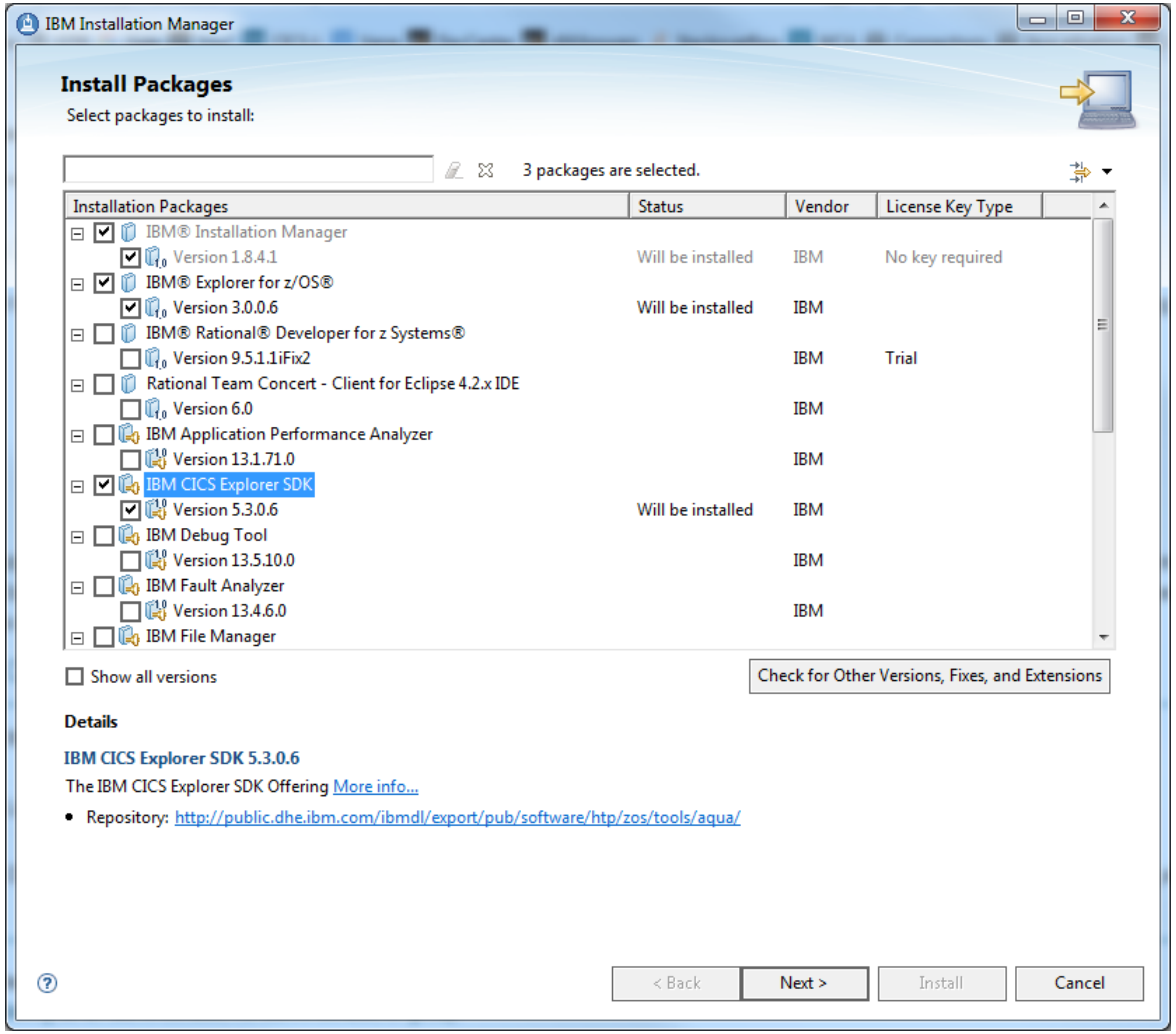
`developer.ibm.com/mainframe/products/downloads`



Option 5 — I am starting from scratch

For a first time user, download the IM installer for IBM Explorer for z/OS Aqua release by selecting one of the platforms and perform the following steps to install:

Launching IBM Installation Manager



IBM Installation Manager

Install Packages

Select packages to install:

3 packages are selected.

Installation Packages	Status	Vendor	License Key Type
⊟ ☑ IBM® Installation Manager			
☑ Version 1.8.4.1	Will be installed	IBM	No key required
⊟ ☑ IBM® Explorer for z/OS®			
☑ Version 3.0.0.6	Will be installed	IBM	
⊟ □ IBM® Rational® Developer for z Systems®			
□ Version 9.5.1.1iFix2		IBM	Trial
⊟ □ Rational Team Concert - Client for Eclipse 4.2.x IDE			
□ Version 6.0		IBM	
⊟ □ IBM Application Performance Analyzer			
□ Version 13.1.71.0		IBM	
⊟ ☑ IBM CICS Explorer SDK			
☑ Version 5.3.0.6	Will be installed	IBM	
⊟ □ IBM Debug Tool			
□ Version 13.5.10.0		IBM	
⊟ □ IBM Fault Analyzer			
□ Version 13.4.6.0		IBM	
⊟ □ IBM File Manager			

□ Show all versions

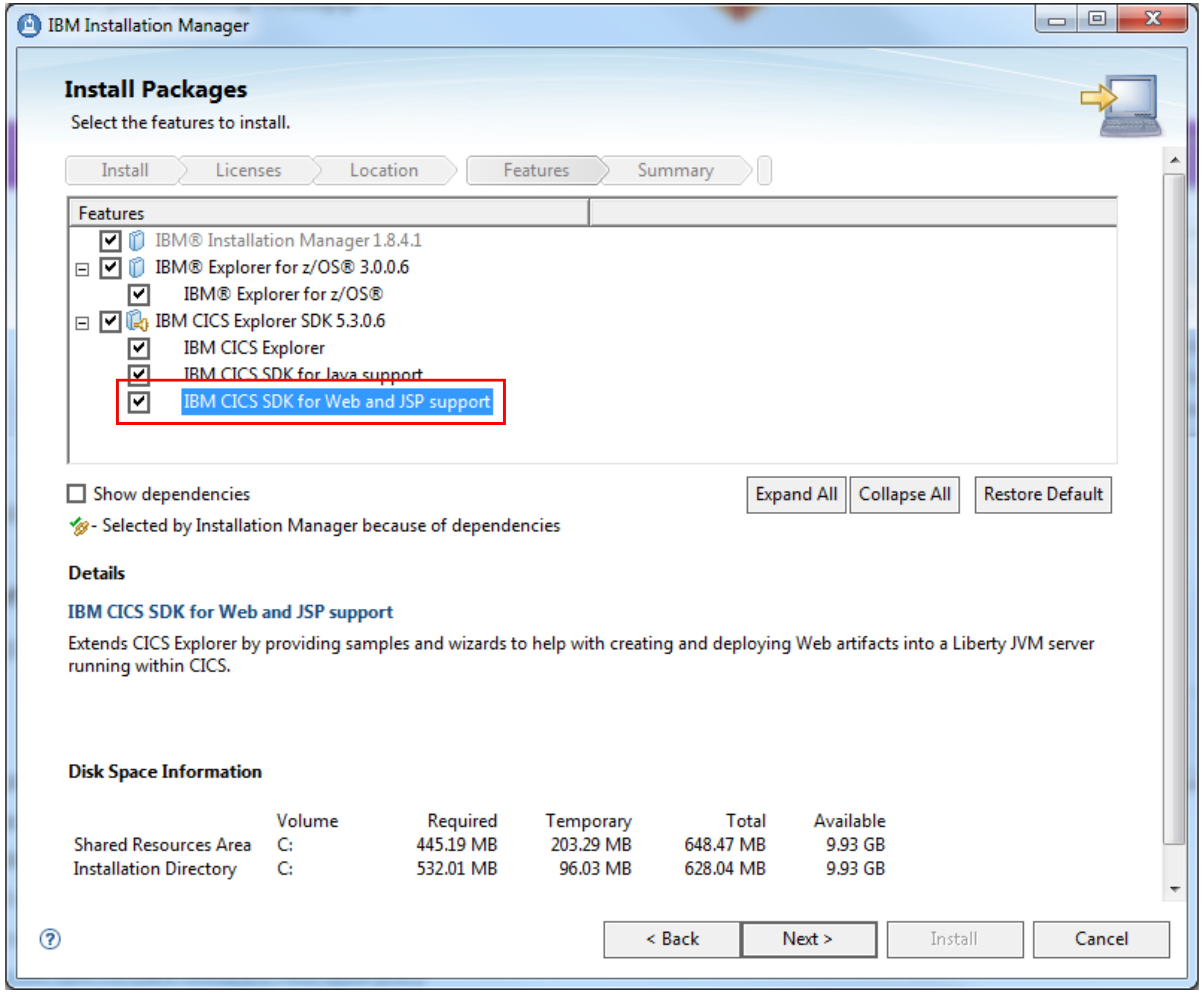
Details

IBM CICS Explorer SDK 5.3.0.6

The IBM CICS Explorer SDK Offering More info...

- Repository: http://public.dhe.ibm.com/ibmdl/export/pub/software/htp/zos/tools/aqua/

Installing packages



Install Packages

Select the features to install.

Features

- ☑ IBM® Installation Manager 1.8.4.1
- ⊟ ☑ IBM® Explorer for z/OS® 3.0.0.6
  - ☑ IBM® Explorer for z/OS®
- ⊟ ☑ IBM CICS Explorer SDK 5.3.0.6
  - ☑ IBM CICS Explorer
  - ☑ IBM CICS SDK for Java support
  - ☑ IBM CICS SDK for Web and JSP support

□ Show dependencies

- Selected by Installation Manager because of dependencies

Details

IBM CICS SDK for Web and JSP support

Extends CICS Explorer by providing samples and wizards to help with creating and deploying Web artifacts into a Liberty JVM server running within CICS.

Disk Space Information

	Volume	Required	Temporary	Total	Available
Shared Resources Area	C:	445.19 MB	203.29 MB	648.47 MB	9.93 GB
Installation Directory	C:	532.01 MB	96.03 MB	628.04 MB	9.93 GB

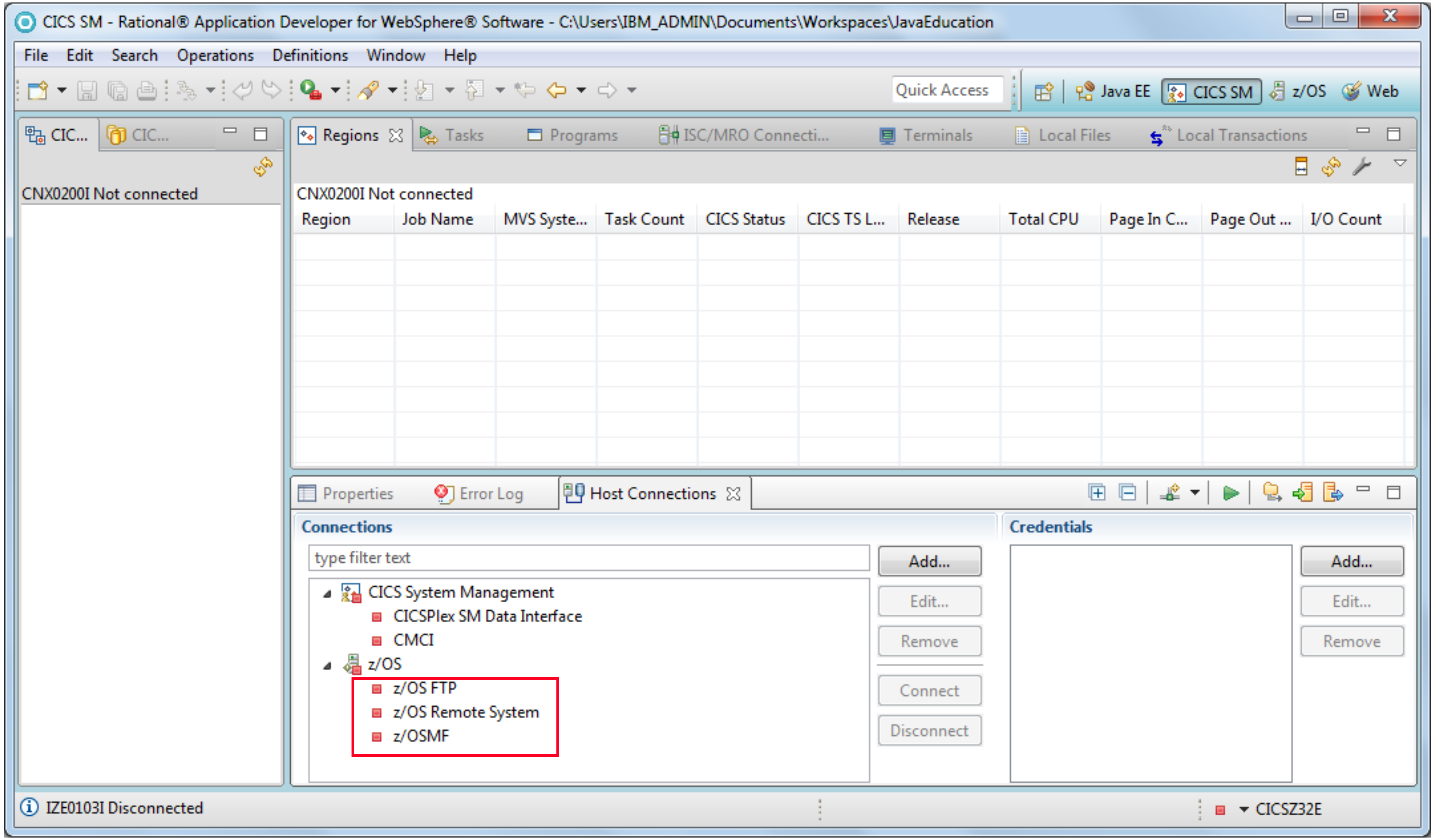
Creating the development environment

Configuring CICS Explorer

Configuring connectivity

- Configure connectivity to CICS
- Configure FTP connection

Enabling FTP connection



CNX0200I Not connected

CNX0200I Not connected

Region	Job Name	MVS Syste...	Task Count	CICS Status	CICS TS L...	Release	Total CPU	Page In C...	Page Out ...	I/O Count

- CICS System Management
  - CICSPlex SM Data Interface
  - CMCI
- z/OS
  - z/OS FTP
  - z/OS Remote System
  - z/OSMF

IZE0103I Disconnected

Connecting to a CICS region

- Create a CMCI connection to CICS

- On your CICS system
  1. Add load libraries SEYUAUTH and SEYULOAD to the CICS JCL
  2. Create a URIMAP
  3. Create a TCPIPSERVICE
  4. Or setup CICSPlex SM WUI server

- See Knowledge Center topic **Setting up access for CICS Explorer**

Summary

- Installed CICS Explorer
- Created FTP and CMCI connections
- Next section - Using CICS Explorer to manage a Liberty JVM

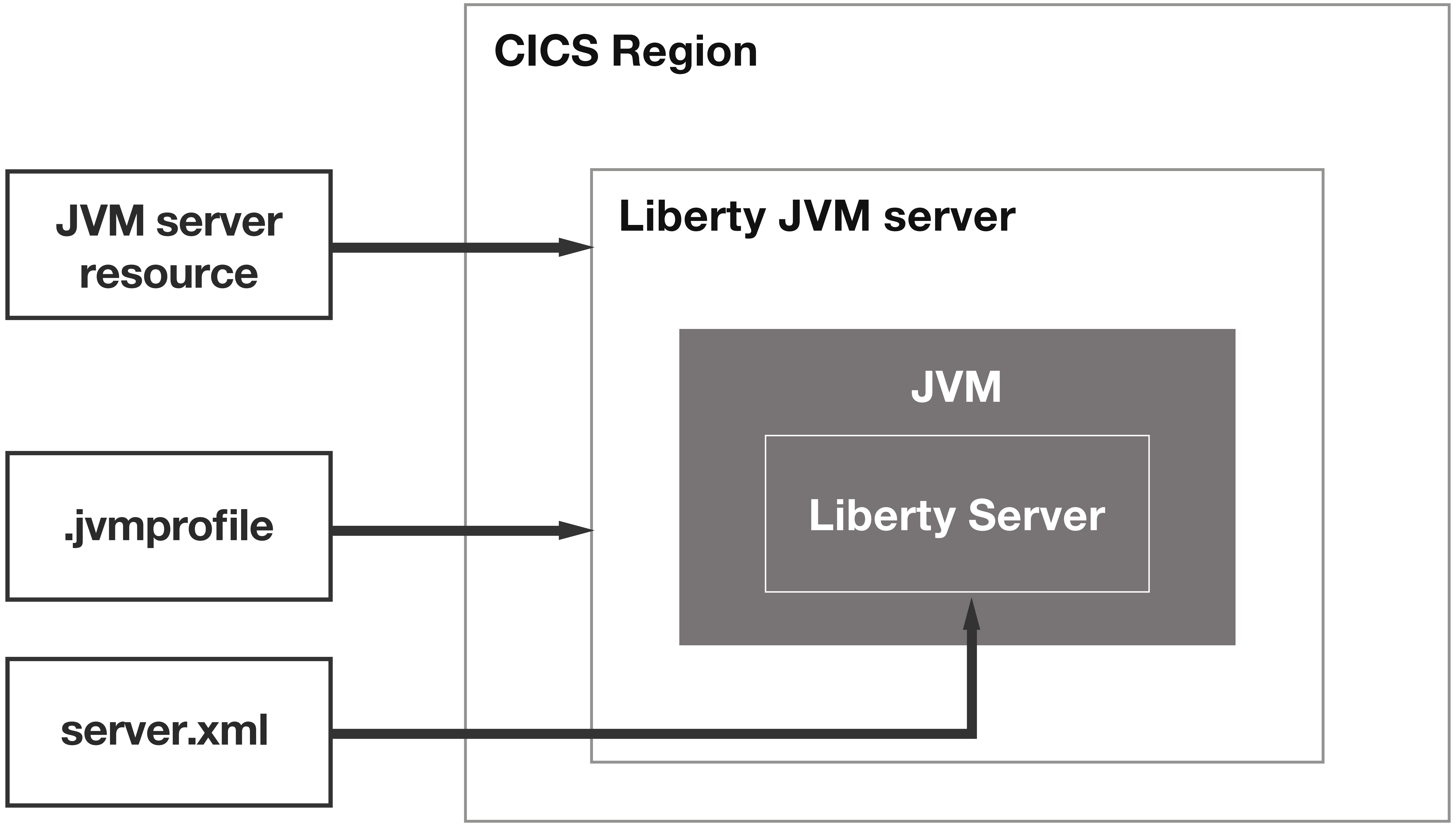
Configuring a CICS Liberty JVM server

Introduction to the Liberty JVM server

Introduction to the Liberty JVM server

- WebSphere Liberty
- Java EE application server
- Tightly integrated with CICS
- Liberty JVM server
- Basic steps of how to configure

Liberty and CICS: A 10,000 feet view



Configuring a CICS Liberty JVM server

Configuring a Liberty JVM server

Configuring Java in CICS

- CICS region started
- SDFJAUTH in STEPLIB
- LE support added

System initialization parameters

```
USSHOME=/usr/lpp/cicsts/cics700
```

JVM profile directory

```
JVMPROFILEDIR=/u/cics1
```

JVM sample profile

```
/usr
   /lpp
      /cicsts
         /cics700
            /docs
            /IBM
            /JVMProfiles
               /DFHJVMAX.jvmprofile
               /DFHJVMST.jvmprofile
               /DFHOSGI.jvmprofile
               /DFHWLP.jvmprofile
```

USSHOME

Sample Liberty JVM profile

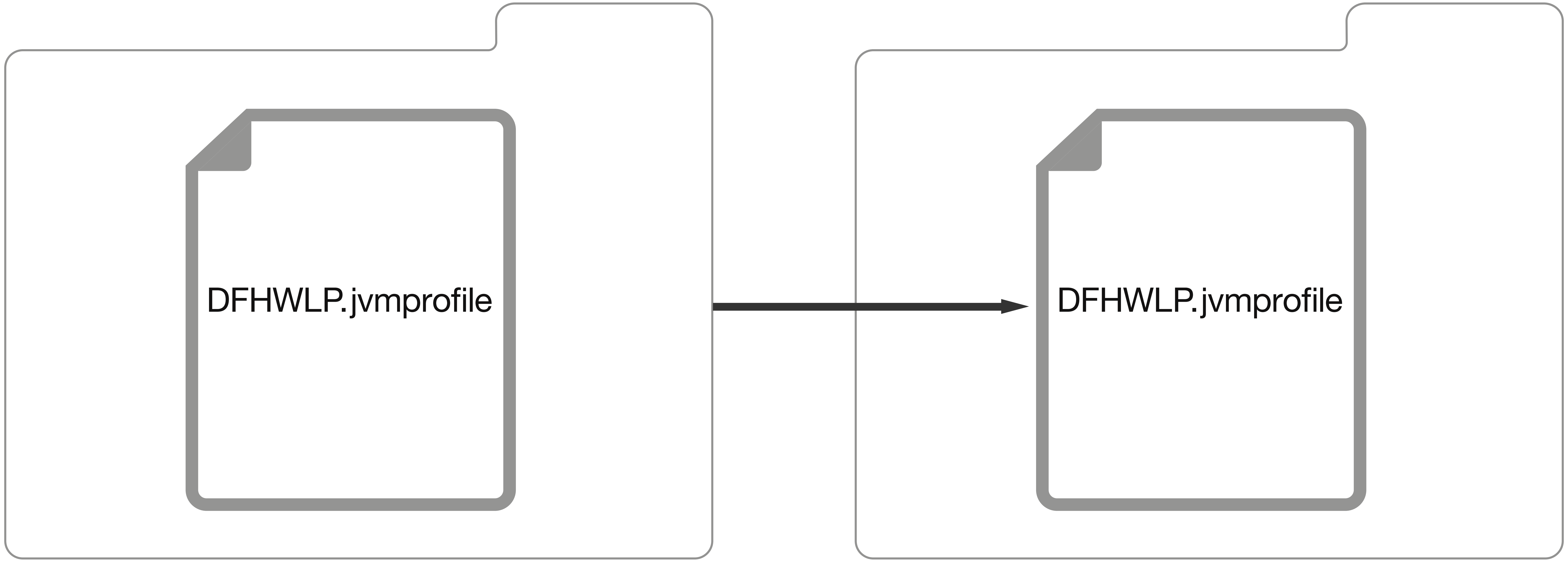
Copying the sample profile

`${USSHOME}/JVMProfiles`

`/usr/lpp/cicsts/cics700/JVMProfiles`

`${JVMPROFILEDIR}`

`/u/cics1`



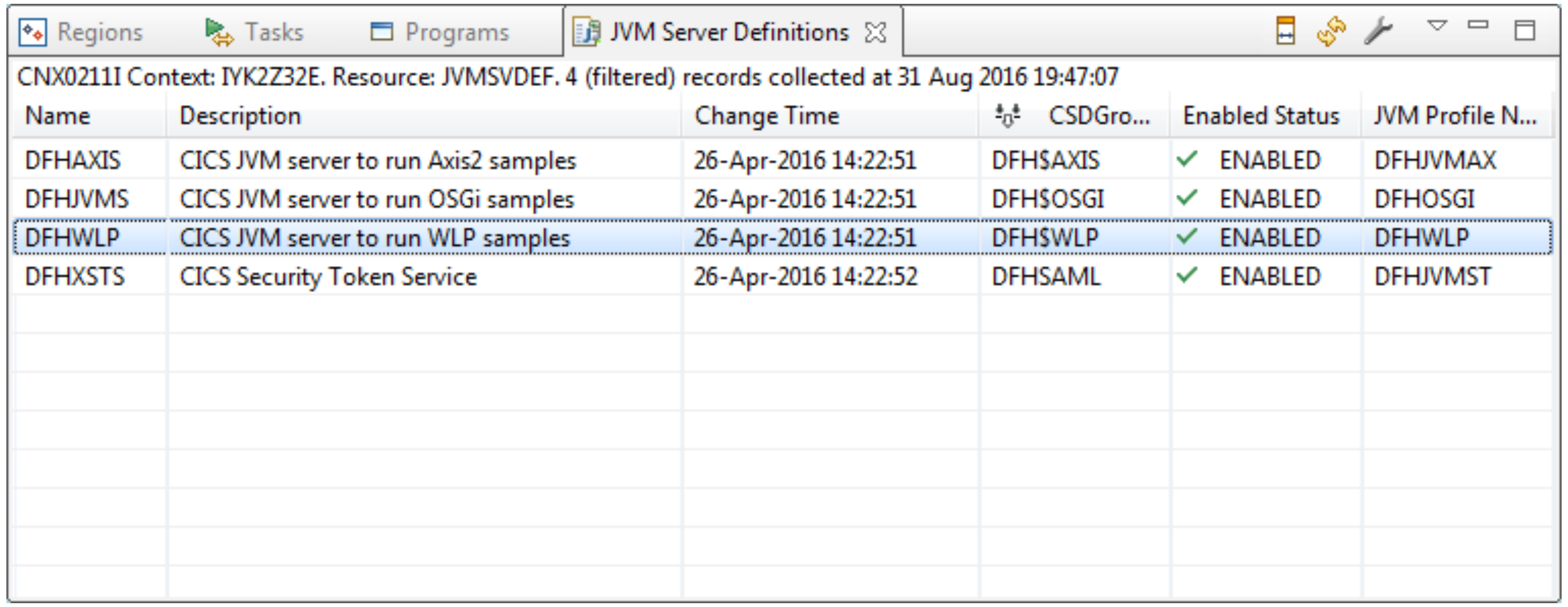
Reviewing and updating the JVM profile

Reviewing and updating the JVM profile

Enabling auto configure

Setting the TCP/IP host values

Selecting the JVM server resource definition



CNX0211I Context: IYK2Z32E. Resource: JVMSVDEF. 4 (filtered) records collected at 31 Aug 2016 19:47:07

Name	Description	Change Time	CSDGro...	Enabled Status	JVM Profile N...
DFHAXIS	CICS JVM server to run Axis2 samples	26-Apr-2016 14:22:51	DFH$AXIS	✓ ENABLED	DFHJVMAX
DFHJVMS	CICS JVM server to run OSGi samples	26-Apr-2016 14:22:51	DFH$OSGI	✓ ENABLED	DFHOSGI
DFHWLP	CICS JVM server to run WLP samples	26-Apr-2016 14:22:51	DFH$WLP	✓ ENABLED	DFHWLP
DFHXSTS	CICS Security Token Service	26-Apr-2016 14:22:52	DFHSAML	✓ ENABLED	DFHJVMST

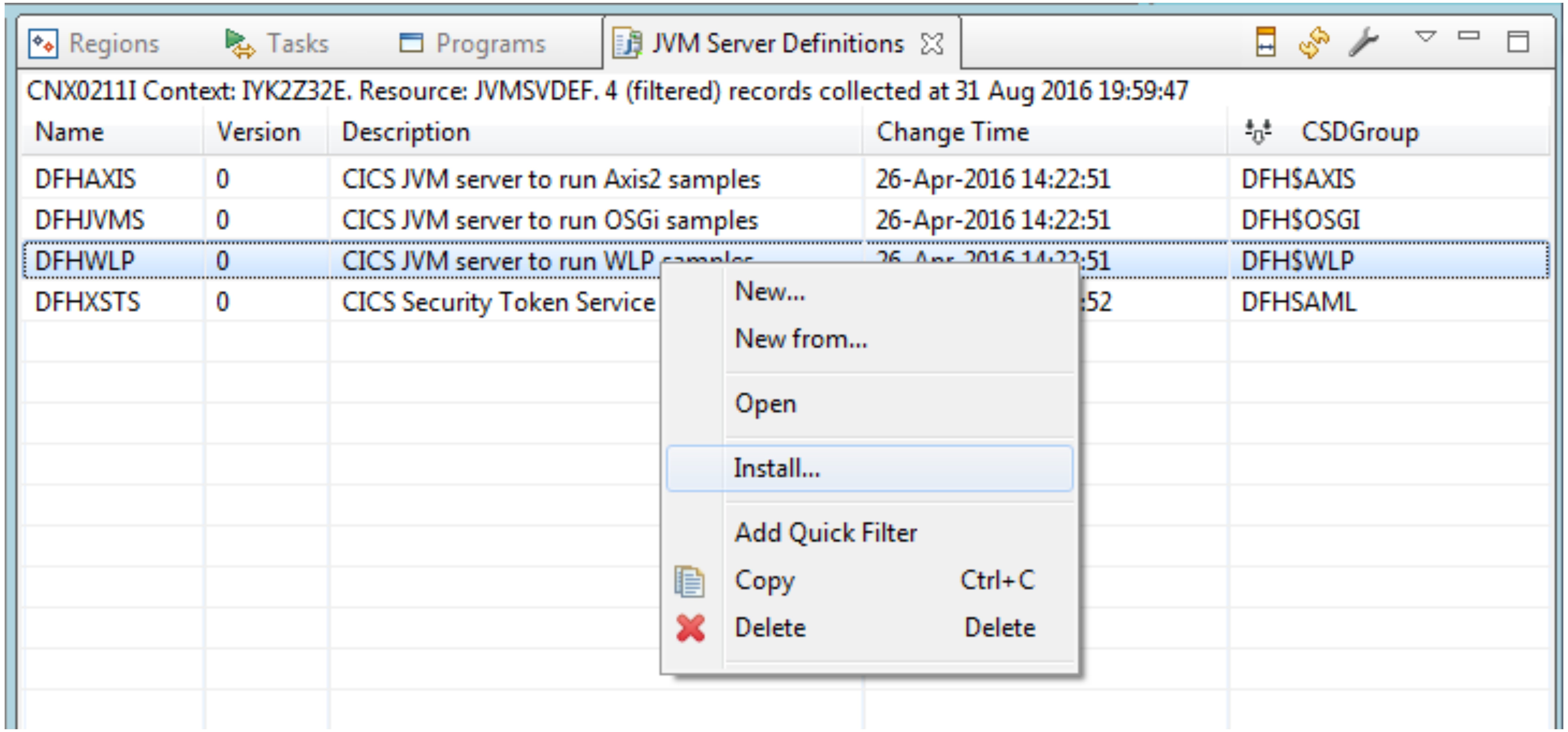
Summary

- Created JVM profile using supplied template DFHWLP.jvmprofile
- Created CICS JVMSERVER resource definition
- Next lecture - Install and enable the JVM server

Configuring a CICS Liberty JVM server

Starting and validating the Liberty JVM server

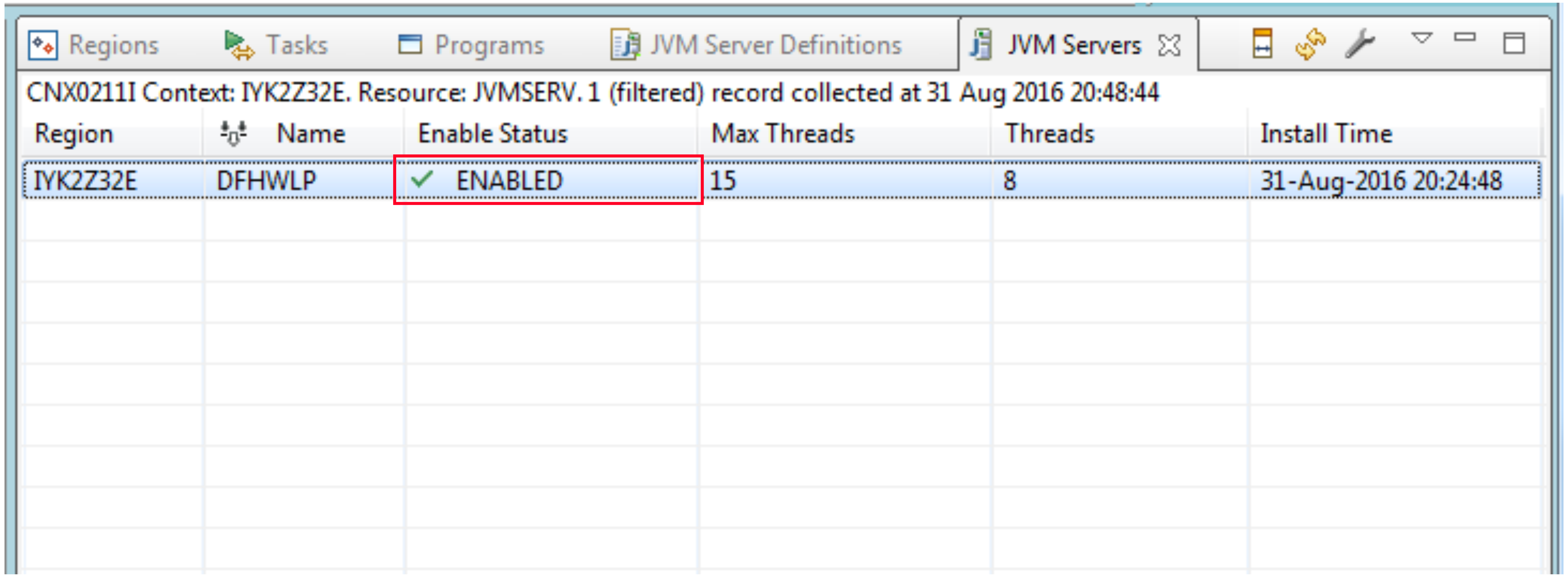
Starting the Liberty JVM server



CNX0211I Context: IYK2Z32E. Resource: JVMSVDEF. 4 (filtered) records collected at 31 Aug 2016 19:59:47

Name	Version	Description	Change Time	CSDGroup
DFHAXIS	0	CICS JVM server to run Axis2 samples	26-Apr-2016 14:22:51	DFH$AXIS
DFHJVMS	0	CICS JVM server to run OSGi samples	26-Apr-2016 14:22:51	DFH$OSGI
DFHWLP	0	CICS JVM server to run WLP samples	26-Apr-2016 14:22:51	DFH$WLP
DFHXSTS	0	CICS Security Token Service	:52	DFHSAML

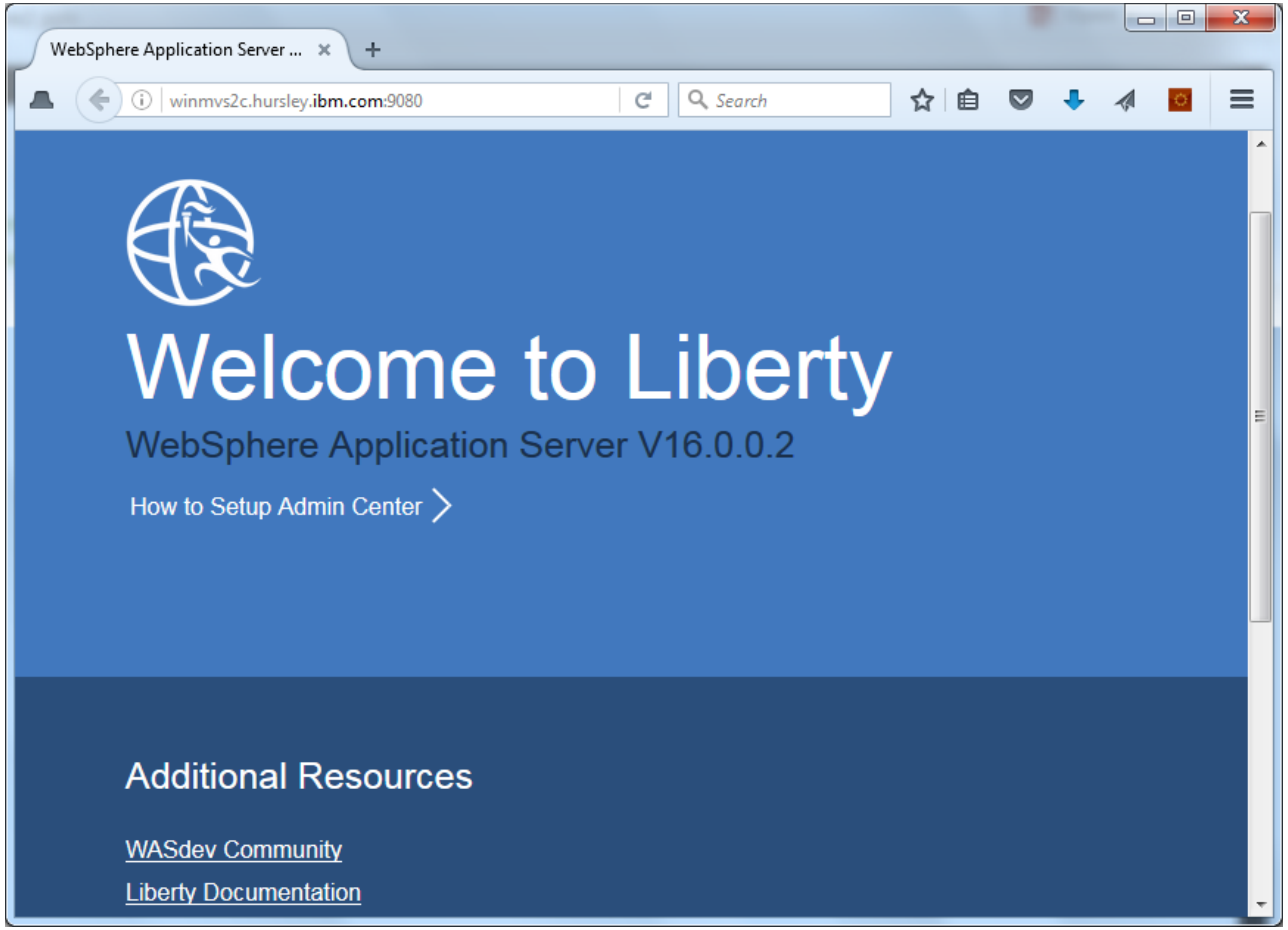
Verifying the Liberty JVM server is running



CNX0211I Context: IYK2Z32E. Resource: JVMSERV. 1 (filtered) record collected at 31 Aug 2016 20:48:44

Region	Name	Enable Status	Max Threads	Threads	Install Time
IYK2Z32E	DFHWLP	✓ ENABLED	15	8	31-Aug-2016 20:24:48

Accessing the Liberty server default welcome page



winmvs2c.hursley.ibm.com:9080

Welcome to Liberty

WebSphere Application Server V16.0.0.2

How to Setup Admin Center

Additional Resources

WASdev Community

Liberty Documentation

Viewing the Liberty message log

Verifying the logs

Verifying the logs

Editing the server.xml file

Adding the JAX-RS feature

Enabling dropins

Editing the server.xml file

Verifying the changes

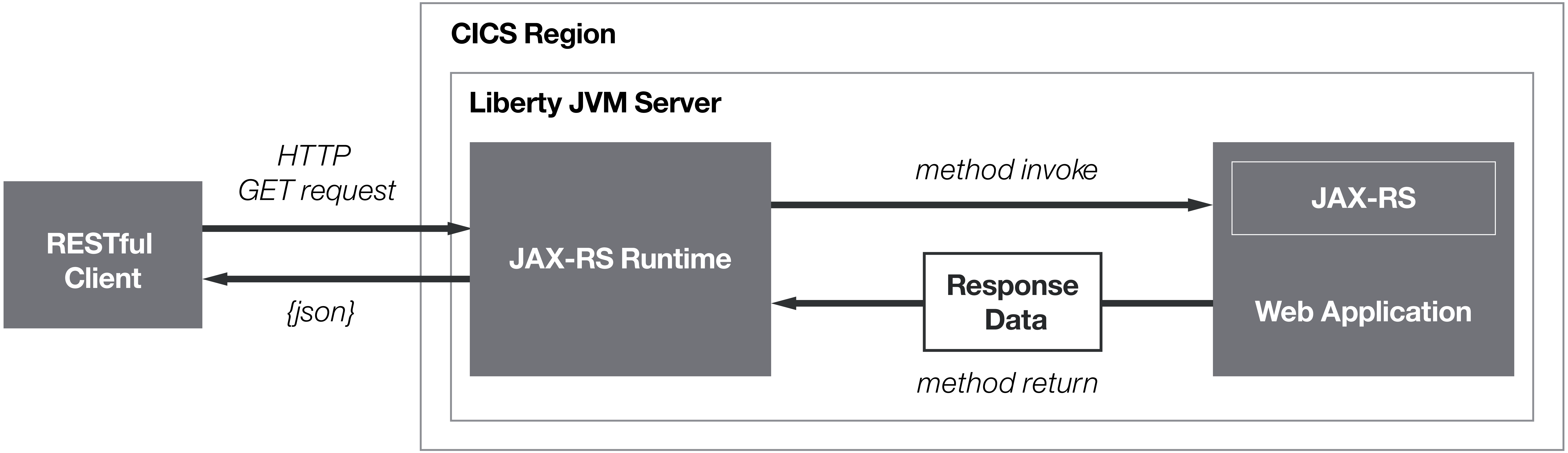
Creating a RESTful service for Liberty in CICS

Introduction to our RESTful web application

RESTful web applications

- REST
- JAX-RS
- Java EE 6
- JSON

Sample application architecture



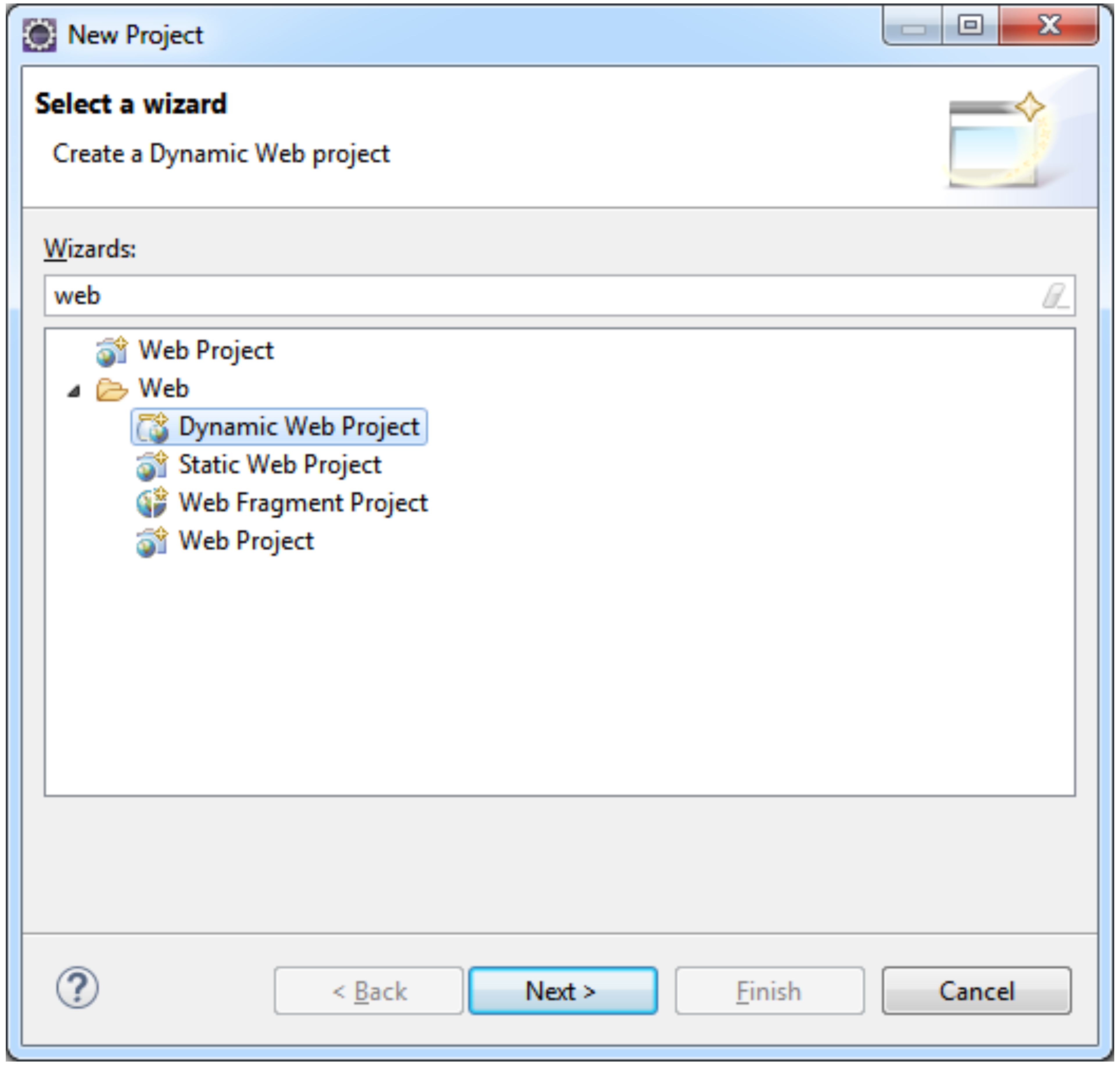
Creating a RESTful service for Liberty in CICS

Developing the application

Sample application

`github.com/cicsdev/cics-java-liberty-restapp`

New project dialog



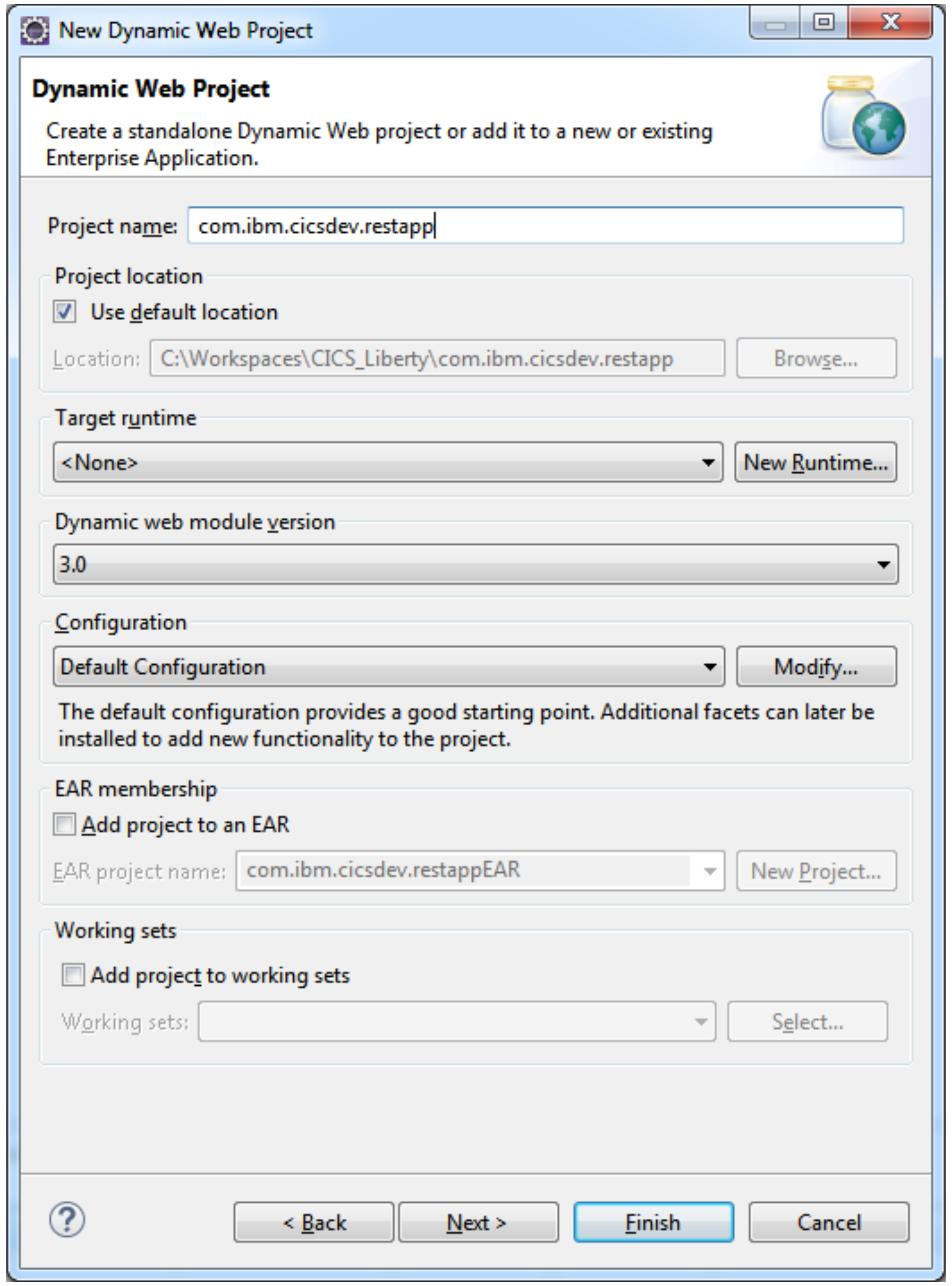
New Project

Select a wizard

Create a Dynamic Web project

Wizards: web

New dynamic web project wizard



Dynamic Web Project

Create a standalone Dynamic Web project or add it to a new or existing Enterprise Application.

Project name: com.ibm.cicsdev.restapp

Project location

☑ Use default location

Location: C:\Workspaces\CICS_Liberty\com.ibm.cicsdev.restapp

Target runtime: <None>

Dynamic web module version: 3.0

Configuration: Default Configuration

The default configuration provides a good starting point. Additional facets can later be installed to add new functionality to the project.

EAR membership

☐ Add project to an EAR

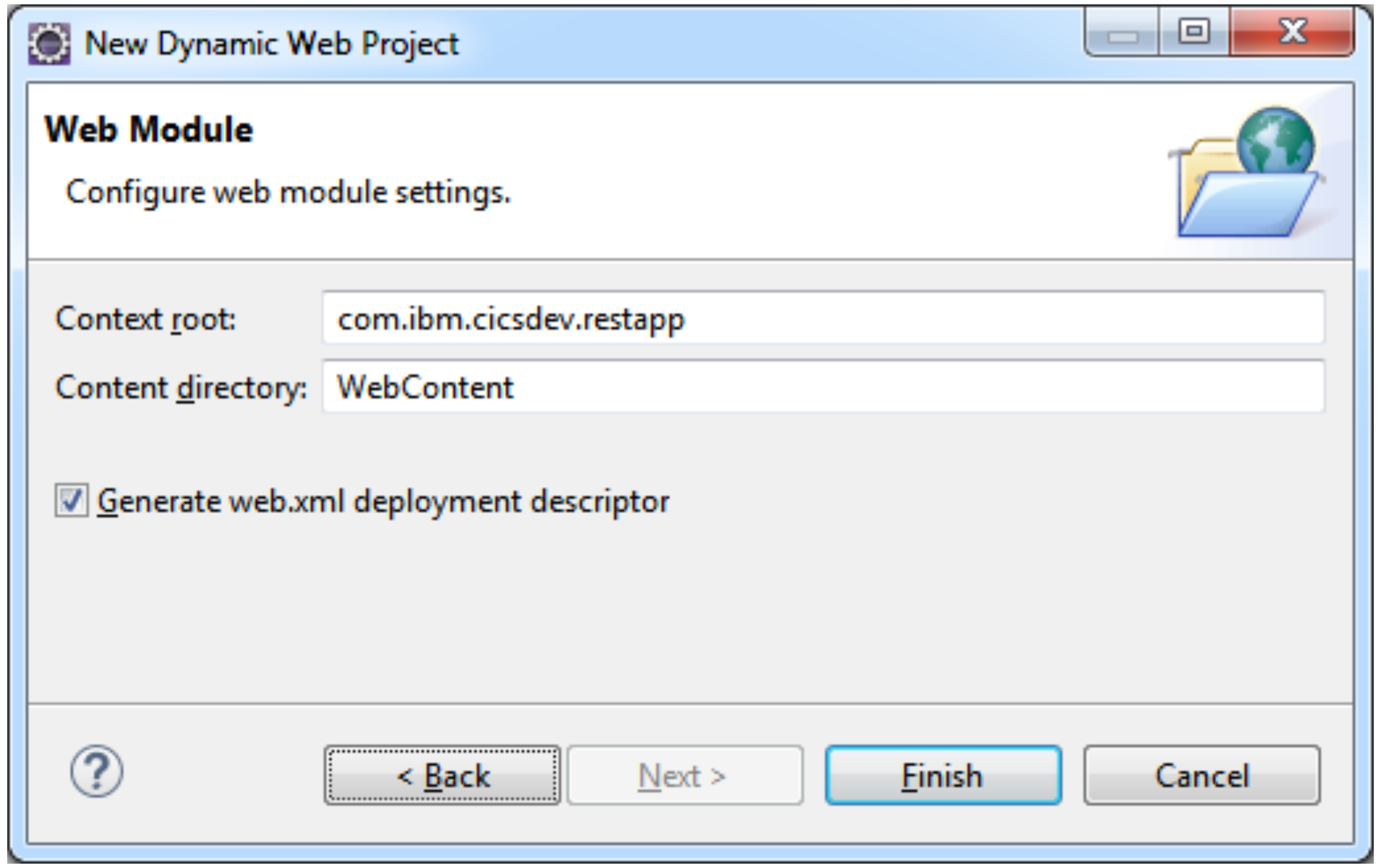
EAR project name: com.ibm.cicsdev.restappEAR

Working sets

☐ Add project to working sets

Working sets:

New dynamic web project wizard



New Dynamic Web Project

Web Module

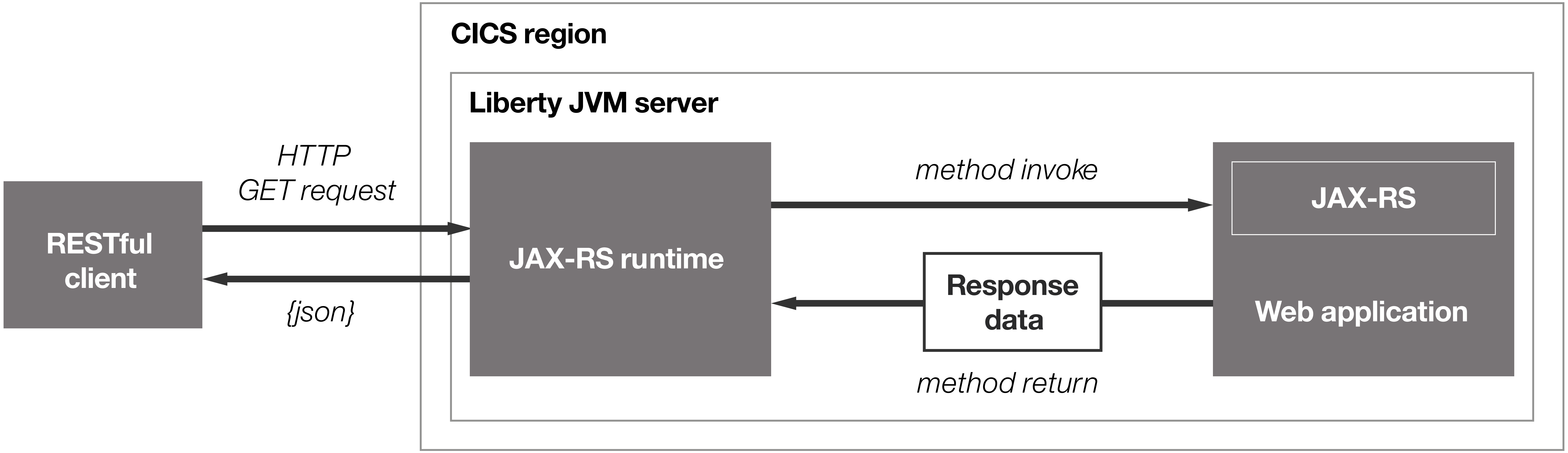
Configure web module settings.

Context root: com.ibm.cicsdev.restapp

Content directory: WebContent

☑ Generate web.xml deployment descriptor

Sample application architecture



Java bean classes

CICSInformation

APPLID

JVM server

Current time

CICS environment

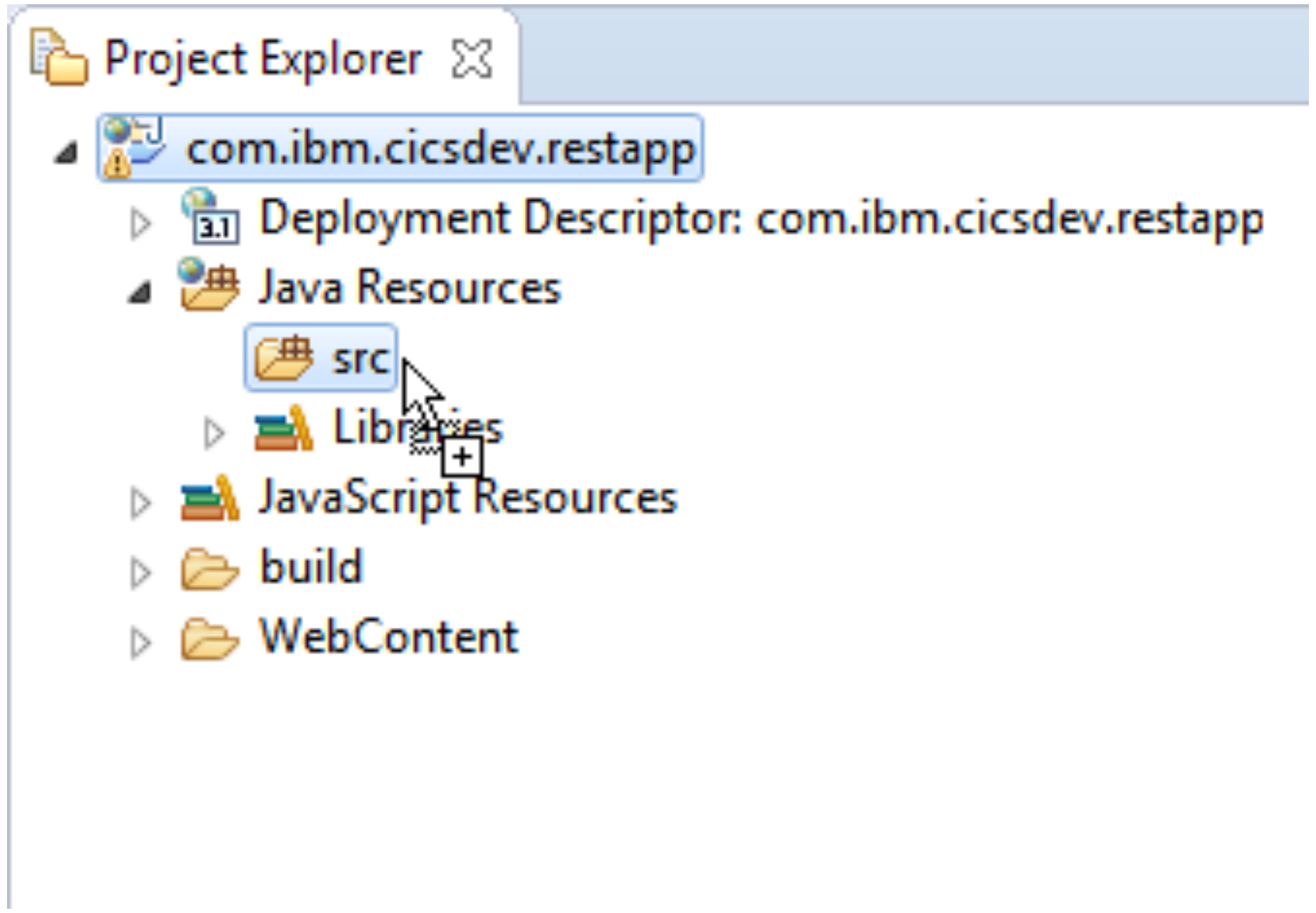
CICSEnvironment

Product name

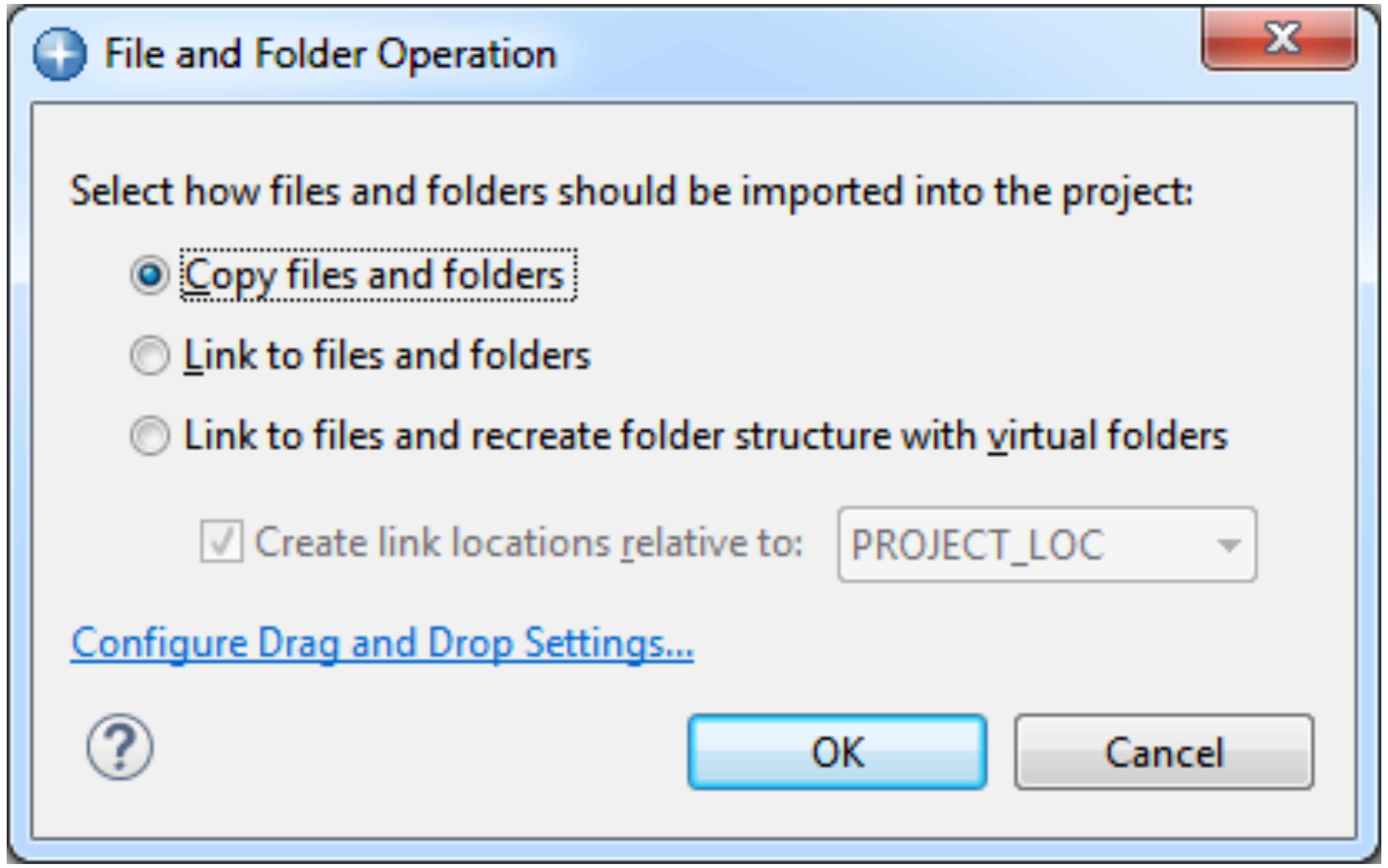
Product version

getCICSInformation method

Importing the code



Importing the code



File and Folder Operation

Select how files and folders should be imported into the project:

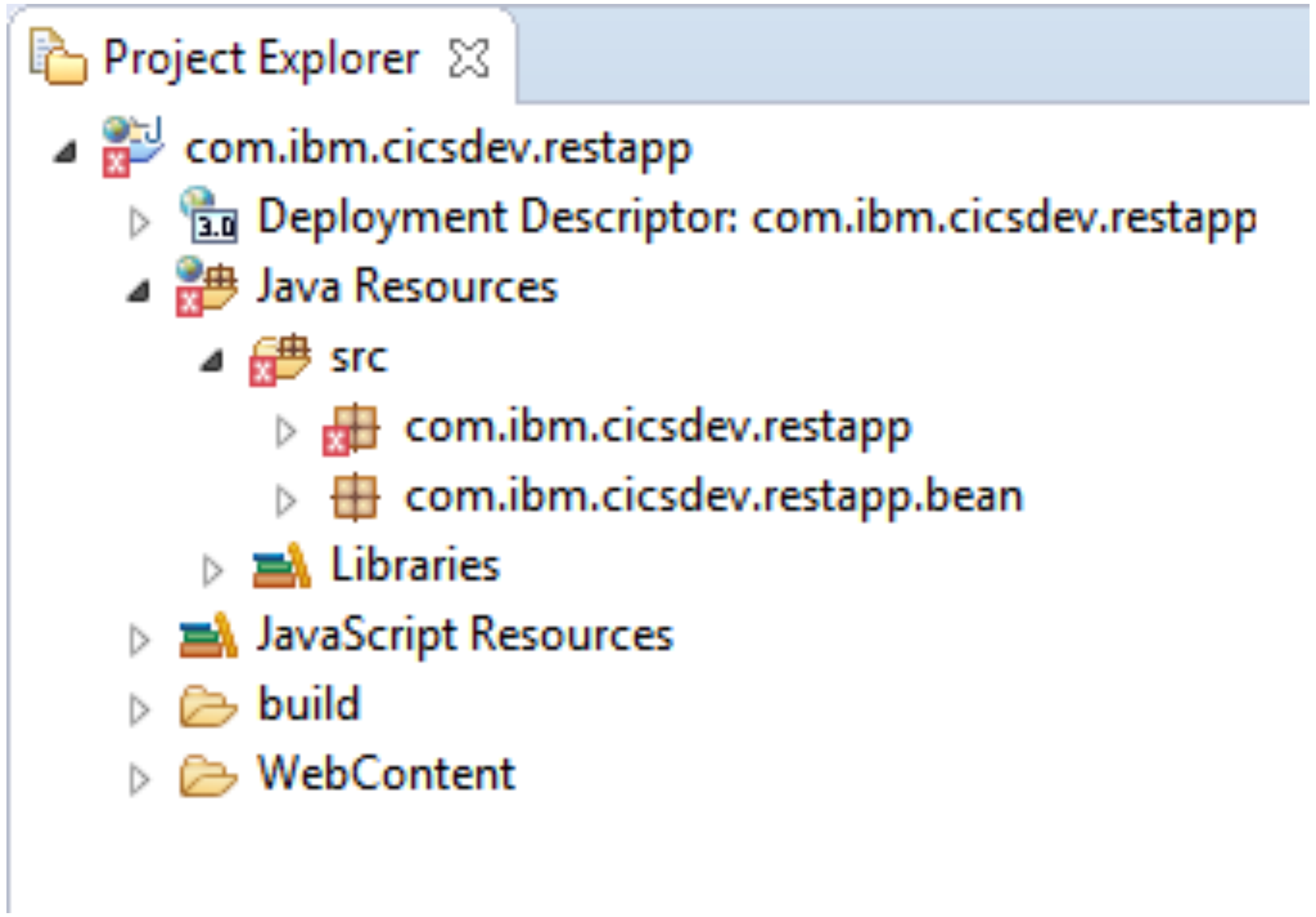
● Copy files and folders

○ Link to files and folders

○ Link to files and recreate folder structure with virtual folders

☑ Create link locations relative to: PROJECT_LOC

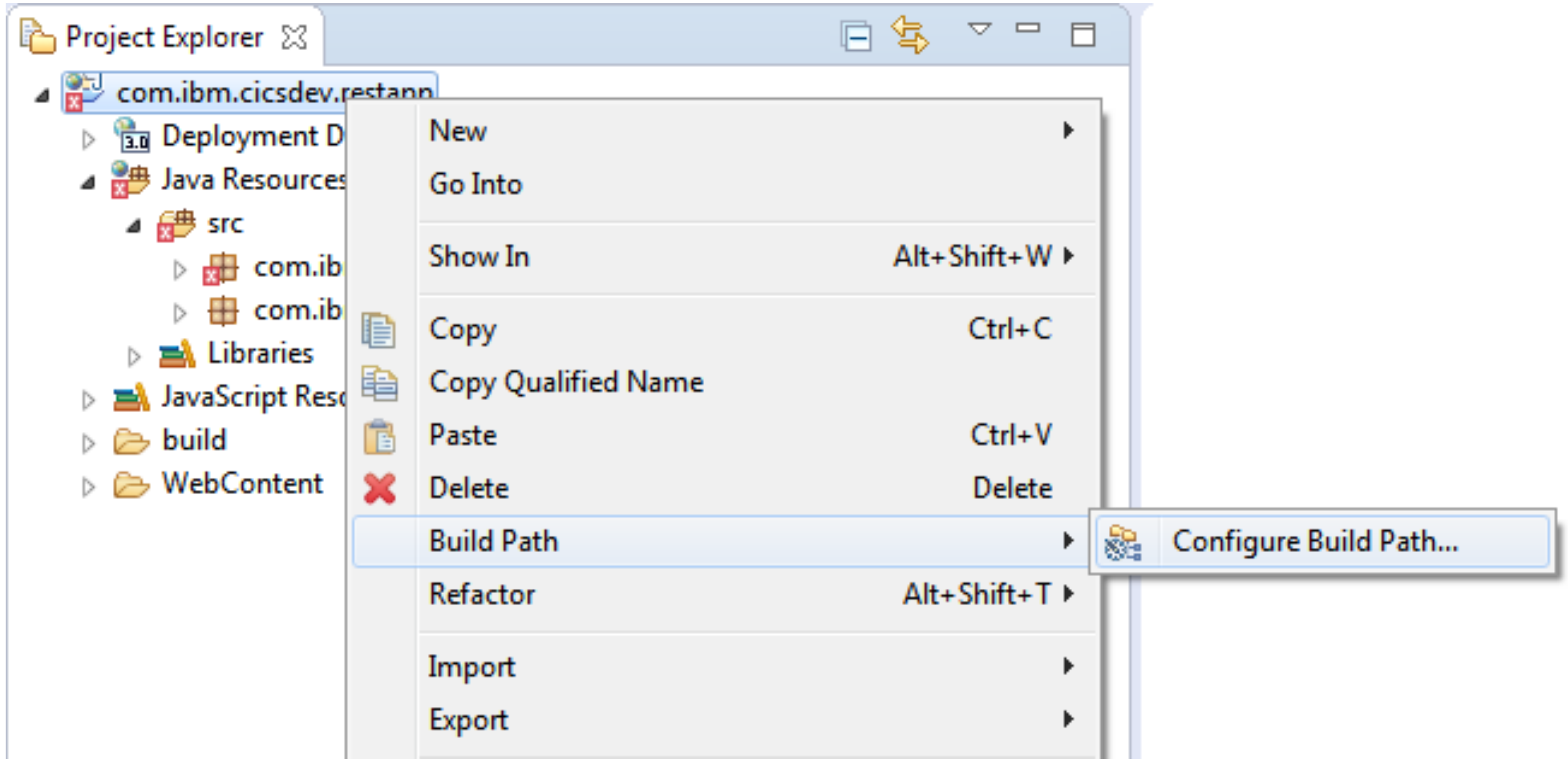
Verifying packages



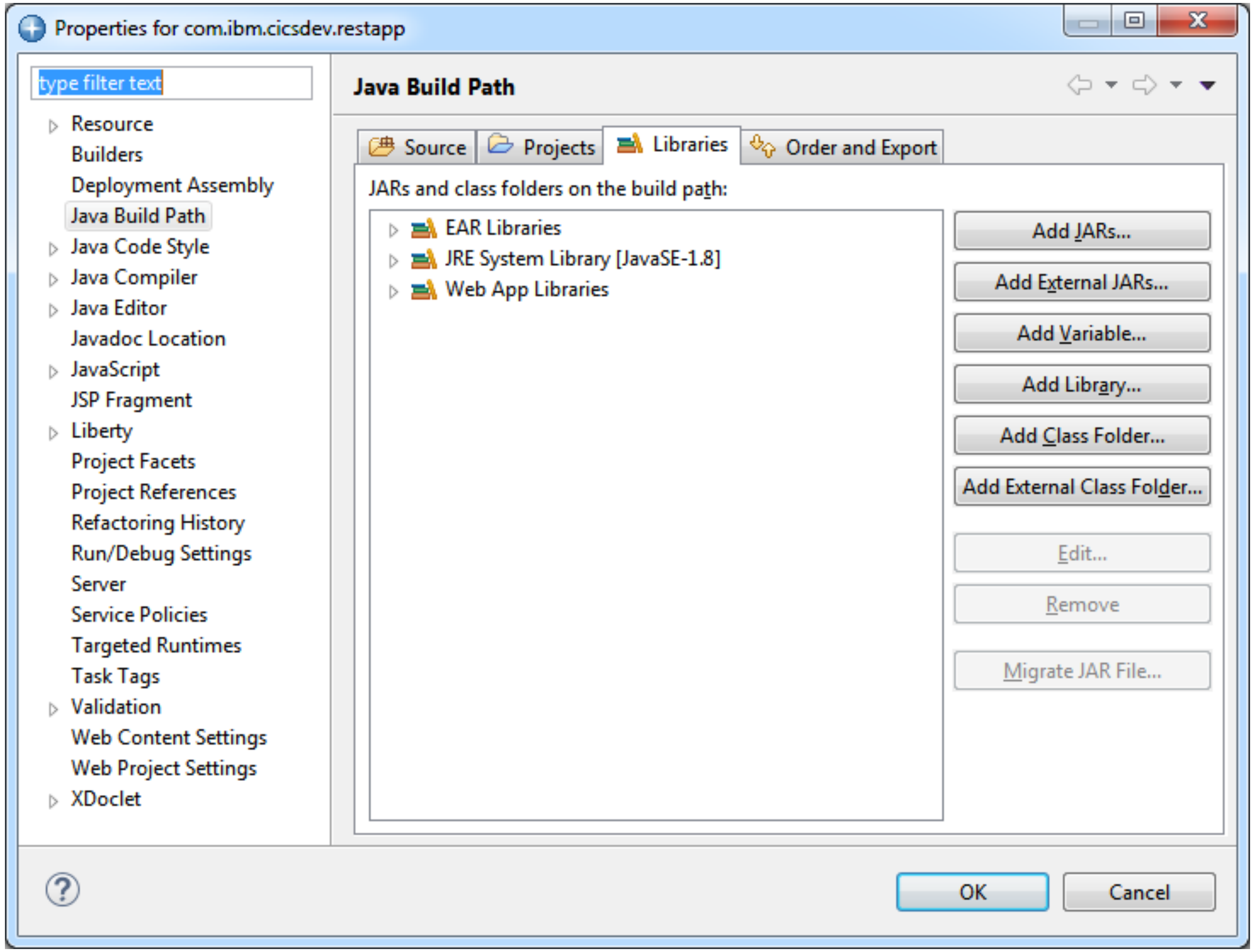
Project Explorer

- com.ibm.cicsdev.restapp
  - Deployment Descriptor: com.ibm.cicsdev.restapp
  - Java Resources
    - src
      - com.ibm.cicsdev.restapp
      - com.ibm.cicsdev.restapp.bean
    - Libraries
  - JavaScript Resources
  - build
  - WebContent

Configuring the build path



Adding libraries to the build path



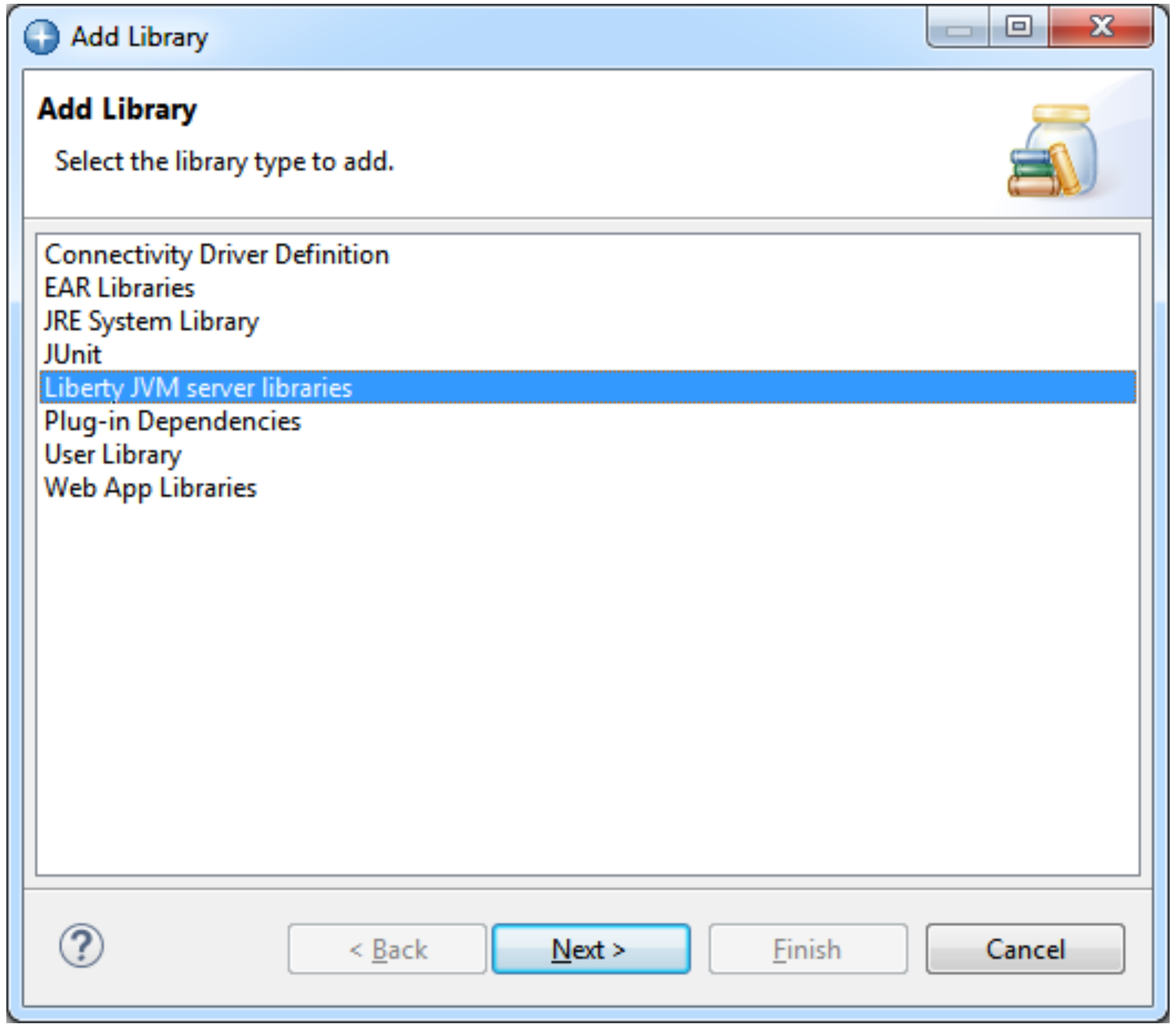
Properties for com.ibm.cicsdev.restapp

Java Build Path

JARs and class folders on the build path:

- EAR Libraries
- JRE System Library [JavaSE-1.8]
- Web App Libraries

Selecting libraries to add

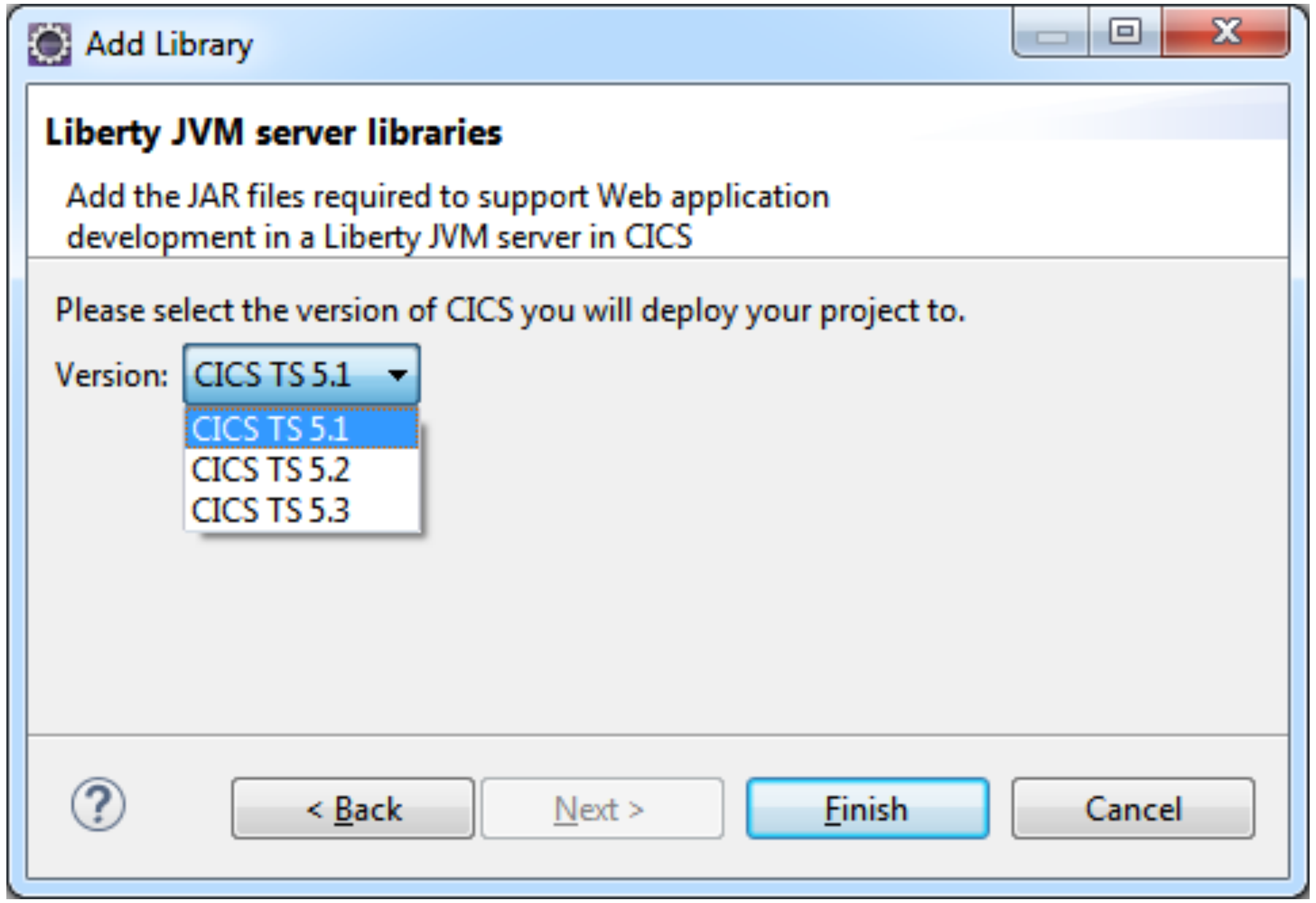


Add Library

Select the library type to add.

- Connectivity Driver Definition
- EAR Libraries
- JRE System Library
- JUnit
- Liberty JVM server libraries
- Plug-in Dependencies
- User Library
- Web App Libraries

Selecting the target CICS version for deployment



Liberty JVM server libraries

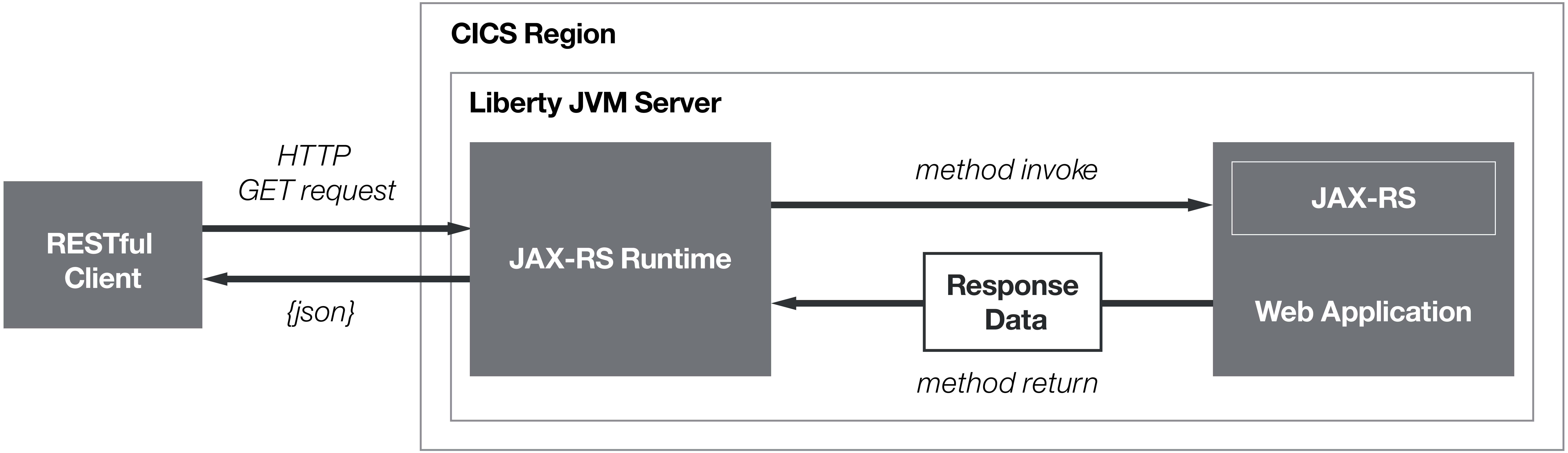
Add the JAR files required to support Web application development in a Liberty JVM server in CICS

Please select the version of CICS you will deploy your project to.

Version: CICS TS 5.1

JAXB annotations

Sample application architecture



JAX-RS annotations

JAX-RS annotations

getCICSInformation method

CICSInformation class

Summary

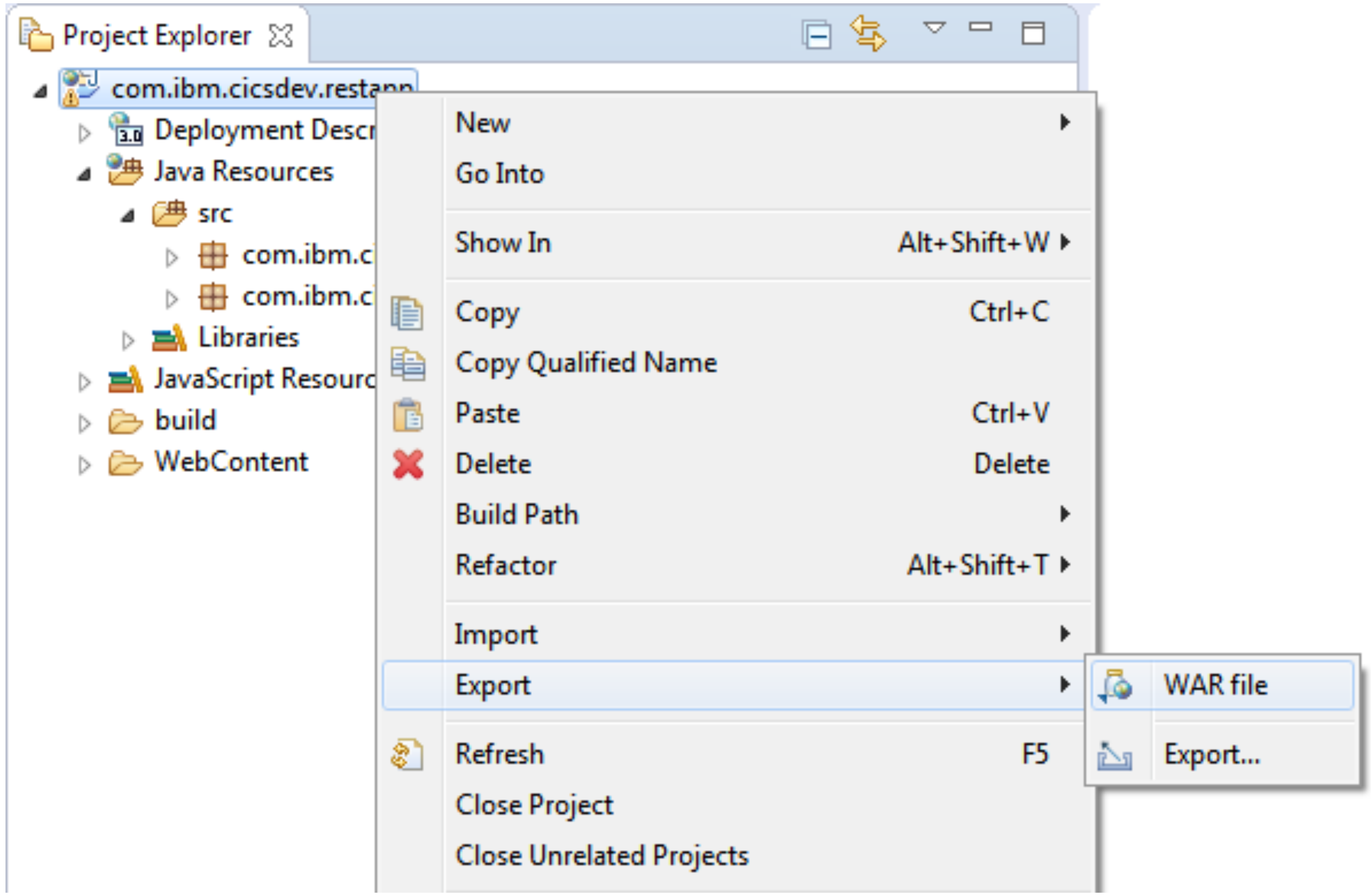
- Completed coding of our application
- Ready to deploy into our Liberty JVM server
- We will extend our application in the next lecture

Creating a RESTful service for Liberty in CICS

Deploying the application to the dropins directory

Deploying via the dropins directory

Exporting a WAR file



Copying WAR file to z/OS

Messages.log file

Messages.log file

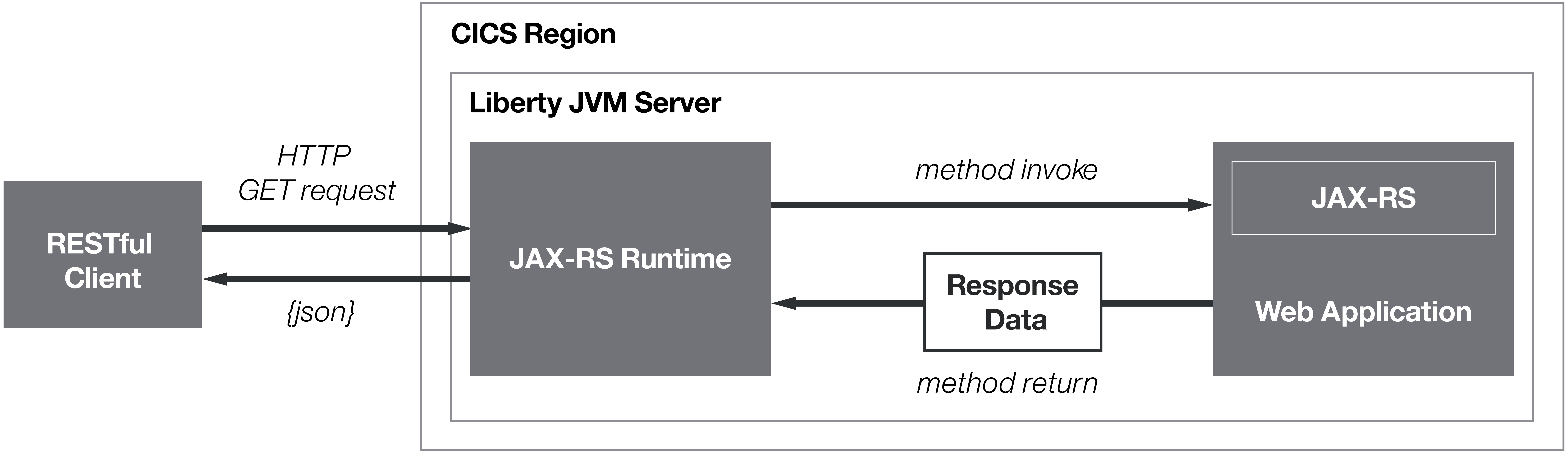
Testing the application

```
http://winmvs2c.hursley.ibm.com:9080/com.ibm.cicsdev.restapp/
```

The URL

```
http://winmvs2c.hursley.ibm.com:9080/com.ibm.cicsdev.restapp/rest/cicsinfo
```

Sample application architecture



JSON response

Summary

- Coded
- Deployed
- Tested

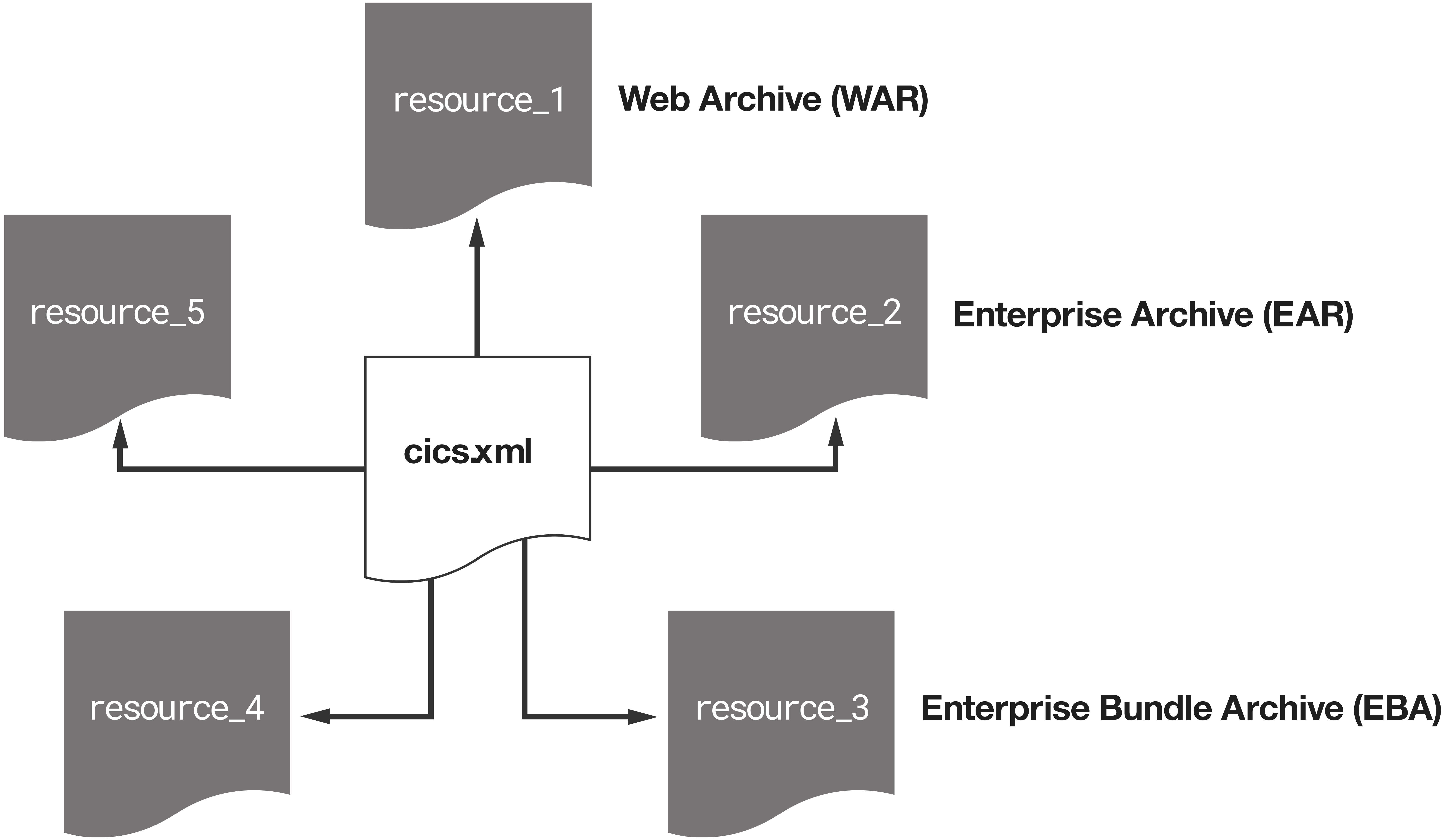
Creating a RESTful service for Liberty in CICS

Deploying the application in a CICS bundle

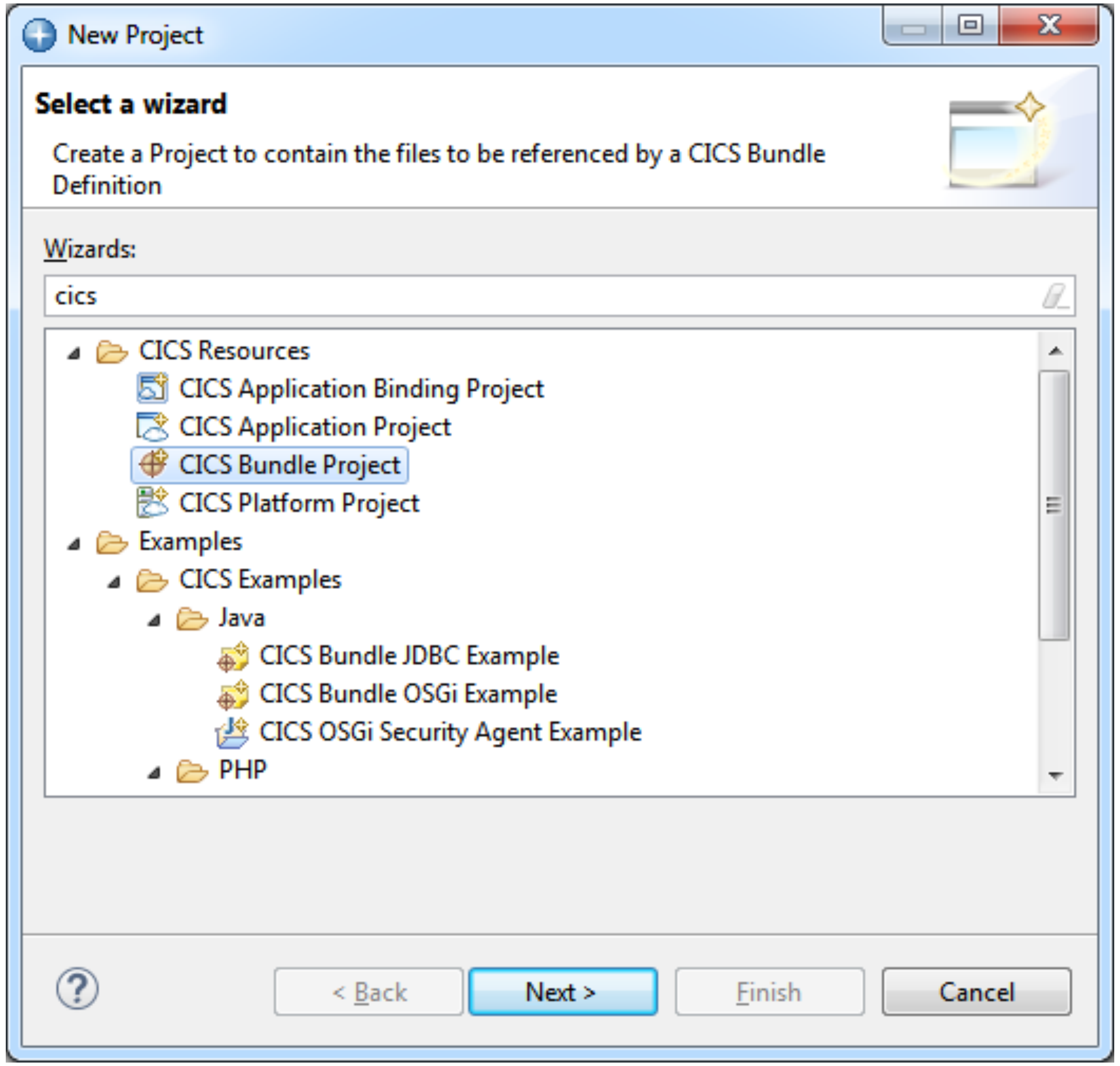
Deploying the application in a CICS bundle

- RESTful web application with dropins
- Useful for rapid application development
- Deploy the WAR file using a CICS bundle
- First delete any existing restapp WAR files from dropins

What is a CICS bundle?



CICS bundle project



New Project

Select a wizard

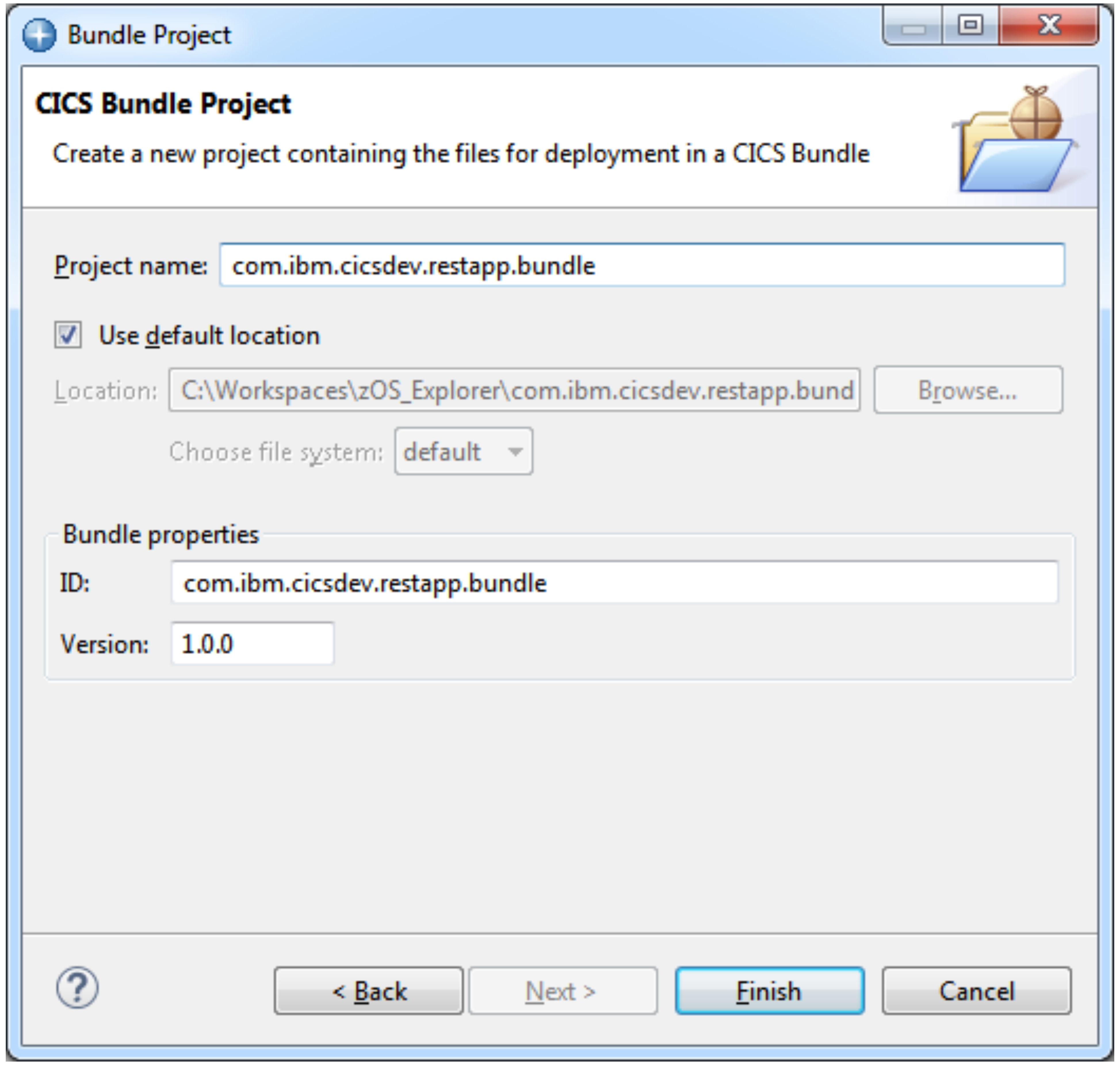
Create a Project to contain the files to be referenced by a CICS Bundle Definition

Wizards:

cics

- CICS Resources
  - CICS Application Binding Project
  - CICS Application Project
  - CICS Bundle Project
  - CICS Platform Project
- Examples
  - CICS Examples
    - Java
      - CICS Bundle JDBC Example
      - CICS Bundle OSGi Example
      - CICS OSGi Security Agent Example
    - PHP

CICS bundle project



Bundle Project

CICS Bundle Project

Create a new project containing the files for deployment in a CICS Bundle

Project name: com.ibm.cicsdev.restapp.bundle

☑ Use default location

Location: C:\Workspaces\zOS_Explorer\com.ibm.cicsdev.restapp.bund

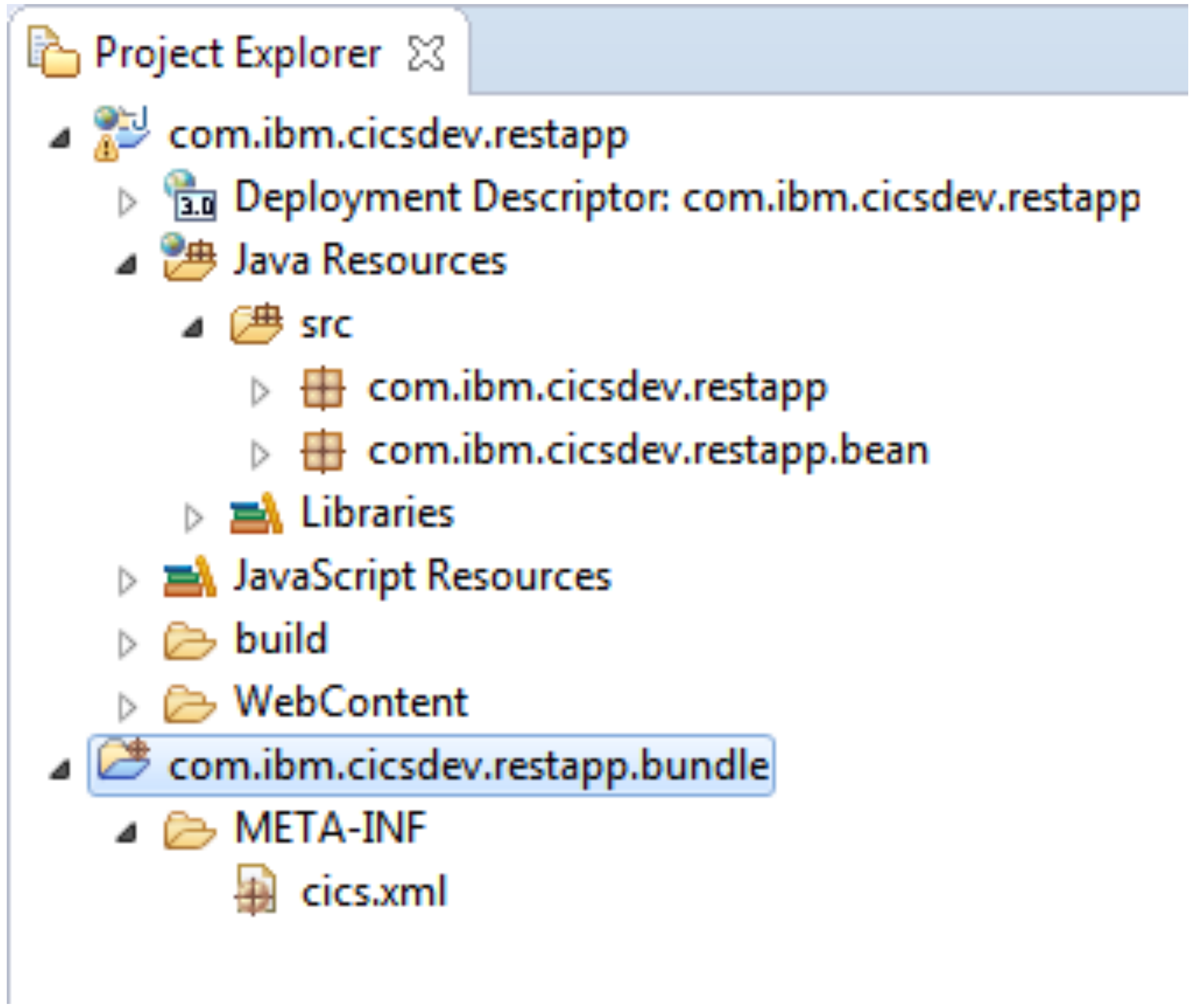
Choose file system: default

Bundle properties

ID: com.ibm.cicsdev.restapp.bundle

Version: 1.0.0

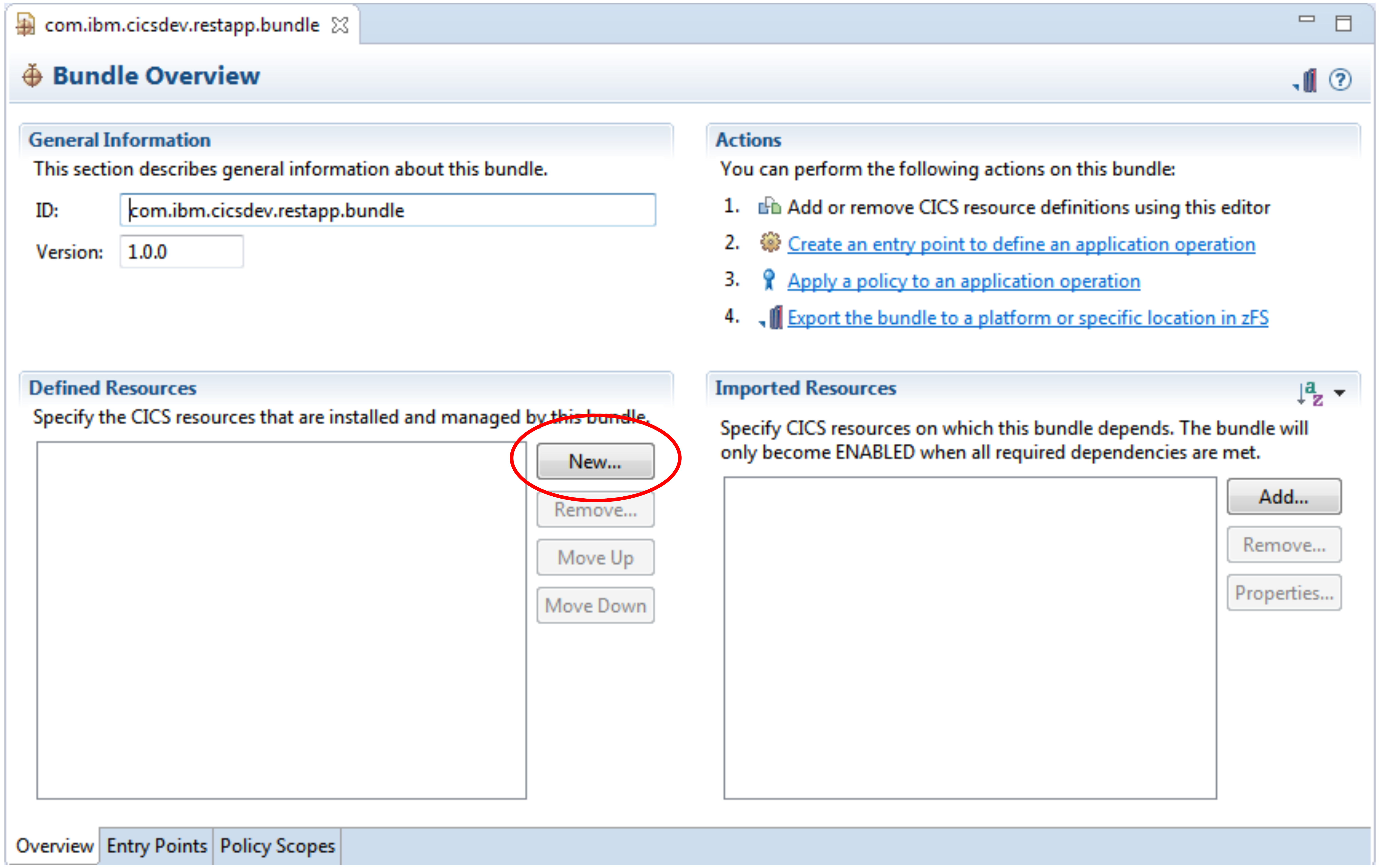
CICS bundle project



Project Explorer

- com.ibm.cicsdev.restapp
  - Deployment Descriptor: com.ibm.cicsdev.restapp
  - Java Resources
    - src
      - com.ibm.cicsdev.restapp
      - com.ibm.cicsdev.restapp.bean
    - Libraries
  - JavaScript Resources
  - build
  - WebContent
- com.ibm.cicsdev.restapp.bundle
  - META-INF
    - cics.xml

Adding a web project to a CICS bundle



com.ibm.cicsdev.restapp.bundle

Bundle Overview

General Information

This section describes general information about this bundle.

ID: com.ibm.cicsdev.restapp.bundle

Version: 1.0.0

Actions

You can perform the following actions on this bundle:

1. Add or remove CICS resource definitions using this editor
2. Create an entry point to define an application operation
3. Apply a policy to an application operation
4. Export the bundle to a platform or specific location in zFS

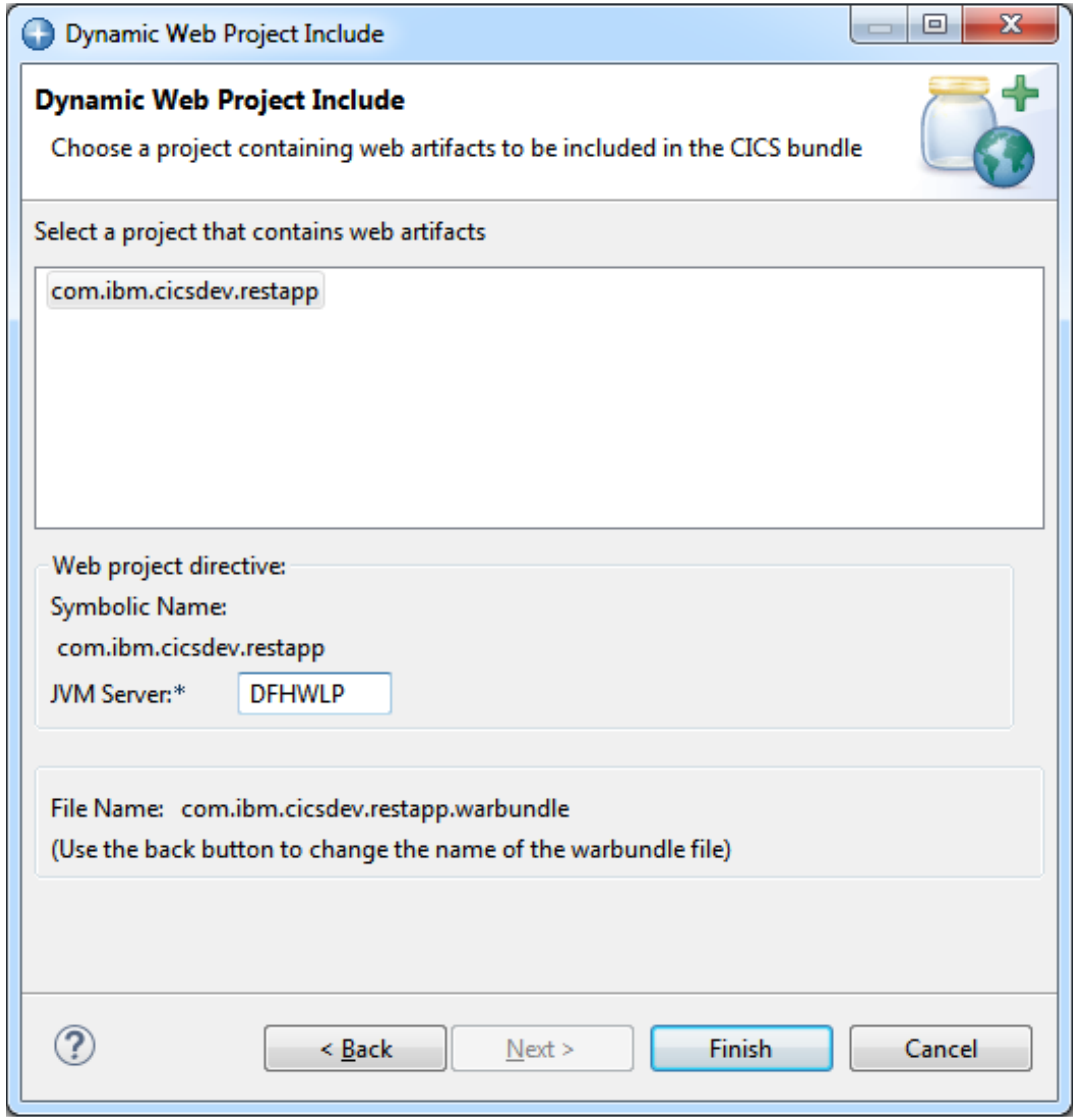
Defined Resources

Specify the CICS resources that are installed and managed by this bundle.

Imported Resources

Specify CICS resources on which this bundle depends. The bundle will only become ENABLED when all required dependencies are met.

Adding a web project to a CICS bundle



Dynamic Web Project Include

Dynamic Web Project Include

Choose a project containing web artifacts to be included in the CICS bundle

Select a project that contains web artifacts

com.ibm.cicsdev.restapp

Web project directive:

Symbolic Name:

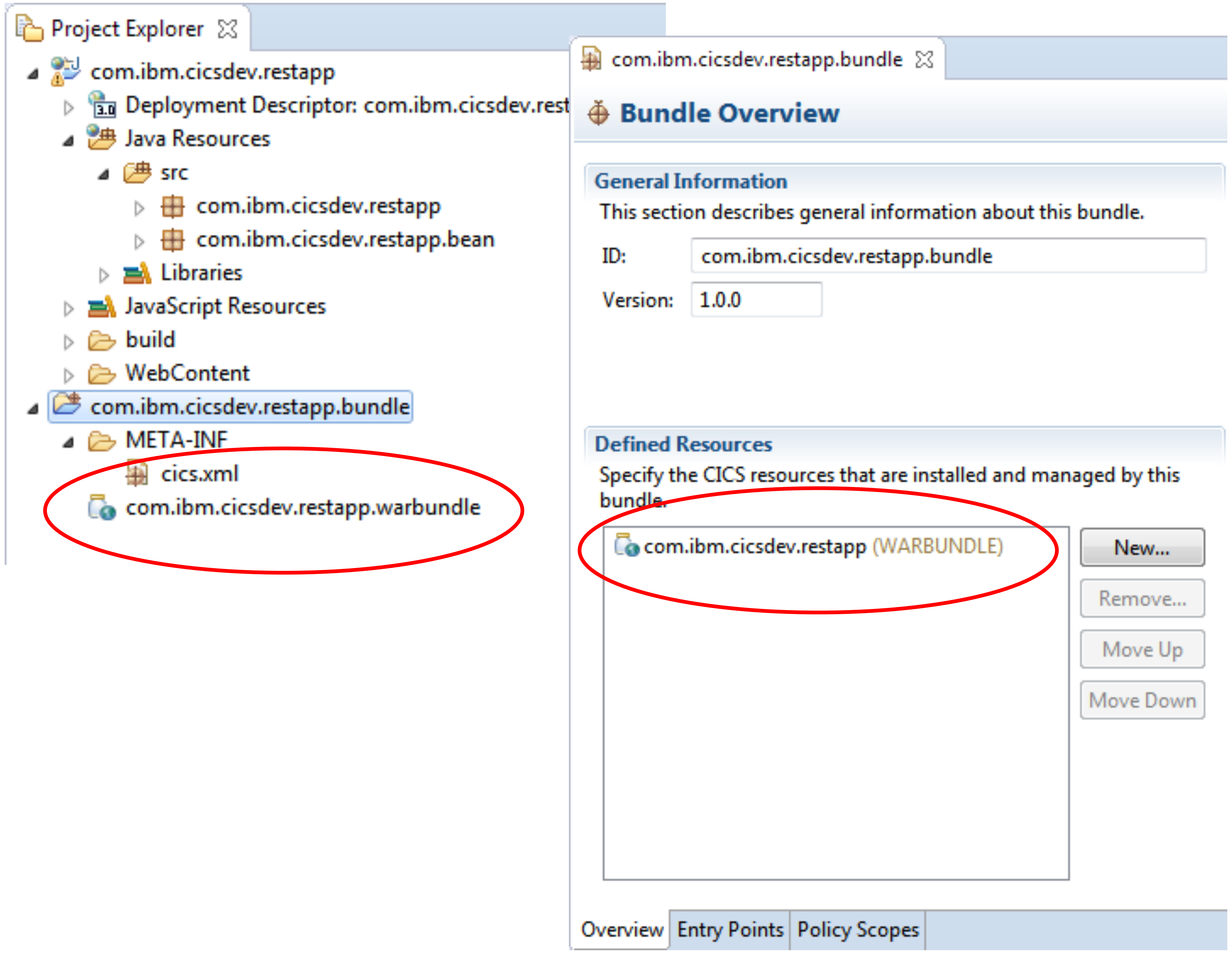
com.ibm.cicsdev.restapp

JVM Server:* DFHWLP

File Name: com.ibm.cicsdev.restapp.warbundle

(Use the back button to change the name of the warbundle file)

Adding a web project to a CICS bundle



Project Explorer

- com.ibm.cicsdev.restapp
  - Deployment Descriptor: com.ibm.cicsdev.rest
  - Java Resources
    - src
      - com.ibm.cicsdev.restapp
      - com.ibm.cicsdev.restapp.bean
    - Libraries
  - JavaScript Resources
  - build
  - WebContent
- com.ibm.cicsdev.restapp.bundle
  - META-INF
    - cics.xml
  - com.ibm.cicsdev.restapp.warbundle

com.ibm.cicsdev.restapp.bundle

Bundle Overview

General Information

This section describes general information about this bundle.

ID: com.ibm.cicsdev.restapp.bundle

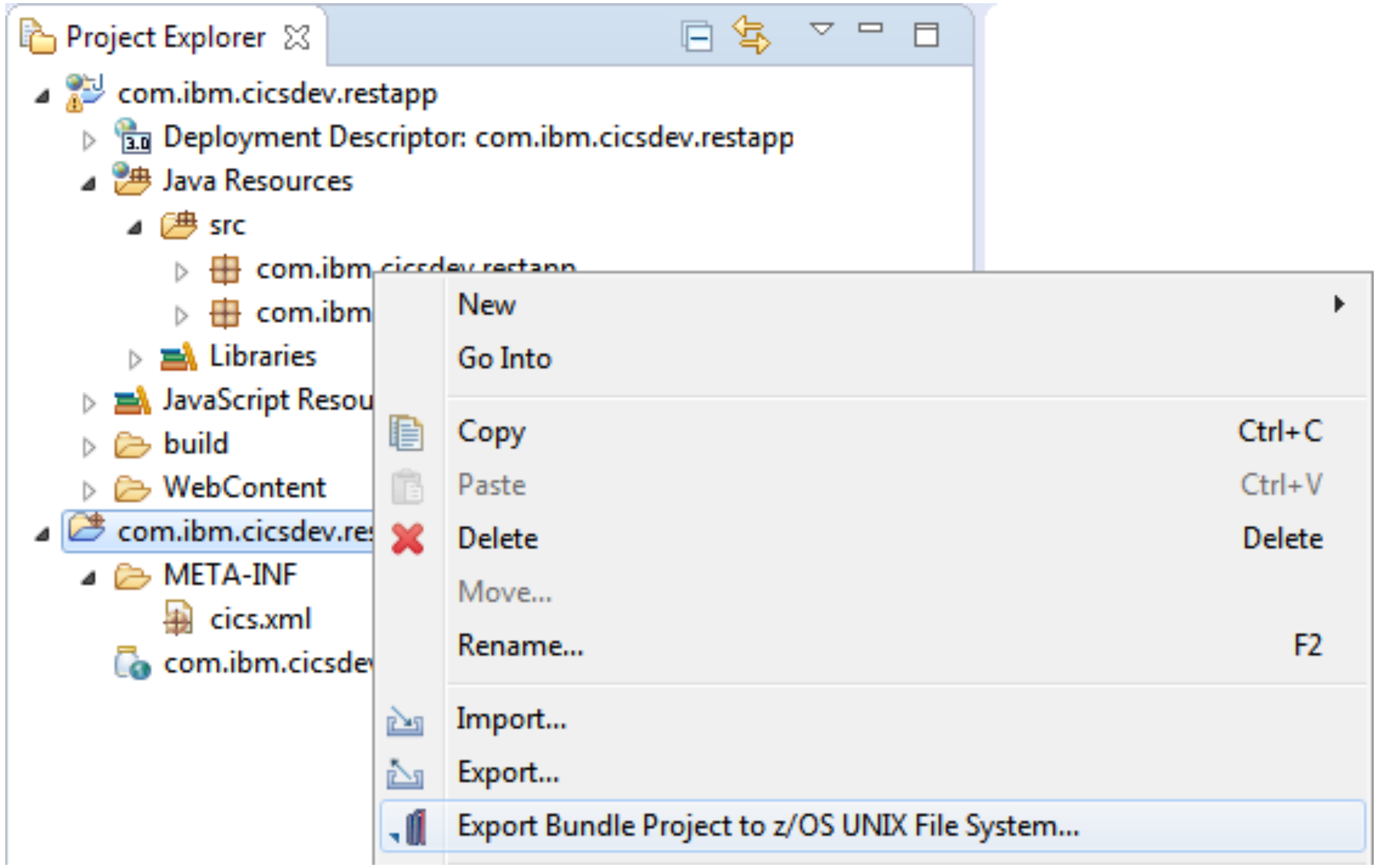
Version: 1.0.0

Defined Resources

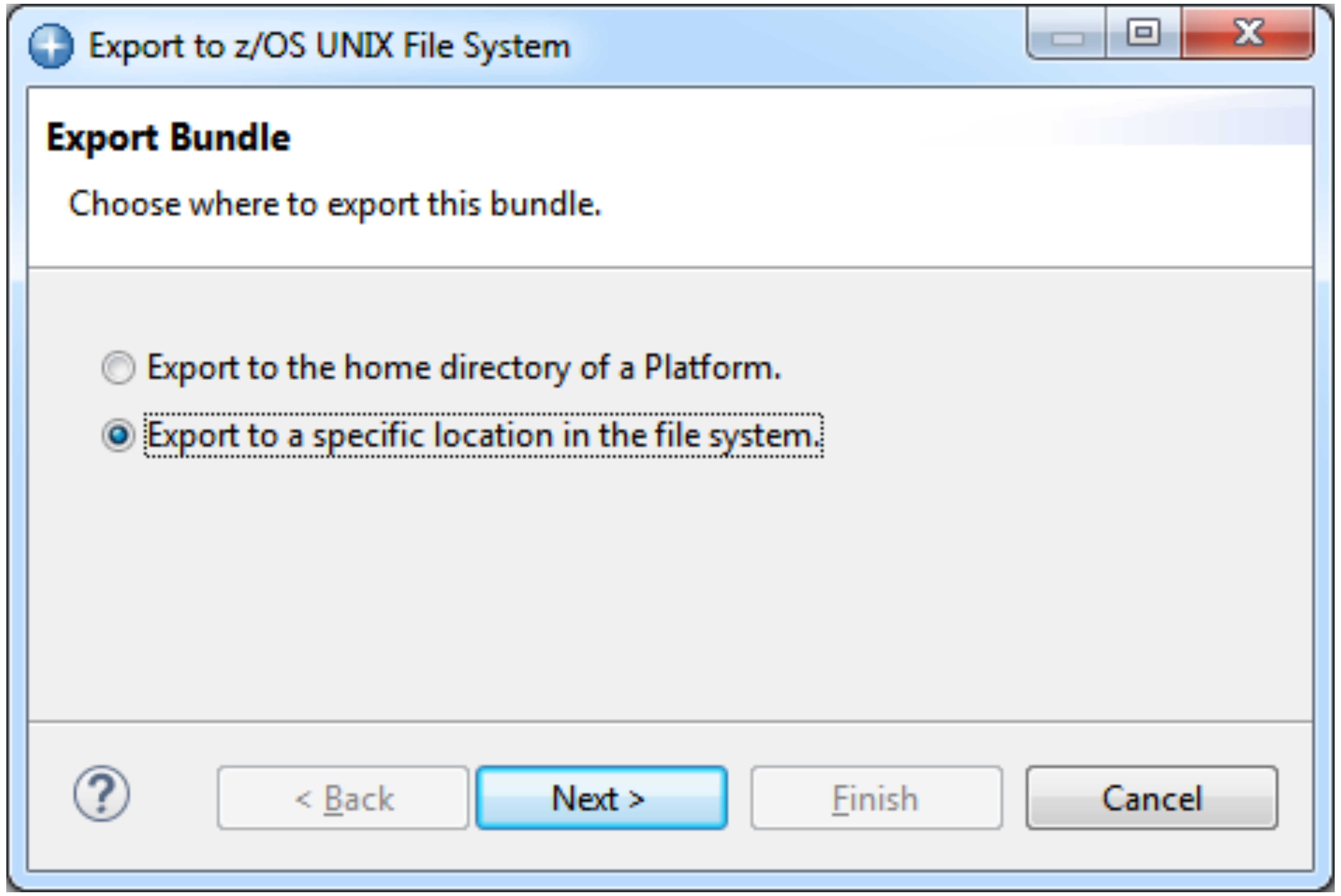
Specify the CICS resources that are installed and managed by this bundle.

com.ibm.cicsdev.restapp (WARBUNDLE)

Exporting a CICS bundle project



Exporting a CICS bundle project



Export to z/OS UNIX File System

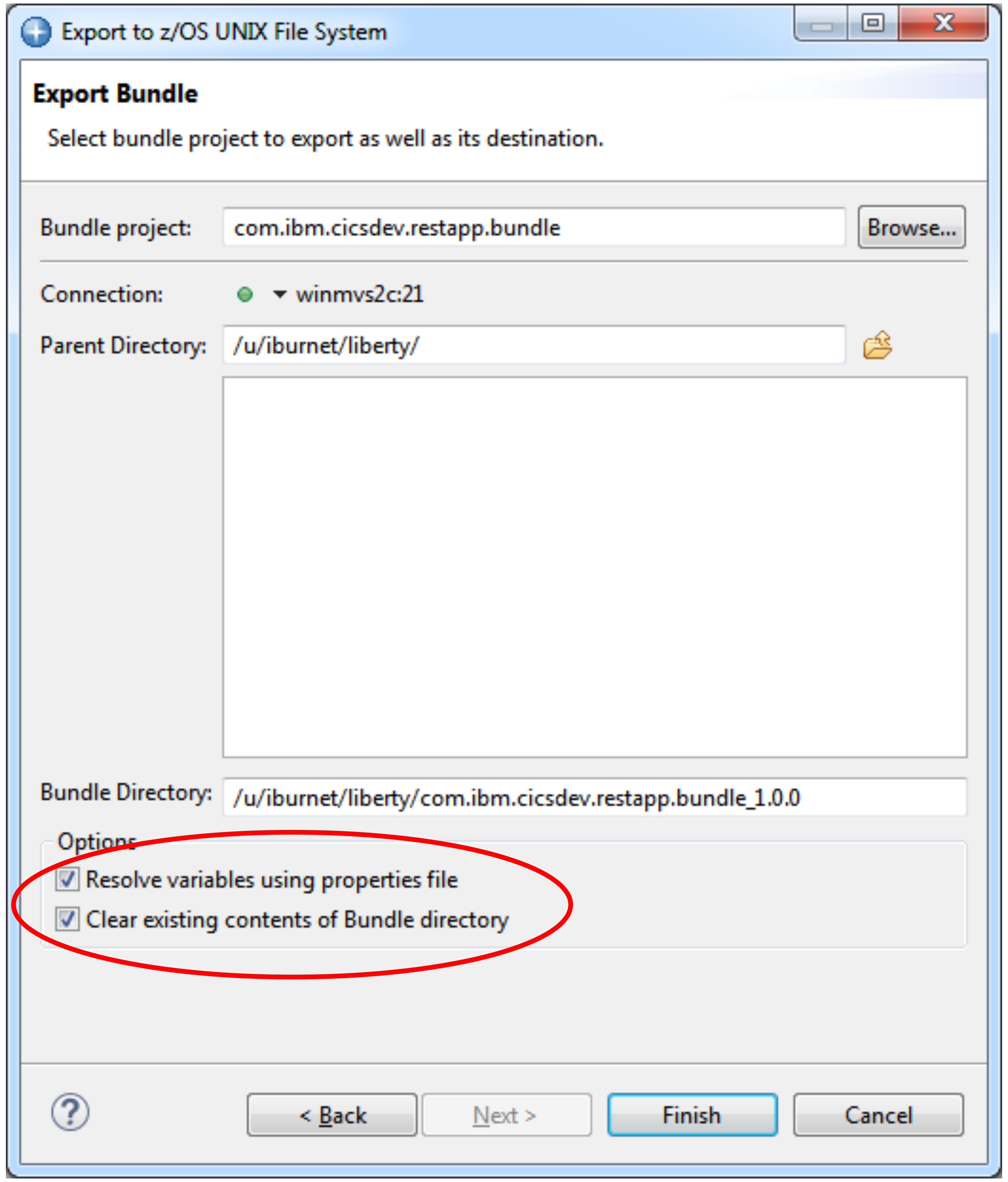
Export Bundle

Choose where to export this bundle.

○ Export to the home directory of a Platform.

● Export to a specific location in the file system.

Exporting a CICS bundle project



Export Bundle

Select bundle project to export as well as its destination.

Bundle project: com.ibm.cicsdev.restapp.bundle

Connection: ● winmvs2c:21

Parent Directory: /u/iburnet/liberty/

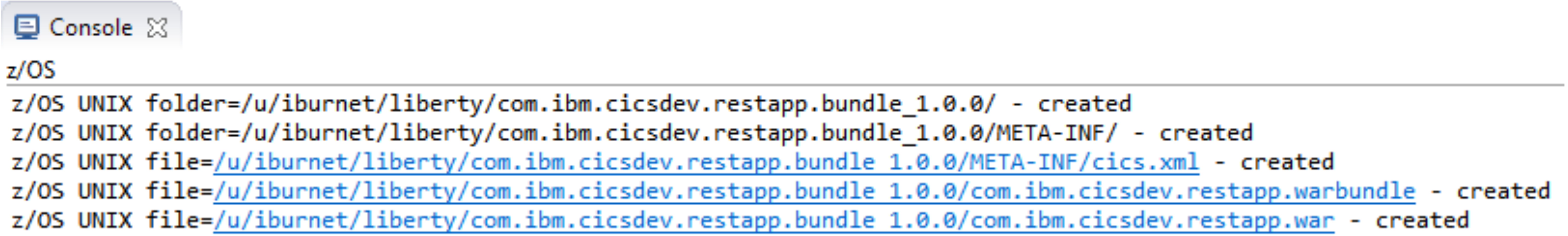
Bundle Directory: /u/iburnet/liberty/com.ibm.cicsdev.restapp.bundle_1.0.0

Options

☑ Resolve variables using properties file

☑ Clear existing contents of Bundle directory

Exporting a CICS bundle project



```
z/OS UNIX folder=/u/iburnet/liberty/com.ibm.cicsdev.restapp.bundle_1.0.0/ - created
z/OS UNIX folder=/u/iburnet/liberty/com.ibm.cicsdev.restapp.bundle_1.0.0/META-INF/ - created
z/OS UNIX file=/u/iburnet/liberty/com.ibm.cicsdev.restapp.bundle_1.0.0/META-INF/cics.xml - created
z/OS UNIX file=/u/iburnet/liberty/com.ibm.cicsdev.restapp.bundle_1.0.0/com.ibm.cicsdev.restapp.warbundle - created
z/OS UNIX file=/u/iburnet/liberty/com.ibm.cicsdev.restapp.bundle_1.0.0/com.ibm.cicsdev.restapp.war - created
```

Creating a CICS bundle definition

Creating a CICS bundle definition

Creating a CICS bundle definition

Installing a CICS bundle definition

Installing a CICS bundle definition

Installing a CICS bundle definition

Installing a CICS bundle definition

Liberty messages.log file

Testing the application

Summary

- Created sample application
- Deployed into CICS
  - Dropins directory
  - CICS bundles

Linking to a CICS COBOL program

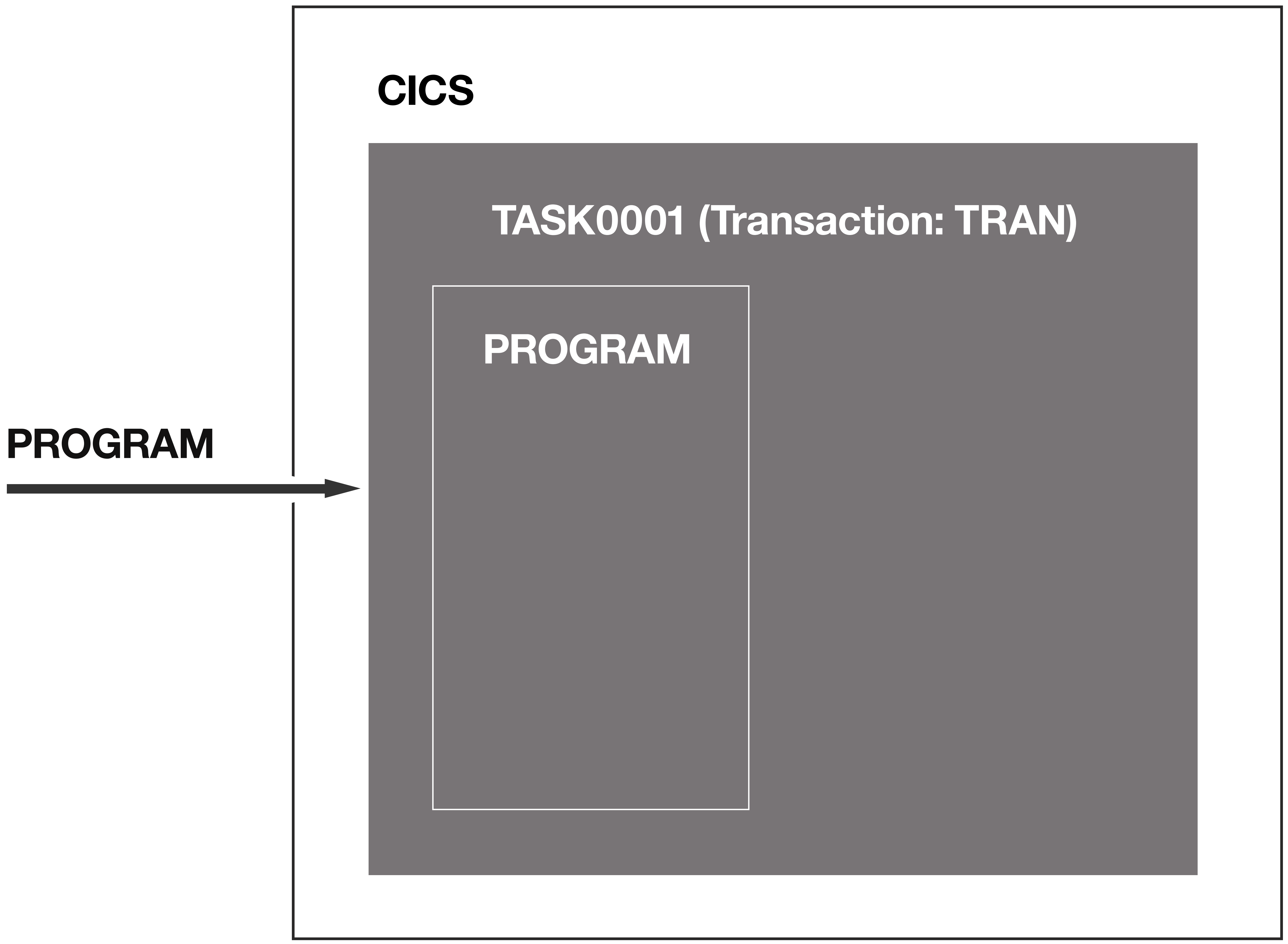
Introduction to linking to a CICS program

Introduction to linking to a CICS program

- CICS concepts
- JCICS
  - Calling an existing COBOL program
- EDUCHAN sample

`github.com/cicsdev/cics-java-liberty-restapp`

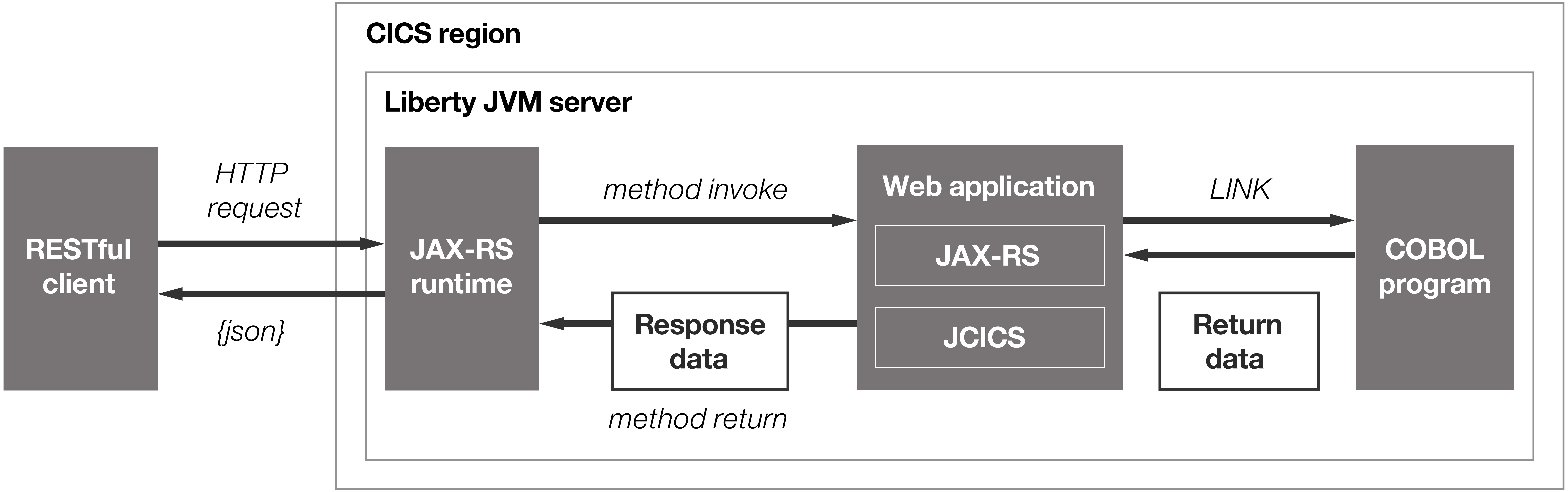
What is a task?



Linking to a CICS COBOL program

Using JCICS to link to a CICS program

Application diagram



Reviewing and updating the sample application

Reviewing and updating the sample application

Reviewing and updating the sample application

Reviewing and updating the sample application

Reviewing and updating the sample application

Deploying the application

Testing the reverse application

Test the application

Using JCICS to link to a CICS program

- Expose an existing COBOL program as a RESTful web application

Summary

- JCICS API to access CICS tasks and programs
- Passing data using channels and containers
- Next section - Summary and highlights

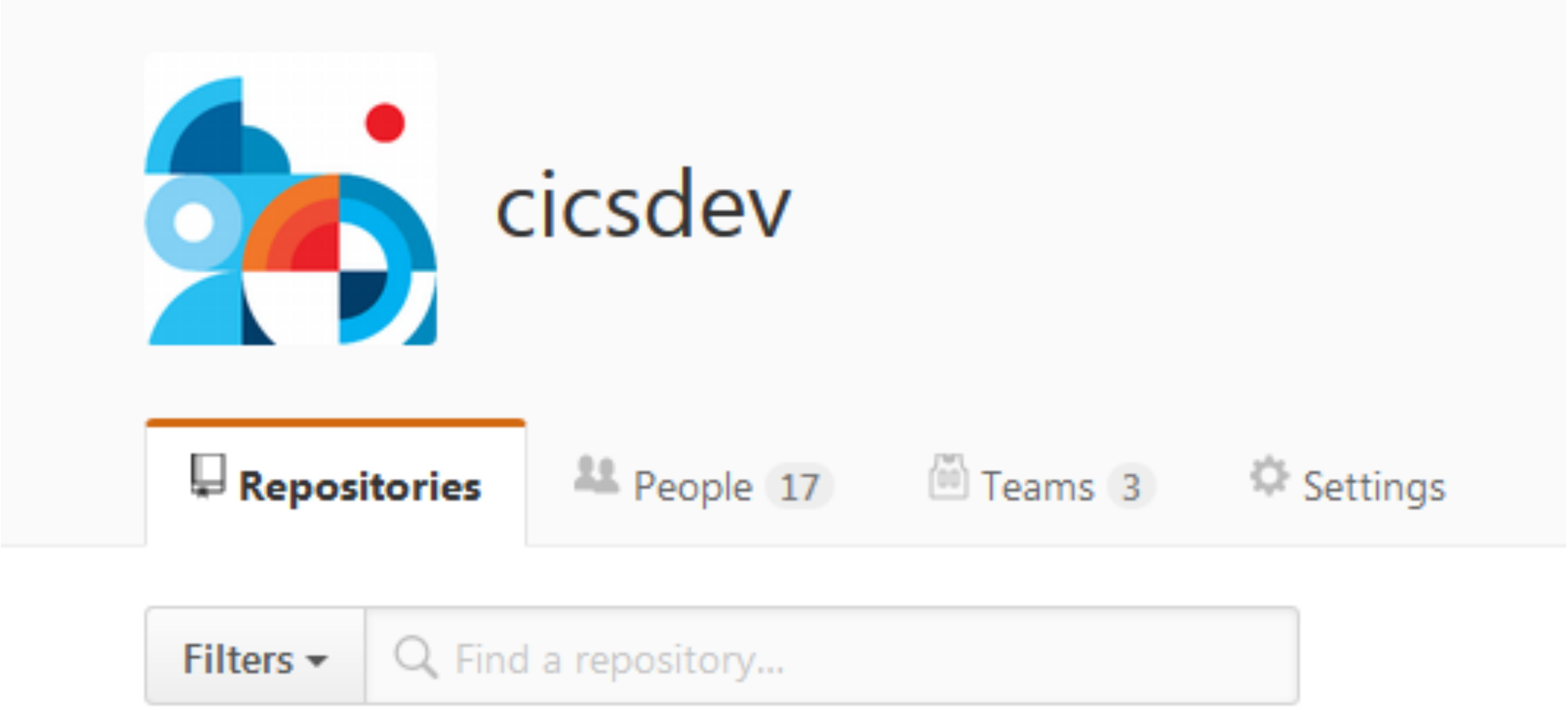
Summary

Review of what we have learned

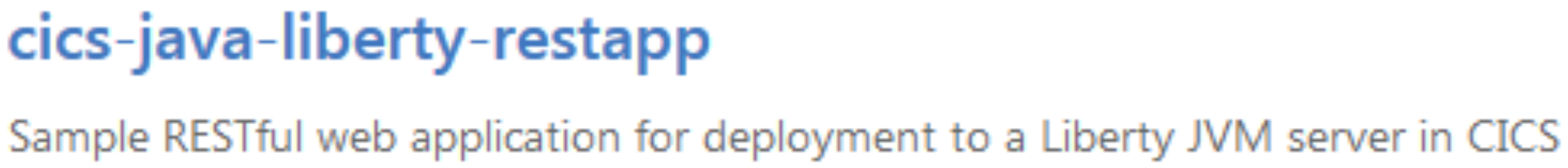
What we learned in this course

1. Creation of a development environment
2. Configuration of a Liberty JVM server
3. Development using JAX-RS
4. Use of the JCICS API
5. Application deployment

Download and install the sample applications



- Install sample RESTful web application



cics-java-liberty-restapp

Sample RESTful web application for deployment to a Liberty JVM server in CICS

- Install EDUCHAN COBOL program