

# Modernized Runtime Extension (MoRE) Lab guide

## PREREQUISITES:

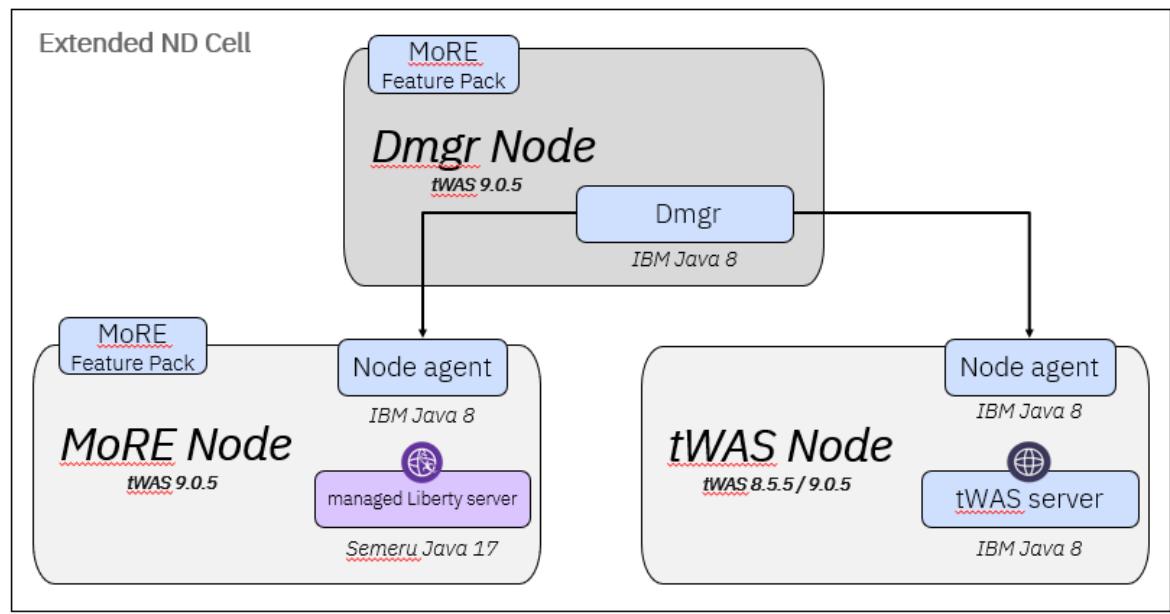
Before running this demo, you must have completed the demo setup documentation:  
<https://ibm.box.com/s/ownhud93d12fd1x3j52rxxacj5dkugv4>

## INTRODUCTION:

MoRE provides the capability to continue using traditional WebSphere Application Server (tWAS) Operational Model to manage Java 17 and Java 8 applications within the same traditional WebSphere administrative environment

## MoRE extends the ND Cell

*Existing WAS ND clients can run both Java 8 and Java 17 applications using their current operating model...*



In this lab, you will see how to extend a WebSphere ND Cell, using the MoRE feature pak, for managed Liberty servers to manage and run Java 17 / Jakarta EE 10 (subset) applications using the familiar WebSphere administrative mode and admin console.

The demo first shows an example application “Mod Resorts” running in WebSphere ND 9.0.23. The application is a simple Java 8 application using JEE 7 APIs.

Included with the demo, is a new version of the Mod Reports application that is built using Java 17 and Jakarta 10 EE APIs, which are supported in the managed Liberty Servers (MoRE).

At the end of this demo, you should be able to showcase how to:

- View the installed WebSphere / MoRE components in IBM Installation Manager
- Create a new managed Liberty server (MoRE) using the WAS admin console
- Install the new Java 17 version of the example application and target the managed Liberty server using the WAS admin console
- Start the managed Liberty server using the WAS admin console
- Locate and view the managed Liberty server logs
- From the web browser, run the new example application running Java 17 in a WebSphere cell.

1. View the installed WebSphere / MoRE components in IBM Installation Manager.

a. Open a new **terminal** window and Launch **IBM Installation Manager**

```
/home/techzone/IBM/InstallationManager_Group/eclipse/IBMIM
```

b. Click the **Uninstall** option. DO NOT UNINSTALL ANY COMPONENTS!

c. Review the installed components

Installation Packages	Version	Vendor
▼ <input checked="" type="checkbox"/>  WebSphere Liberty		
<input type="checkbox"/>  IBM WebSphere Application Server Liberty Network Deployment	25.0.0.2	IBM
<input type="checkbox"/>  IBM Semeru Runtime Certified Edition Version 17	17.0.12.1	IBM
▼ <input type="checkbox"/>  IBM WebSphere Application Server V9.0		
<input type="checkbox"/>  IBM WebSphere Application Server Network Deployment	9.0.5.23	IBM
<input type="checkbox"/>  IBM Modernized Runtime Extension for Java (MoRE)	1.0.0.0	IBM
<input type="checkbox"/>  IBM SDK, Java Technology Edition, Version 8	8.0.8.40	IBM

Alternatively, you could use the command `imcl listInstalledPackages` instead:

(Use the option `-long` to get additional information)

```
/home/techzone/IBM/InstallationManager_Group/eclipse/tools/imcl  
listInstalledPackages -long
```

2. Start if needed the WebSphere Application Server, view the installed applications
    - a. Check the status of your application server environment and if necessary, start it
- Deployment manager

```
To check the status:  
/home/techzone/IBM/WebSphere/AppServer/profiles/Dmgr01/bin/serverStatus  
.sh -all  
  
To start the Dmgr:  
/home/techzone/IBM/WebSphere/AppServer/profiles/Dmgr01/bin/startManager  
.sh
```

#### Node agent

```
To check the status:  
/home/techzone/IBM/WebSphere/AppServer/profiles/AppSrv01/bin/serverStat  
us.sh -all  
  
To start the Node agent:  
/home/techzone/IBM/WebSphere/AppServer/profiles/AppSrv01/bin/startNode.  
sh
```

- b. Launch the **Chrome** browser in the demo environment
  - c. Navigate to the **WebSphere Admin Console**, and login, if not already logged in.

<https://localhost:9043/ibm/console>

User ID: techzone

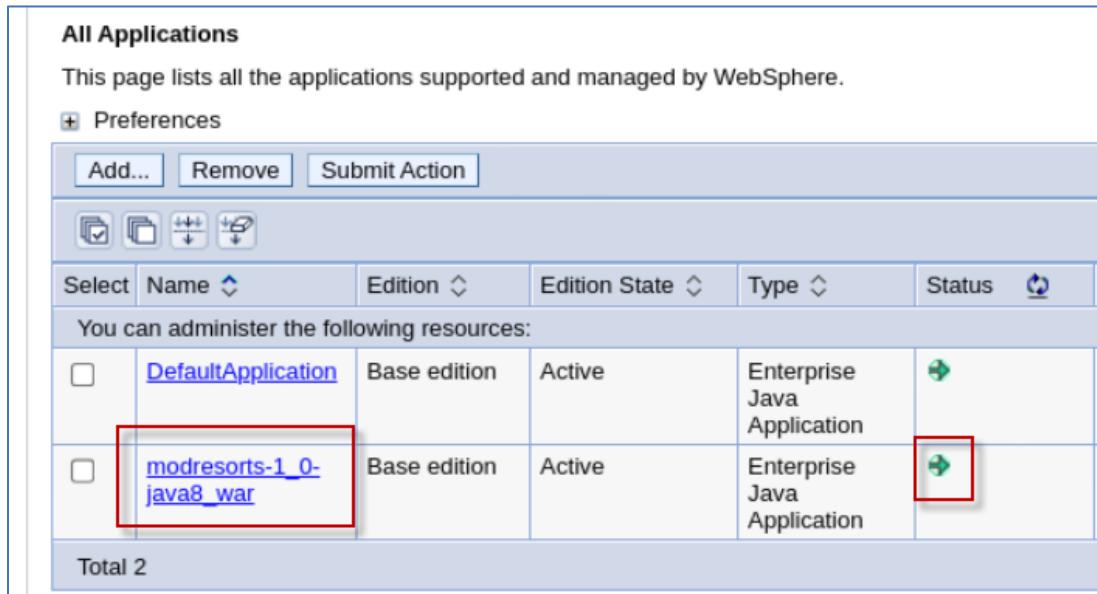
Password: IBMDem0s!

- d. Navigate to **Server Types > WebSphere application servers**
  - e. Start **server1** that is running on traditional WebSphere 9.0.0.5.23

The server1 is now in the “**started**” state

f. Navigate to Applications > All applications

The Mod Resorts Java 8 application should be **running**



All Applications

This page lists all the applications supported and managed by WebSphere.

Preferences

Add... Remove Submit Action

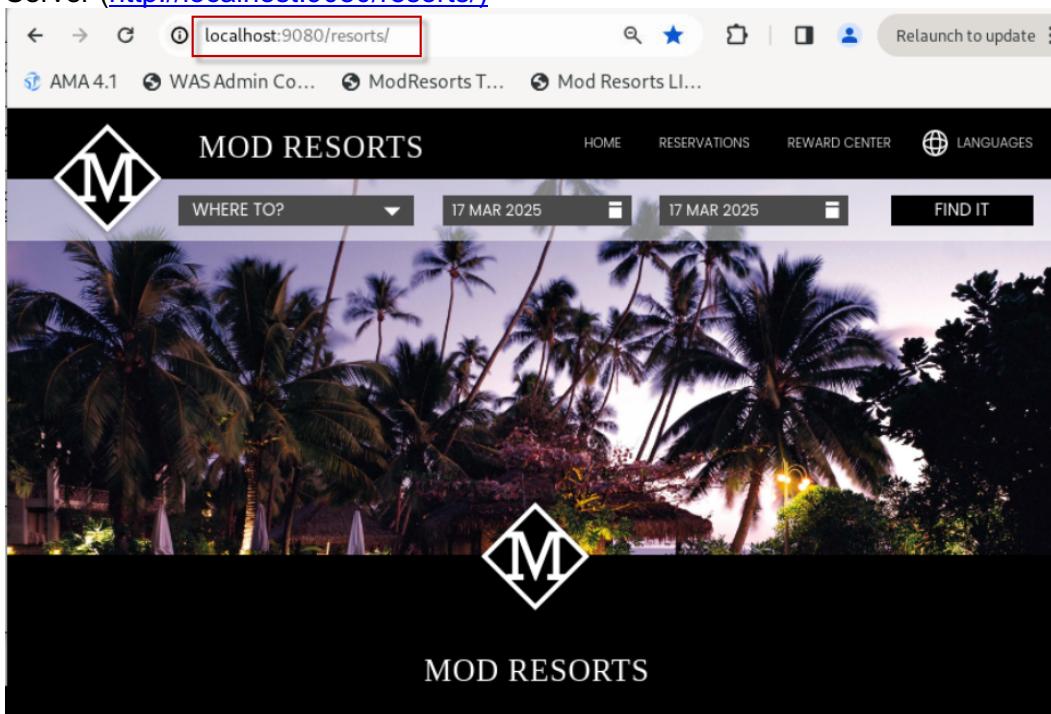
Select Name Edition Edition State Type Status

You can administer the following resources:

	Name	Edition	Edition State	Type	Status
<input type="checkbox"/>	<a href="#">DefaultApplication</a>	Base edition	Active	Enterprise Java Application	
<input type="checkbox"/>	<a href="#">modresorts-1_0-java8_war</a>	Base edition	Active	Enterprise Java Application	

Total 2

3. Run the original version of Mod Resorts application
  - a. Open a new browser tab in **Chrome** browser in the demo environment
  - b. Navigate to the original Mod Resorts application running on WebSphere Application Server (<http://localhost:9080/resorts/>)



localhost:9080/resorts/ Relaunch to update :

AMA 4.1 WAS Admin Co... ModResorts T... Mod Resorts LI...

MOD RESORTS HOME RESERVATIONS REWARD CENTER LANGUAGES

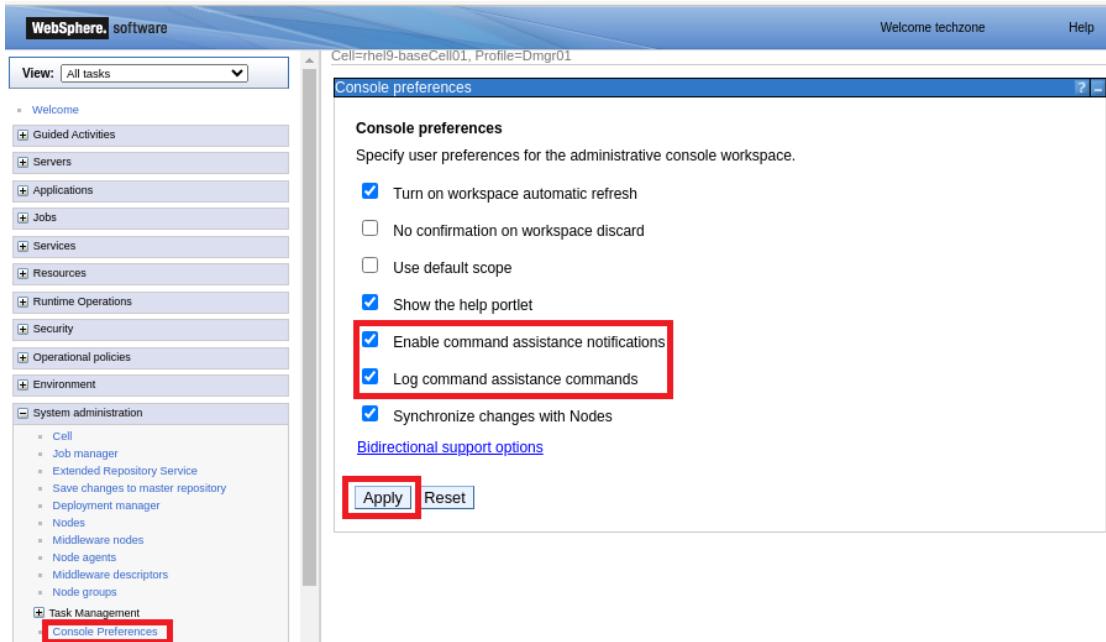
WHERE TO? 17 MAR 2025 17 MAR 2025 FIND IT

MOD RESORTS

4. Enable the command assistant so that you can see the wsadmin commands related to the steps done in the console.

a. Navigate to **System administration > Console Preferences**.

Select the preferences **Enable command assistant notification** and **Log command assistance commands**, then click on **Apply**.



5. Create a new Managed Liberty Server using the WebSphere Admin Console.
  - a. From the WAS Admin Console, navigate to **Servers > Server Types > WebSphere application Servers**
  - b. Click the **New...** Button
  - c. Select **Managed Liberty server** from the drop-down menu

The screenshot shows the WebSphere Admin Console interface. On the left, there's a navigation tree under 'Servers' with 'Server Types' expanded, showing options like 'WebSphere application servers' (which is highlighted with a red box). On the right, the main window is titled 'Application servers' with the sub-header 'Application servers'. It says, 'Use this page to view a list of the application servers in your environment and the status of each servers. You can also use this page to change the status of a specific application server.' Below this is a toolbar with buttons: 'New...', 'Delete', 'Templates...', 'Start', 'Stop', 'Restart', 'ImmediateStop', and 'Terminal'. A table follows, with the first row showing a checkbox, the name 'server1', the node 'rhel9-baseNode01', the host name 'rhel9-base.gym.lan', the version 'ND 9.0.5.23', and an empty column. The total count is shown as 'Total 1'.

Complete the steps in the “**Create a new application server**” dialog to create the managed Liberty Server

- d. **Step 1:** Select a node and specify managed Liberty server name (Node must be running 9.0.5.23 or higher)
  - Select node: **rhel9-baseNode01 (ND 9.0.5.23)**
  - Server name: **LibertyServer2**
- e. **Step 2:** Select a server template
  - Accept the default Managed Liberty server template
- f. **Step 3:** Specify properties
  - Ensure **Generate Unique Ports** is **checked** (to avoid port conflicts with WAS servers on the same machine)
- g. **Step 4:** confirm new server
  - Click **Finish** to create the new managed Liberty Server

- h. As you enabled command assistance notification, you can see the related wsadmin command. Click on **View administrative scripting command for last action**.

Cell=rhel9-baseCell01, Profile=Dmgr01

Close page

Application servers

Messages

- i New server is created successfully.
- i Modify variables, resources, and other server configuration settings, such as message broker queue names before running the newly created server.
- ! Changes have been made to your local configuration. You can:
  - [Save](#) directly to the master configuration.
  - [Review](#) changes before saving or discarding.

An option to synchronize the configuration across multiple nodes can be disabled in [Preferences](#).

! The server may need to be restarted for these changes to take effect.

Help

Field help  
For field help information, select a field label or list marker when the help cursor is displayed.

Page help  
[More information about this page](#)

Command Assistance  
[View administrative scripting command for last action](#)

- i. The wsadmin command to create a Liberty server is shown in a pop-up window.

Administrative Scripting Commands

### Administrative Scripting Commands

The wsadmin scripting commands that map to actions on the administrative console display in the Jython language.

⊕ Preferences

Administrative Scripting Command
<a href="#">AdminTask.createManagedLibertyServer('rhel9-baseNode01', ['-name LibertyServer2 -templateName default -genUniquePorts true'])</a>
<a href="#">AdminTask.listServers(['-serverType APPLICATION_SERVER'])</a>
<a href="#">AdminTask.listServers(['-serverType MANAGED_LIBERTY_SERVER'])</a>
Total 3

Finally close the pop-up window.

- j. **Save** the changes to synchronize the master configuration

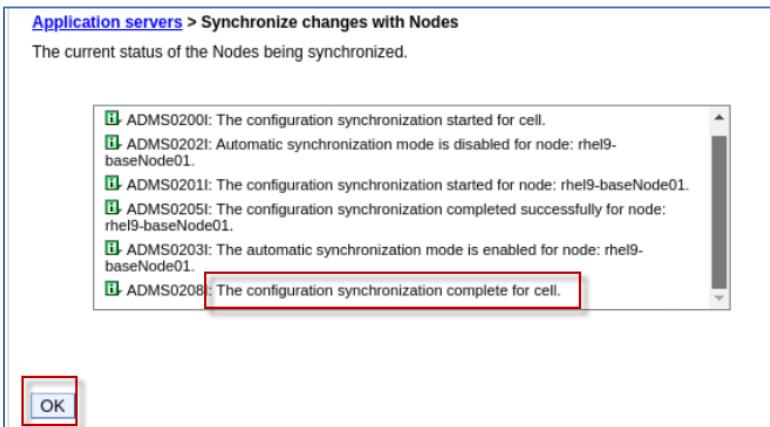
Messages

- i New server is created successfully.
- i Modify variables, resources, and other server configuration settings, such as message broker queue names before running the newly created server.
- ! Changes have been made to your local configuration. You can:
  - [Save](#) directly to the master configuration.
  - [Review](#) changes before saving or discarding.

An option to synchronize the configuration across multiple nodes can be disabled in [Preferences](#).

! The server may need to be restarted for these changes to take effect.

- k. When the configuration synchronization is complete, click the **OK** button



- l. The new managed Liberty server is now created, and in the “**stopped**” state

**Application servers**

Use this page to view a list of the application servers in your environment and the status of each of these servers. You can also use this page to change the status of a specific application server.

**Preferences**

New...	Delete	Templates...	Start	Stop	Restart	ImmediateStop	Terminate
<input type="checkbox"/>							
Select	Name	Node	Host Name	Version	Cluster Name	Status	

You can administer the following resources:

<input type="checkbox"/>	<a href="#">LibertyServer2</a>	rhel9-baseNode01	rhel9-base.gym.lan	LIBERTY 25.0.0.2 MORE 1.0.0.0	
<input type="checkbox"/>	<a href="#">server1</a>	rhel9-baseNode01	rhel9-base.gym.lan	ND 9.0.5.23	

Total 2

6. Install a modernized version of mod resorts application (java 17 / Jakarta EE 10) to the managed Liberty server
  - a. From WAS Admin console, navigate to **Applications > New Application**
  - b. Click the **New Enterprise Application** link and complete the steps in the **Install application dialog**
  - c. Path to the new application:
    - Click the **Choose File** button and navigate to the modernized Mod Resorts WAR file on the local demo environment.

**HOME > techzone > demos > ManagedLiberty-MoRE > modresorts-2.0.0-java17.war**

- Then click the **Select** button on the **Open File** dialog.

- d. Target Runtime Environment

- Select **WebSphere Liberty** from the list of Target Runtimes

The screenshot shows the 'Preparing for the application installation' dialog. In the 'Path to the new application' section, the 'Local file system' radio button is selected, and the 'Full path' field contains 'modresorts-...-java17.war'. In the 'Target Runtime Environment' section, the 'Target Runtime' dropdown is set to 'WebSphere Liberty'. At the bottom of the dialog are 'Next' and 'Cancel' buttons.

Then click the **Next** button

- e. How do you want to install the application
  - Select the **Fast Path** radio button as the installation options.
  - Then click the **Next** button
- f. Step 1: Select installation options
  - Accept the default values and click the **Next** button
- g. Step 2: Map modules to servers
  - **Check** the box next to the modresorts-2.0.0 module
  - Ensure the **LibertyServer2** (or name of your managed Liberty server) is the target server
  - Click the **Apply** button

- Click **Next** to continue

**Step 1 Select installation options**

**→ Step 2: Map modules to servers**

**Step 3 Map context roots for Web modules**

**Step 4 Summary**

**Map modules to servers**

Specify targets such as application servers or clusters of application servers where you want to install the modules that are contained in your application. Modules can be installed on the same application server or dispersed among several application servers. Also, specify the Web servers as targets that serve as routers for requests to this application. The plug-in configuration file (plugin-cfg.xml) for each Web server is generated, based on the applications that are routed through.

Clusters and servers:

WebSphere:cell=rhel9-baseCell01,node=rhel9-baseNode01,server=LibertyServer2
---

**Apply**

**Select Module URI Server**

<input checked="" type="checkbox"/>	modresorts-2.0.0-java17.war	modresorts-2.0.0-java17.war	WebSphere:cell=rhel9-baseCell01,node=rhel9-baseNode01,server=LibertyServer2
-------------------------------------	-----------------------------	-----------------------------	---

#### h. Step 3: map Context roots for web modules

- Accept the default context root of “/resorts” for the web module
- Click **Next** to continue

#### i. Step 4: Summary

- On the summary page, click on **View administrative scripting command for last action**.

Cell=rhel9-baseCell01, Profile=Dmgr01 Close page

Install New Application

Specify options for installing enterprise applications and modules.

**Summary**

Options	Values
Directory to install application	
Distribute application	Yes
Use Binary Configuration	Yes
Application name	modresorts-2_0_java17_war

Application edition

**Help**

**Field help**  
For field help information, select a field label or list marker when the help cursor is displayed.

**Page help**  
[More information about this page](#)

**Command Assistance**  
[View administrative scripting command for last action](#)

- The wsadmin command to install an application to the Liberty server is shown.

**Administrative Scripting Commands**

**Administrative Scripting Commands**

The wsadmin scripting commands that map to actions on the administrative console display in the Jython language.

Preferences

Administrative Scripting Command
AdminApp.install('C:/fakepath/modresorts-2.0.0-java17.war', ['-distributeApp -useMetaDataFromBinary -appname modresorts-2_0_java17_war -noreloadEnabled -validateInstall warn -filepermission *.dll=755# *.so=755# *.a=755# *.sl=755 -noallowDispatchRemoteInclude -noallowServiceRemoteInclude -asyncRequestDispatchType DISABLED -noenableClientModule -clientMode isolated -novalidateSchema -contextroot /resorts -MapModulesToServers [[ modresorts-2.0.0-java17.war modresorts-2.0.0-java17.war,WEB-INF/web.xml WebSphere:cell=rhel9-baseCell01,node=rhel9-baseNode01,server=LibertyServer2 ]]]')
Total 1

- Close the pop-up window, then click the **Finish** button to install the mod resorts application to the managed Liberty server
  - Click the **Save** link to synchronize the changes to the master configuration.
  - Click the **OK** button one the configuration is synchronized
7. Start the new managed Liberty **server** from the WAS admin console
- a. Navigate to **Servers > All servers**
  - b. Select the managed Liberty server “**LibertyServer2**” from the server list and click the **Start** button.

The managed Liberty Server should now be running

**Middleware servers**

Use this page to view a list of all middleware servers such as WebSphere Application Server, generic server, proxy server, ODR, etc. in your environment. You can also use this page to change the status of a specific application server.

Preferences

New...	Delete	Templates...	Start	Stop	Terminate	Submit Action	Select mode	Set mode
<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Select	Name	Node	Cluster Name	Status	Maintenance mode	Version	Type	
You can administer the following resources:								
<input type="checkbox"/>	<a href="#">LibertyServer2</a>	rhel9-baseNode01		LIBERTY 25.0.0.2	Managed Liberty server			
<input type="checkbox"/>	<a href="#">server1</a>	rhel9-baseNode01		ND 9.0.5.23	WebSphere application serv			
Total 2								

8. View the console log of the managed Liberty server

The managed Liberty server configuration and logs are in the following directory structure:

/home/techzone/IBM/WebSphere/AppServer/profiles/AppSrv01/managedLiberty/usr/servers

- a. Use the **gedit** editor on the dem environment to view the console log.

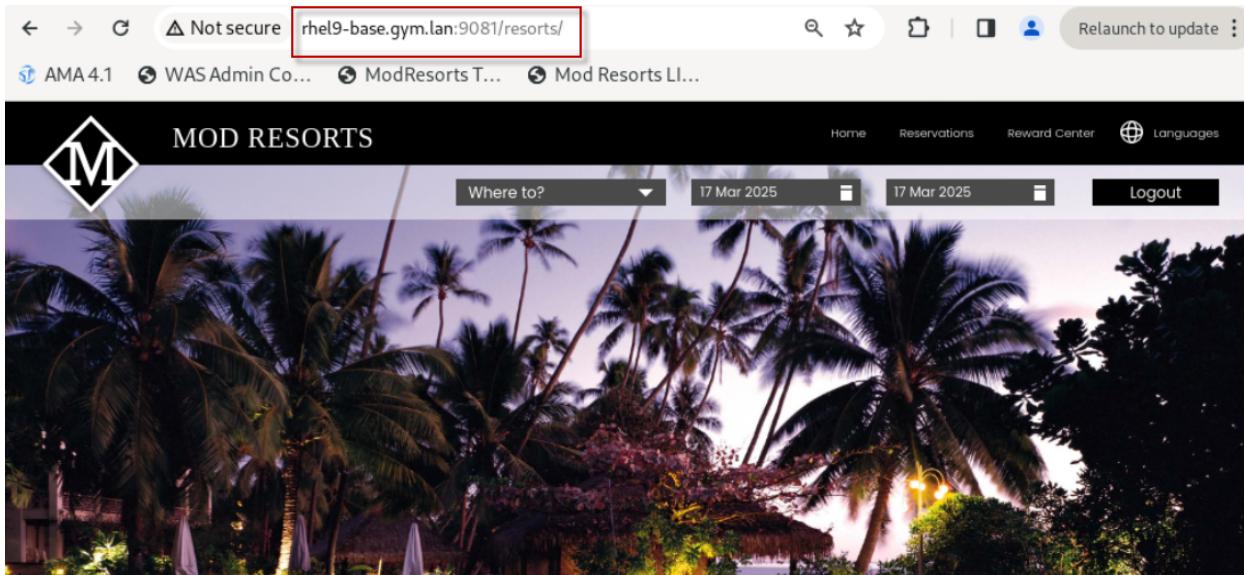
```
gedit  
/home/techzone/IBM/WebSphere AppServer/profiles/AppSrv01/managedLiberty/usr/serv  
ers/LibertyServer2/logs/console.log
```

- b. Note the following in the log:

- The managed Liberty server is running Liberty 25.0.0.2
- The managed liberty server is running Java 17
- Note the features that were installed on the managed Liberty server
- The URL for the mod resorts application is: <http://rhel9-base.gym.lan:9081/resorts>

```
console.log  
~/IBM/WebSphere/AppServer/profiles/AppSrv01/...agedLiberty/usr/servers/LibertySer...  
1 Launching LibertyServer2 (WebSphere Application Server 25.0.0.2/wlp-1.0.98.cl250220250209-1902) on Eclipse  
OpenJ9 VM, version 17.0.12+7 [en_US]  
2 [AUDIT ] CWWKE0001I: The server LibertyServer2 has been launched.  
3 [AUDIT ] CWWKG0028A: Processing included configuration resource: /home/techzone/IBM/WebSphere/AppServer/  
profiles/AppSrv01/managedLiberty/usr/servers/LibertyServer2/_server_sensitiveData.xml  
4 [AUDIT ] CWWKG0028A: Processing included configuration resource: /home/techzone/IBM/WebSphere/AppServer/  
profiles/AppSrv01/managedLiberty/usr/servers/LibertyServer2/rhel9-baseCell01_mime-types.xml  
5 [AUDIT ] CWWKZ0058I: Monitoring dropins for applications.  
6 [WARNING ] CWPKI0041W: One or more key stores are using the default password.  
7 [WARNING ] CWWKZ0126W: The /resorts context root value that is specified on the modresorts-2_0_0-javal7_war  
application has no effect.  
8 [AUDIT ] CWWKT0016I: Web application available (default_host): http://rhel9-base.gym.lan:9081/resorts/  
9 [AUDIT ] CWWKZ0001I: Application modresorts-2_0_0-javal7_war started in 0.571 seconds.  
10 [AUDIT ] CWWKF0012I: The server installed the following features: [appSecurity-5.0, cdi-4.0,  
distributedMap-1.0, expressionLanguage-5.0, federatedRegistry-1.0, jdbc-4.3, jndi-1.0, jsonb-3.0, jsonp-2.1,  
ldapRegistry-3.0, restfulWS-3.1, restfulWSClient-3.1, servlet-6.0, ssl-1.0, transportSecurity-1.0].  
11 [AUDIT ] CWWKF0011I: The LibertyServer2 server is ready to run a smarter planet. The LibertyServer2 server  
started in 3.290 seconds.
```

9. Run the modernized Mod Resorts application on the new managed Liberty Server
- Open a new tab in the **Chrome** browser on the demo environment
  - Go to URL: <http://rhel9-base.gym.lan:9081/resorts>



## 10. Summary

In this demo, we learned how Java 17 applications can be deployed and managed via managed Liberty servers in an existing WebSphere Application Server – ND cell.

In this scenario, we used the very familiar WebSphere Administrative console to create the new managed Liberty server, install Java 17 applications to the server, start the server, view the console log, and run the modernized example application in the same WebSphere cell where other Java 8 applications are also running.

Now some extended stuff:

### Part 1: Change some settings in the managed Liberty

- 1 Open a terminal window and use tail to monitor the Liberty logs.

```
tail -f  
/home/techzone/IBM/WebSphere AppServer/profiles/AppSrv01/managedLiberty/usr/serv  
ers/LibertyServer2/logs/messages.log
```

- 2 Switch back to the browser and open the All servers view in the administration console
  - a. Navigate to **Servers > All servers**
  - b. Click on the managed Liberty server "**LibertyServer2**" from the server list.

The screenshot shows the WebSphere Application Server Administration Console. On the left, there's a sidebar with a 'View' dropdown set to 'All tasks'. Below it are sections for 'Welcome', 'Guided Activities', 'Servers' (with 'All servers' highlighted with a red box), and 'Server Types' (listing various server types like WebSphere application servers, Liberty profile servers, etc.). The main content area is titled 'Middleware servers' and displays a table of servers. The table has columns for 'Select', 'Name', 'Node', 'Cluster Name', 'Status', 'Maintenance mode', 'Version', 'Type', and 'Action'. Two rows are visible: one for 'LibertyServer2' (node 'rhel9-baseNode01', status green, version 'LIBERTY 25.0.0.2 MORE 1.0.0.0', type 'Managed Liberty server') and another for 'server1' (node 'rhel9-baseNode01', status green, version 'ND 9.0.5.23', type 'WebSphere application server'). A note at the top of the table says: 'Use this page to view a list of all middleware servers such as WebSphere Application Server, generic server, proxy server, ODR, etc. in your environment and the status of each of these servers. You can also use this page to change the status of a specific application server.' There are also 'New...', 'Delete', 'Templates...', 'Start', 'Stop', 'Terminate', 'Submit Action', and 'Select mode' buttons at the top of the table.

- c. Your panel should look like this:

- d. Click on **Ports**

- e. Click on **Port Name** to change the sorting. Now you shozkld see the WC\_defaulthost close the top. As you can see the port is shown as 9081. Click on **WC\_defaulthost**.

Select	Port Name	Host	Port	Transport Details
<input type="checkbox"/>	WC_defaulthost_secure	*	9444	<a href="#">View associated transports</a>
<input type="checkbox"/>	WC_defaulthost	*	9081	<a href="#">View associated transports</a>
<input type="checkbox"/>	SOAP_CONNECTOR_ADDRESS	rhel9-base.gym.lan	8925	No associated transports

- f. Change the port from 9081 to 9082.

Middleware servers

[Middleware servers](#) > [LibertyServer2](#) > [Ports](#) > WC\_defaulthost

Specifies the TCP/IP ports this server uses for connections.

Configuration

General Properties

Port Name: WC\_defaulthost

\* Host: \*

\* Port: 9082

Apply OK Reset Cancel

- g. Click on **Apply**.
- h. Take a look at the command assistant command.

Cell=rhel9-baseCell01, Profile=Dmgr01

Close page

Middleware servers

Messages

- Changes have been made to your local configuration. You can:
  - Save directly to the master configuration.
  - Review changes before saving or discarding.

An option to synchronize the configuration across multiple nodes can be disabled in [Preferences](#).  
 ▲ The server may need to be restarted for these changes to take effect.

Help

Field help  
For field help information, select a field label or list marker when the help cursor is displayed.

Page help  
[More information about this page](#)

Command Assistance  
[View administrative scripting command for last action](#)

Middleware servers > LibertyServer2 > Ports > WC\_defaulthost

Administrative Scripting Commands

**Administrative Scripting Commands**

The wsadmin scripting commands that map to actions on the administrative console display in the Jython language.

+ Preferences

Administrative Scripting Command

```
AdminTask.modifyServerPort('LibertyServer2', ['-nodeName rhel9-baseNode01 -endPointName WC_defaulthost -host * -port 9082 -modifyShared true'])
```

Total 1

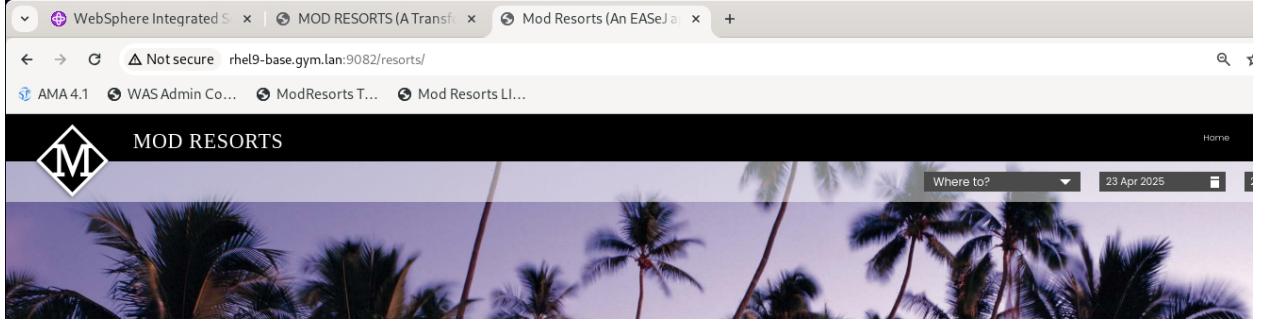
- i. Close the window and click **Save** to save the changes.
- j. Switch to the terminal window and you should see that Liberty is now listening on port 9082.

```

techzone@rhel9-base:~ — tail -f /home/techzone/IBM/WebSphere/AppServer/profiles/AppSrv01/managedLiberty/usr/server...
I CWWK00220I: TCP Channel defaultHttpEndpoint has stopped listening for requests on host * (IPv6) port 9081.
A CWWKT0028I: Web application moved (default_host): https://rhel9-base.gym.lan:9444/resorts/
A CWWKG0017I: The server configuration was successfully updated in 0.625 seconds.
I CWWK00219I: TCP Channel defaultHttpEndpoint has been started and is now listening for requests on host * (IPv6) port 9082.
I SRVE9103I: A configuration file for a web server plugin was automatically generated for this server at /home/techzone/IBM/WebSphere/AppServer/profiles/AppSrv01/managedLiberty/usr/servers/LibertyServer2/logs/state/plugin-cfg.xml.

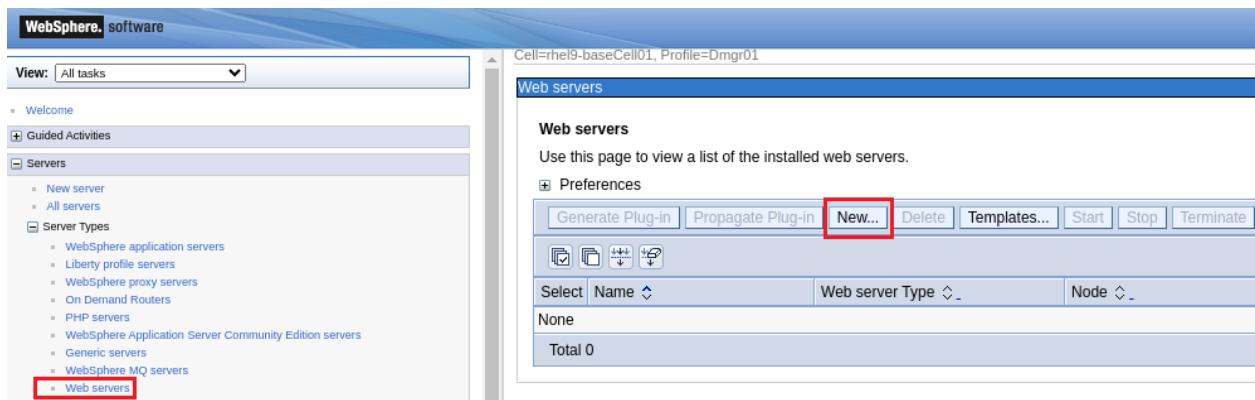
```

- k. The application is now accessible at port 9082.



11. Create a web server definition to explain the load balancing

- a. Click on **Servers > Server Types > Web servers** to open the Web servers panel.  
Then click **New** to create a new web server definition.



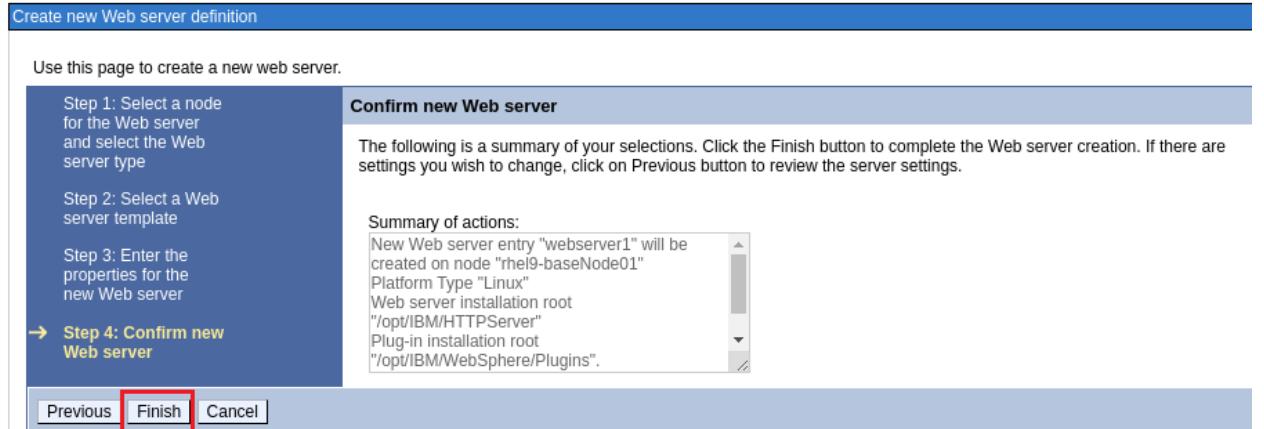
- b. In Step 1, enter **webserver1** for the Server name and leave the default setting for the rest, then click on **Next**.

The screenshot shows the 'Create new Web server definition' wizard. It has four steps:

- Step 1: Select a node for the Web server and select the Web server type
- Step 2: Select a Web server template
- Step 3: Enter the properties for the new Web server
- Step 4: Confirm new Web server

The 'Select a node for the Web server and select the Web server type' step is active. It includes a dropdown for 'Select node' (set to 'rhe19-baseNode01'), a 'Server name' input field ('\* webserver1') which is highlighted with a red box, and a 'Type' dropdown ('IBM HTTP Server'). At the bottom are 'Next' and 'Cancel' buttons, with 'Next' also highlighted with a red box.

- c. For Step 2 and 3, leave the default setting and click **Next**.
- d. In Step 4, review the settings, then click **Finish**.



- e. Click **Save** to save the changes.

Cell=rhel9-baseCell01, Profile=Dmgr01

**Web servers**

**Messages**

- New server is created successfully.
- Modify variables, resources, and other server configuration settings, such as message broker queue names before running the newly created server.
- ⚠ Changes have been made to your local configuration. You can:
  - [Save](#) directly to the master configuration.
  - [Review](#) changes before saving or discarding.

An option to synchronize the configuration across multiple nodes can be disabled in [Preferences](#).

⚠ The server may need to be restarted for these changes to take effect.

**Web servers**

Use this page to view a list of the installed web servers.

**Preferences**

Actions						
Generate Plug-in	Propagate Plug-in	New...	Delete	Templates...	Start	Stop
Select	Name ↴	Web server Type ↴	Node ↴	Host Name ↴	Version ↴	Status ↴
You can administer the following resources:						
<input type="checkbox"/>	<a href="#">webserver1</a>	IBM HTTP Server	rhel9-baseNode01	rhel9-base.gym.lan	ND 9.0.5.23	
Total 1						

- f. Click **OK** to return to the web server panel. Then click on **webserver1** to see the settings.

Web servers							
Web servers							
Use this page to view a list of the installed web servers.							
<input checked="" type="checkbox"/> Preferences							
<input type="button" value="Generate Plug-in"/>	<input type="button" value="Propagate Plug-in"/>	<input type="button" value="New..."/>	<input type="button" value="Delete"/>	<input type="button" value="Templates..."/>	<input type="button" value="Start"/>	<input type="button" value="Stop"/>	<input type="button" value="Terminate"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Select	Name <input type="button" value="▼"/>	Web server Type <input type="button" value="▼"/>	Node <input type="button" value="▼"/>	Host Name <input type="button" value="▼"/>	Version <input type="button" value="▼"/>	Status <input type="button" value="▼"/>	
You can administer the following resources:							
<input type="checkbox"/>	<a href="#">webserver1</a>	IBM HTTP Server	rhel9-baseNode01	rhel9-base.gym.lan	ND 9.0.5.23		
Total 1							

- g. Click on Plug-in properties to see the details.

Web servers							
<a href="#">Web servers &gt; webserver1</a>							
Use this page to configure a web server that provides HTTP and HTTPS support to application servers.							
<a href="#">Configuration</a>							
<b>General Properties</b> <hr/> Web server name <input type="text" value="webserver1"/> Type <input type="button" value="IBM HTTP Server"/> * Port <input type="text" value="80"/> * Web server installation location <input type="text" value="/opt/IBM/HTTPServer"/> * Configuration file name <input type="text" value="\${WEB_INSTALL_ROOT}/conf/httpd.conf"/> <span style="border: 1px solid #ccc; padding: 2px;">Edit</span>				<b>Configuration settings</b> <hr/> <ul style="list-style-type: none"> <li><a href="#">Web Server Virtual Hosts</a></li> <li><a href="#">Global Directives</a></li> </ul> <b>Additional Properties</b> <hr/> <ul style="list-style-type: none"> <li><a href="#">Log file</a></li> <li><a href="#">Intelligent Management</a></li> <li><a href="#">Configuration File</a></li> <li><a href="#">Process definition</a></li> <li><a href="#">Plug-in properties</a></li> <li><a href="#">Custom properties</a></li> </ul> <input checked="" type="checkbox"/> Ports			
<input type="button" value="Apply"/> <input type="button" value="OK"/> <input type="button" value="Reset"/> <input type="button" value="Cancel"/>							

- h. On the Plug-in properties page, click on **View** to open the plugin configuration file.

[Web servers](#) > [webserver1](#) > Plug-in properties

Use this page to configure a web server plug-in. The plug-in passes HTTP requests from a web server to WebSphere(R) application servers.

Runtime

Configuration

Plug-in properties

Ignore DNS failures during Web server startup

\* Refresh configuration interval

60 seconds

Repository copy of Web server plug-in files:

\* Plug-in configuration file name

plugin-cfg.xml

[View](#)

Automatically generate the plug-in configuration file

Automatically propagate plug-in configuration file

\* Plug-in key store file name

plugin-key.kdb

[Manage keys and certificates](#)

[Copy to Web server key store directory](#)

Additional Properties

- = [Request and Response](#)
- = [Caching](#)
- = [Request Routing](#)
- = [Custom Properties](#)

- i. The plugin configuration contains next to the routing information for the WAS traditional server1 also the routing information for the Liberty server.

#### Plug-in configuration file

```
<?xml version="1.0" encoding="ISO-8859-1"?><!--HTTP server plugin config file for the webserver rhel9-baseCell01.rhel9-b
<Config ASDisableNagle="false" AcceptAllContent="true" AppServerPortPreference="HostHeader" ChunkedResponse="false" FIPS=
  <Log LogLevel="Error" Name="/opt/IBM/WebSphere/Plugins/logs/webserver1/http_plugin.log"/>
  <Property Name="ESIEnable" Value="false"/>
  <Property Name="ESIMaxCacheSize" Value="1024"/>
  <Property Name="ESIInvalidationMonitor" Value="false"/>
  <Property Name="ESIEnableToPassCookies" Value="false"/>
  <Property Name="ESICacheidFull" Value="false"/>
  <Property Name="PostSizeLimit" Value="-1"/>
  <Property Name="PostBufferSize" Value="0"/>
  <Property Name="PluginInstallRoot" Value="/opt/IBM/WebSphere/Plugins//>
  <Property Name="Keyfile" Value="/opt/IBM/WebSphere/Plugins/config/webserver1/plugin-key.kdb"/>
  <Property Name="Stashfile" Value="/opt/IBM/WebSphere/Plugins/config/webserver1/plugin-key.sth"/>
<VirtualHostGroup Name="default_host">
  <VirtualHost Name="*:9080"/>
  <VirtualHost Name="*:80"/>
  <VirtualHost Name="*:9443"/>
  <VirtualHost Name="*:5060"/>
  <VirtualHost Name="*:5061"/>
  <VirtualHost Name="*:443"/>
</VirtualHostGroup>
<ServerCluster CloneSeparatorChange="false" GetDWLMTTable="true" IgnoreAffinityRequests="true" LoadBalance="RoundRobin">
  <Server ConnectTimeout="5" ExtendedHandshake="false" MaxConnections="-1" Name="rhel9-baseNode01_server1" ServerIOT="1">
    <Transport ConnectionTTL="28" Hostname="rhel9-base.gym.lan" Port="9080" Protocol="http"/>
    <Transport ConnectionTTL="28" Hostname="rhel9-base.gym.lan" HostnameAlias="rhel9-base.gym.lan" Port="9443" Protocol="https">
      <Property Name="keyring" Value="/opt/IBM/WebSphere/Plugins/config/webserver1/plugin-key.kdb"/>
      <Property Name="stashfile" Value="/opt/IBM/WebSphere/Plugins/config/webserver1/plugin-key.sth"/>
    </Transport>
  </Server>
</ServerCluster>
<ServerCluster CloneSeparatorChange="false" GetDWLMTTable="true" IgnoreAffinityRequests="true" LoadBalance="RoundRobin">
  <Server ConnectTimeout="5" ExtendedHandshake="false" MaxConnections="-1" Name="rhel9-baseNode01_LibertyServer2" ServerIOT="2">
    <Transport ConnectionTTL="28" Hostname="rhel9-base.gym.lan" Port="9082" Protocol="http"/>
    <Transport ConnectionTTL="28" Hostname="rhel9-base.gym.lan" HostnameAlias="rhel9-base.gym.lan" Port="9444" Protocol="https">
      <Property Name="keyring" Value="/opt/IBM/WebSphere/Plugins/config/webserver1/plugin-key.kdb"/>
      <Property Name="stashfile" Value="/opt/IBM/WebSphere/Plugins/config/webserver1/plugin-key.sth"/>
    </Transport>
  </Server>
</ServerCluster>
<UriGroup Name="default host server1 rhel9-baseNode01 Cluster URIs">
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/snoop/*"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/hitcount"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="*.jsp"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="*.jsv"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="*.jsw"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/j_security_check"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/ibm_security_logout"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/servlet/*"/>
  <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/resorts/*"/>
</UriGroup>
<Route ServerCluster="server1_rhel9_baseNode01_Cluster" UriGroup="default host server1 rhel9-baseNode01_Cluster_URIs">
  <UriGroup Name="default host LibertyServer2 rhel9-baseNode01 Cluster URIs">
    <Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/resorts/*"/>
  </UriGroup>
</Route>
```

- j. By adding the Liberty port 9082 to virtual hosts and installing an HTTP Server with the updated plug-in, the created Liberty instance becomes part of the load balancing without any additional effort like plug-in merge or so.

Optional lab:

### **Use Application Modernization Accelerator to analyze applications**

#### **Step 1: Install Application Modernization Accelerator.**

1. Open a terminal window in the VM.
2. Download and extract the Application Accelerator package

```
mkdir ~/ama42
cd ~/ama42
wget https://public.dhe.ibm.com/ibmdl/export/pub/software/websphere/ta/application-modernization-accelerator-local-4.2.0.zip
unzip application-modernization-accelerator-local-4.2.0.zip
cd application-modernization-accelerator-local-4.2.0/
```

3. Execute the following command to start the installation

```
./launch.sh
```

In the wizard, perform the following steps:

- Enter **1** for 1) IBM Application Modernization Accelerator 4.2 (Evaluation)
- Enter **1** to accept license agreement
- Enter **1** to install AMA

Finally you should see something like

**Application Modernization Accelerator 4.2.0 is available for use  
at the following URL> <https://rhel9-base.gym.lan:443>**

4. Access AMA from a browser via URL <https://rhel9-base.gym.lan:443>.

#### **Step 2: Create a workspace with sample content**

In AMA, click von **Create workspace** to create a workspace.

IBM Application Modernization Accelerator

Workspaces ⓘ

Search workspaces

Create workspace +

Enter as workspace name **samples** and select to include **sample data**, then click **Create**.

Create a new workspace

Name your workspace

Workspace name

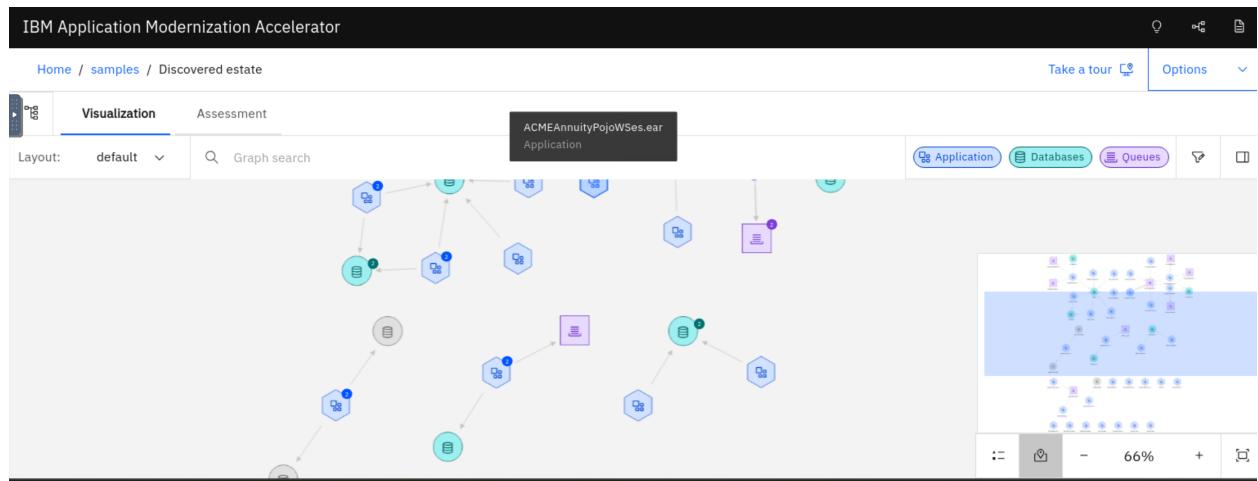
Include sample data

Yes

Cancel Create

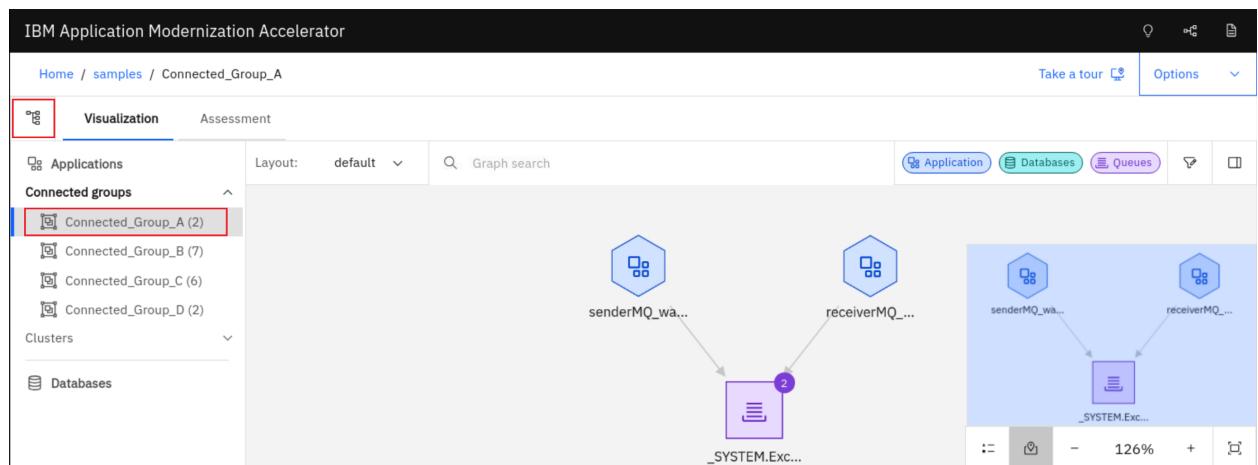
Once the workspace has been created and the sample data has been imported and analyzed, you should get a visualization view.

### Step 3: Analyze the applications



Connection groups help to see which applications are connected to each other.

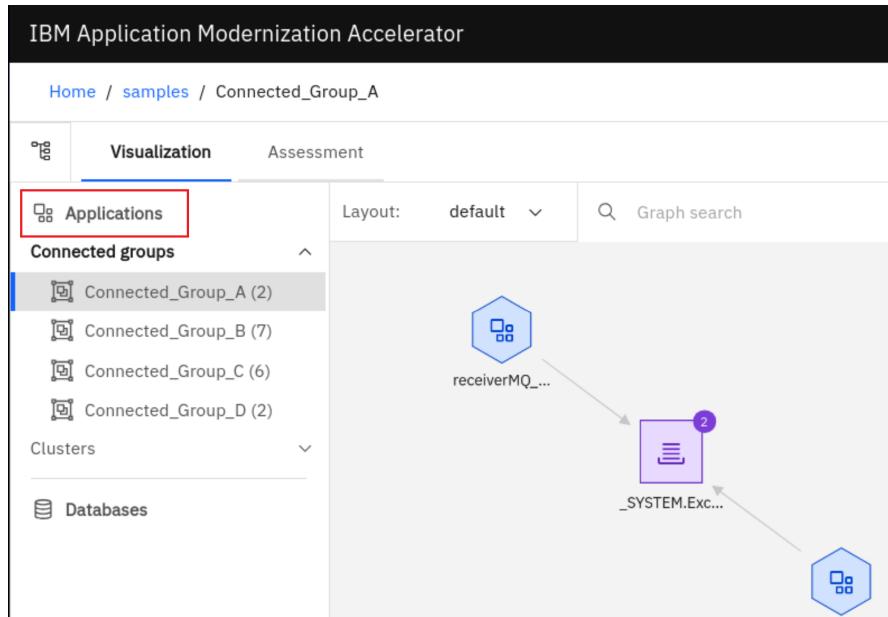
Open the menu on the left and select **Connected\_Group\_A**.



You can see that two applications are connected to each other via queue.

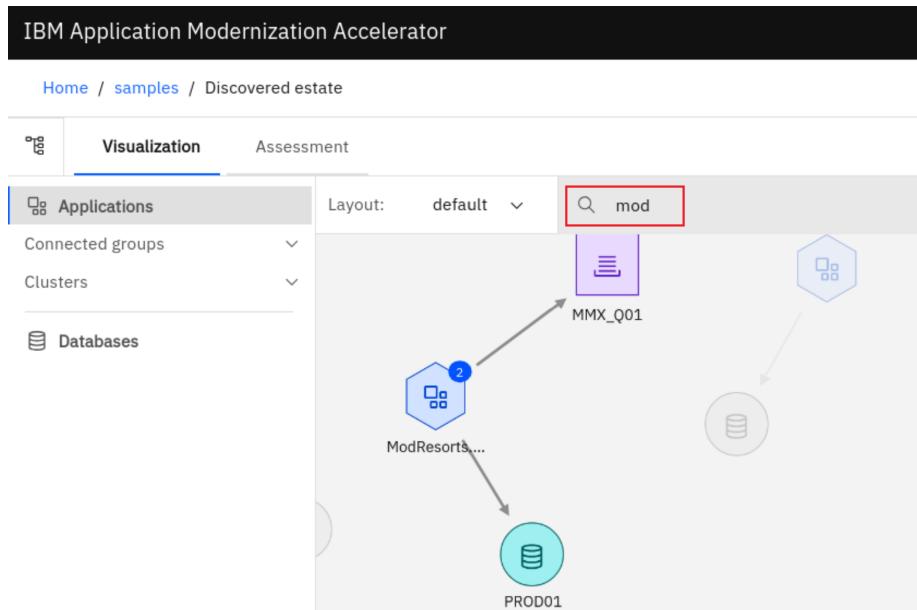
This insight might help you if you want to move one of the applications including messaging system into the cloud or into containers.

Switch back to the overall view by clicking on **Applications**.



Now let's take a look at the modresorts application.

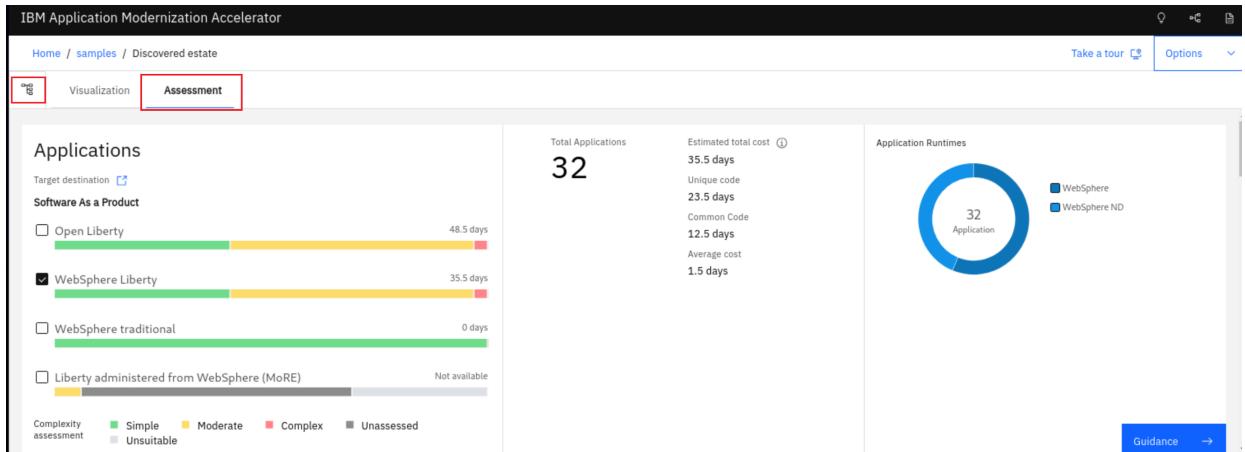
Type **mod** into the search bar to find the modresorts application



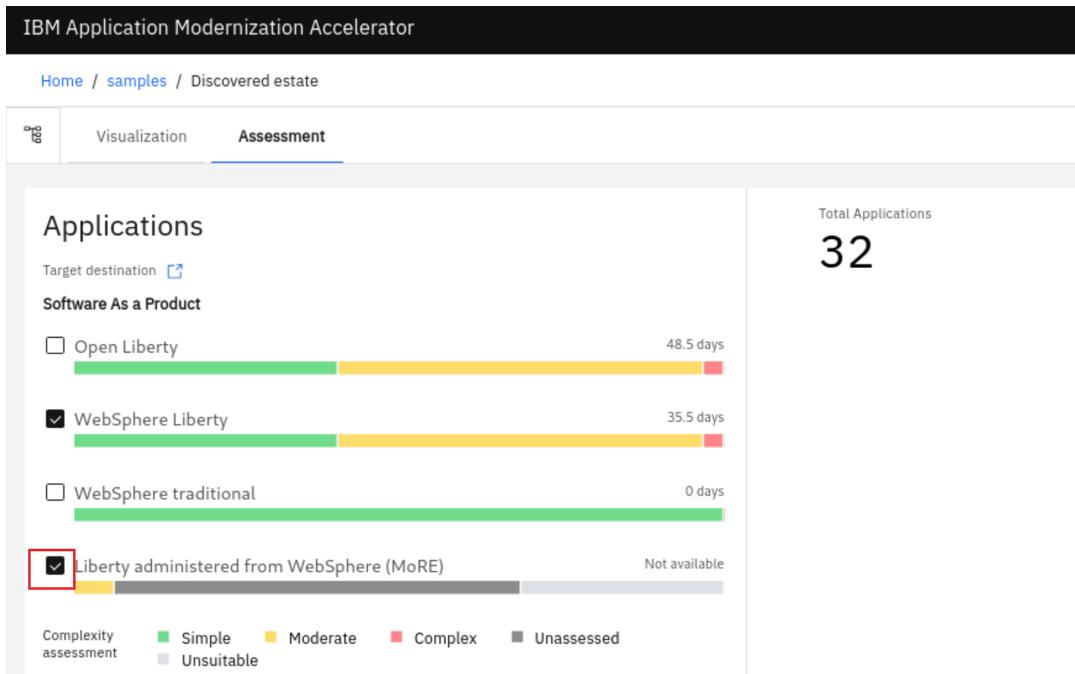
As you can see, the application has a dependency on a database and a queue.

Click on **Assessment** to see the efforts to migrate to Liberty or MoRE.

Then close the tree with the connected groups.



Select **Liberty administered from WebSphere (MoRE)** to see the assessment for target MoRE.



Scroll down to see the applications that could be migrated to MoRE.

To filter, enter **modr** into the search bar.

Java application	Complexity	Issues	Code changes	Development cost	Tags	Connections	View plan
moderste-1_0_war.ear Liberty administered from WebSphere (MoRE)	Moderate	1 0 4	Part-Automated	1 day	<a href="#">+ Tag</a>	0	<a href="#">→</a>
moderste-1_0_war.ear WebSphere Liberty	Simple	0 0 1	None	0 days	<a href="#">+ Tag</a>	0	<a href="#">→</a>
ModResorts.ear Liberty administered from WebSphere (MoRE)	Unassessed	0 0 0	None	0 days	<a href="#">+ Tag</a>	2	<a href="#">→</a>
ModResorts.ear WebSphere Liberty	Moderate	2 0 5	Automated	0.5 days	<a href="#">+ Tag</a>	2	<a href="#">→</a>

As you can see, the sample collection contains two modresorts applications.

Hover over the one in grey and you will get the information that the application has not been assessed for MoRE.

Java application	Complexity	Issues	Code changes	Development cost	Tags	Connections	View plan
moderste-1_0_war.ear Liberty administered from WebSphere (MoRE)	Moderate	1 0 4	Part-Automated	1 day	<a href="#">+ Tag</a>	0	<a href="#">→</a>
moderste-1_0_war.ear WebSphere Liberty	Simple	0 0 1	None	0 days	<a href="#">+ Tag</a>	0	<a href="#">→</a>
ModResorts.ear Liberty administered from WebSphere (MoRE)	Unassessed	0 0 0	None	0 days	<a href="#">+ Tag</a>	2	<a href="#">→</a>
ModResorts.ear WebSphere Liberty	Moderate	2 0 5	Automated	0.5 days	<a href="#">+ Tag</a>	2	<a href="#">→</a>

This means that the data collection was created with an older version of the data collector tool.

Click on the modresorts application that has been assessed for target MoRE.

Java application	Complexity	Issues	Code changes	Development cost	Tags	Connections	View plan
moderste-1_0_war.ear Liberty administered from WebSphere (MoRE)	Moderate	1 0 4	Part-Automated	1 day	<a href="#">+ Tag</a>	0	<a href="#">→</a>
moderste-1_0_war.ear WebSphere Liberty	Simple	0 0 1	None	0 days	<a href="#">+ Tag</a>	0	<a href="#">→</a>
ModResorts.ear Liberty administered from WebSphere (MoRE)	Unassessed	0 0 0	None	0 days	<a href="#">+ Tag</a>	2	<a href="#">→</a>
ModResorts.ear WebSphere Liberty	Moderate	2 0 5	Automated	0.5 days	<a href="#">+ Tag</a>	2	<a href="#">→</a>

On the application details page, scroll down to the section with unique code issues.

The screenshot shows the 'Unique Code Issues' section for the 'modresorts-1\_0\_war.ear' application. The section is titled 'Technology issues' and contains the following items:

Issue Description	Severity	Count	Impact
Update your application to use the new Jakarta EE package name	Red	1	0.5 days
Java SE 11 general information and potential issues	Blue	3	0.1 days
Java SE 17 general information and potential issues	Blue	3	0.1 days
Additional information	Grey	0	0 days

As you can see, there is some manual effort required to migrate the application to Jakarta EE 10. The changes to migrate from Java 8 to Java 17 can be automated.

Click on **View migration plan** to see the migration plan.

The screenshot shows the same application details page as before, but with the 'View migration plan' button highlighted by a red box. This button is located in the top right corner of the 'Unique Code Issues' section.

## Download migration plan

Your migration plan helps you develop and test any application changes required for your target runtime

### Application migration overview

Review your application migration overview

Application name	Workspace
modresorts-1_0_war.ear	samples
Source environment	Migration target
IBM WebSphere Application Server	Liberty administered from WebSphere (MoRE)

Common code files  
0

### Preview Files

Preview the files included in your download. These files may reference other files only available when you download the migration bundle.

[server.xml](#)

On the Migration plan page, ou can see that a server.xml file has been created. This will help the developer to test the application locally in his development environment based on Liberty before deploying it to MoRE.

Click on **server.xml** to open the file.

File preview

### server.xml

Contains the Liberty server configuration for the application(s) you are migrating. It configures application dependencies such as database connections and messaging. The server.xml may need some updates - for example, adding passwords that have been removed during automatic data cleansing. For multi-application migrations the included server.xml files are available in the bundle.

[Learn more.](#)

```

0   <?xml version="1.0" encoding="UTF-8" standalone="no"?>
1   <!--Generated by IBM Migration Application Bundler 0.0.29-SNAPSHOT
2   Tue May 20 12:26:47 UTC 2025--><server description="Configuration generated by binaryAppScanner">
3       <featureManager>
4           <!--The following features are available in all editions of Liberty.-->
5               <feature>appSecurity-5.0</feature>
6               <feature>cdi-4.0</feature>
7               <feature>federatedRegistry-1.0</feature>
8               <feature>jdbc-4.3</feature>
9               <feature>jndi-1.0</feature>
10              <feature>jsonb-3.0</feature>
11              <feature>jsonp-2.1</feature>
12              <feature>ldapRegistry-3.0</feature>
13              <feature>restfulWS-3.1</feature>
14              <feature>restfulWSClient-3.1</feature>

```

Show more

Cleanup:

Click on **Home** to switch back to the main screen.

IBM Application Modernization Accelerator

[Home](#) / [samples](#) / [modresorts-1\\_0\\_war.ear](#) / Migration plan

**This concludes the lab.**

Optional lab:

**Use Application Modernization Accelerator to create a data collection.**

### Step 1: Create a workspace without sample content

In AMA, click von **Create workspace** to create a workspace.

The screenshot shows the IBM Application Modernization Accelerator (AMA) interface. At the top, it says "IBM Application Modernization Accelerator". Below the header is a network graph with a central teal circle connected to several blue hexagons and purple squares. Three specific nodes are highlighted with numbered boxes: 1 points to a blue hexagon at the bottom left, 2 points to a purple square in the middle left, and 3 points to a blue hexagon in the middle right. The main content area is titled "Workspaces" with an information icon. It includes a search bar labeled "Search workspaces" and a blue button labeled "Create workspace +".

Enter as workspace name **wasnd** but do not select include **sample data**, then click **Create**.

A modal dialog box titled "Create a new workspace" is shown. It has a close button "X" in the top right corner. The title bar says "Create a new workspace". The main area is titled "Name your workspace". Below it is a "Workspace name" field containing "wasnd". Underneath is a "Include sample data" section with a toggle switch set to "No". At the bottom are two buttons: "Cancel" on the left and a large blue "Create" button on the right.

## Step 2: Download the data collector

Click on **Open Discovery Tool**

Upload information about your estate.

After the discovery tool is downloaded, run it to collect information about your estate (applications, databases, and queues).

If you already ran the discovery tool, upload your results to start planning your migration journey.

[Open discovery tool](#) [Upload results](#)

[Documentation](#)

Keep the OS to Linux and select to download the discovery tool.

IBM Application Modernization Accelerator

Home / wasnd / Discovery tool

## Discovery tool

The Discovery tool gathers information about the deployments in your environment to help Application Modernization Accelerator access your estate and determine your migration readiness.

Complete these steps for each operating system where your organization runs applications and databases.

### Download the Discovery tool

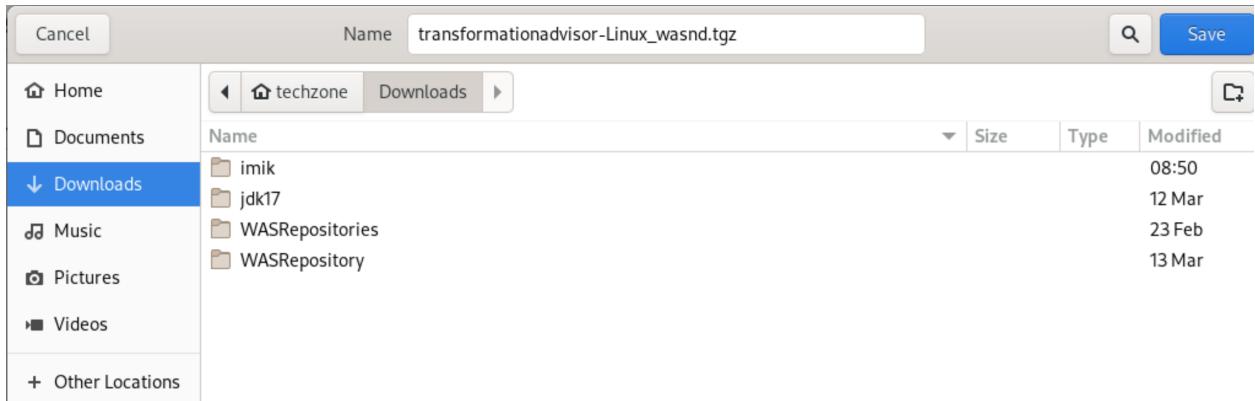
To download the Discovery tool, specify the operating system where your applications or databases are running.

Source Operating System

Linux [Download discovery tool](#)

For instructions, see the [documentation](#)

Make sure that **Downloads** is selected as target and click on **Save**.



### Step 3: Extract the data collector

1. Open a terminal window in the VM.
2. Download and extract the Application Accelerator package

```
mkdir ~/ama42collector  
cd ~/ama42collector  
tar -zxf ~/Downloads/transformationadvisor-Linux_wasnd.tgz  
cd transformationadvisor-4.2.0/
```

3. Execute the following command to see the options.

```
bin/transformationadvisor --help
```

Press **1** to accept the license agreement.

4. Execute the following command to create the collection.

```
bin/transformationadvisor -w /home/techzone/IBM/WebSphere/AppServer
```

A panel will open to show that one profile and 2 applications have been found. Wait until the analysis has been done.

Finally you will get a message that the data collection was generated and uploaded.

```
techzone@rhel9-base:~/ama42collector/transformationadvisor-4.2.0
```

```
=====
| Status: Finished
+-----+
| Configuration analysis: Completed
+-----+
|                               Profile
| Currently processing: 1/1
| Profile name: Dmgr01
+-----+
|                               Applications
| Total: 2
| Completed: 2
+-----+
|                               Time
| Elapsed time: 00:00:32
| Time remaining: 00:00:00
+-----+
|                               Progress
| >>>>>>>>>>>>>>>>>>>>>>>>>>>>>>> 100%
+-----+
| Current Operation:
| Done collecting transformation advisor data.
|

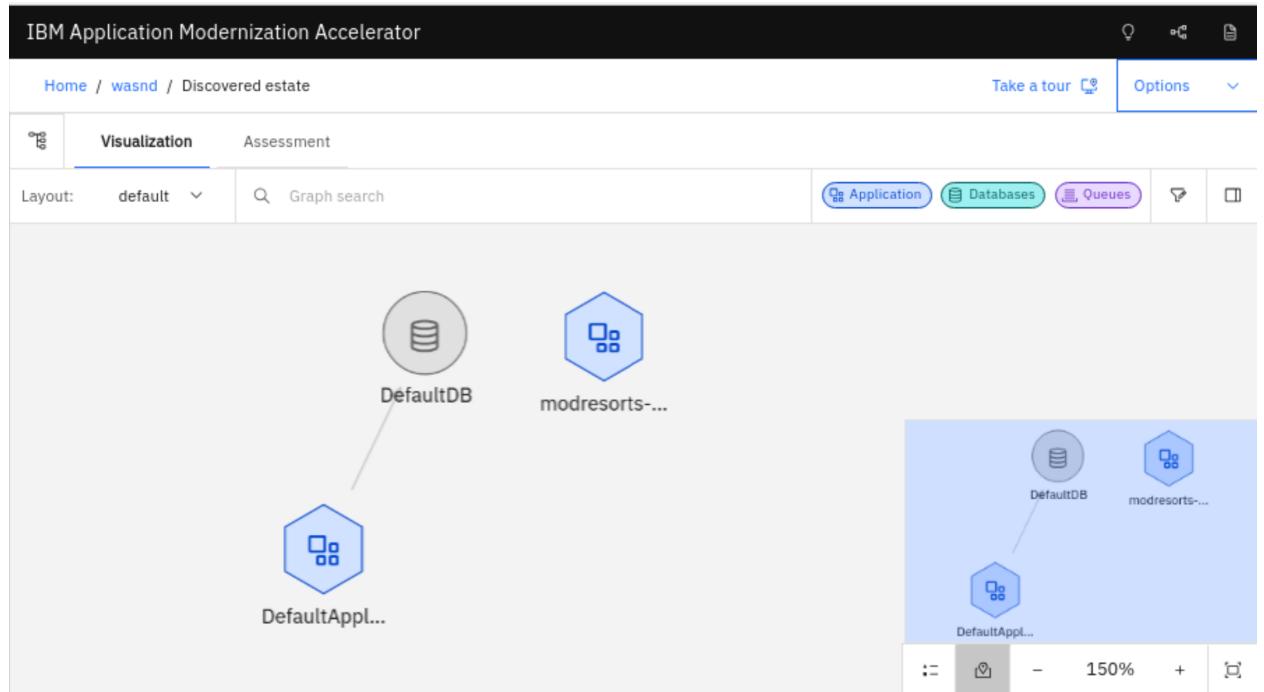


=====
The data collection was generated to the following file: /home/techzone/ama42collector/transformationadvisor-4.2.0/Dmgr01.zip
Here is the response from the Transformation Advisor server: {"message":"Bulk upload started. Track the status at: https://rhel9-base.gym.lan:2220/lands_advisor/advisor/v2/collectionArchives/bulkImport/status/2451117809861640 "}
[techzone@rhel9-base transformationadvisor-4.2.0]$
```

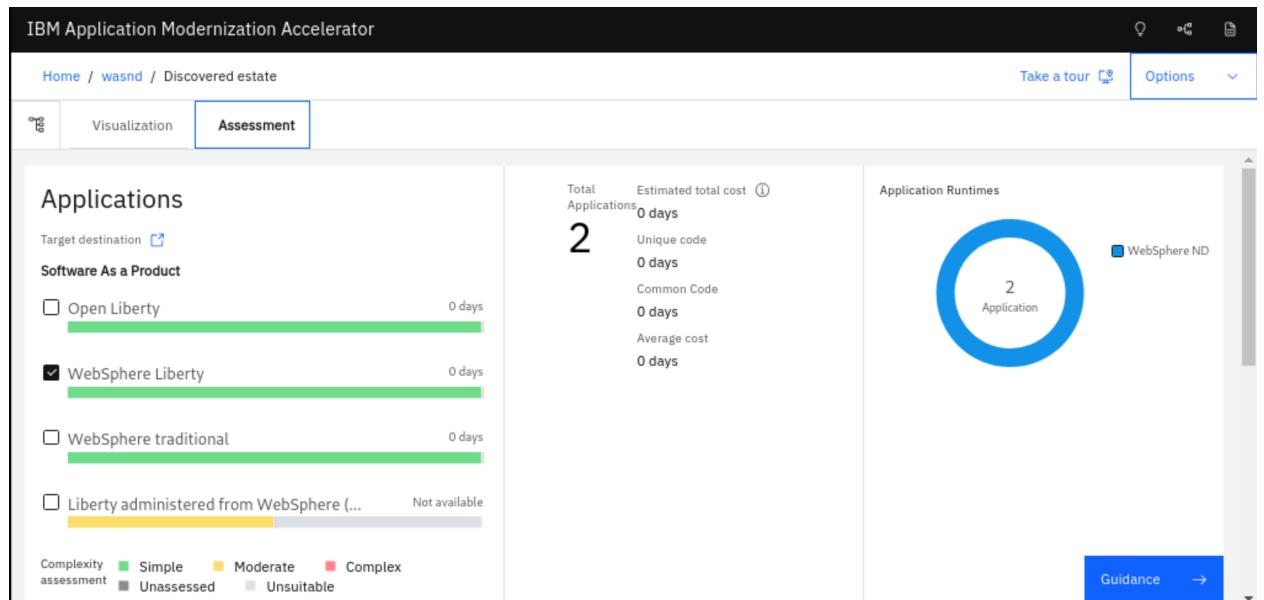
#### Step 4: View the collection in AMA

Switch back to the browser window with AMA.

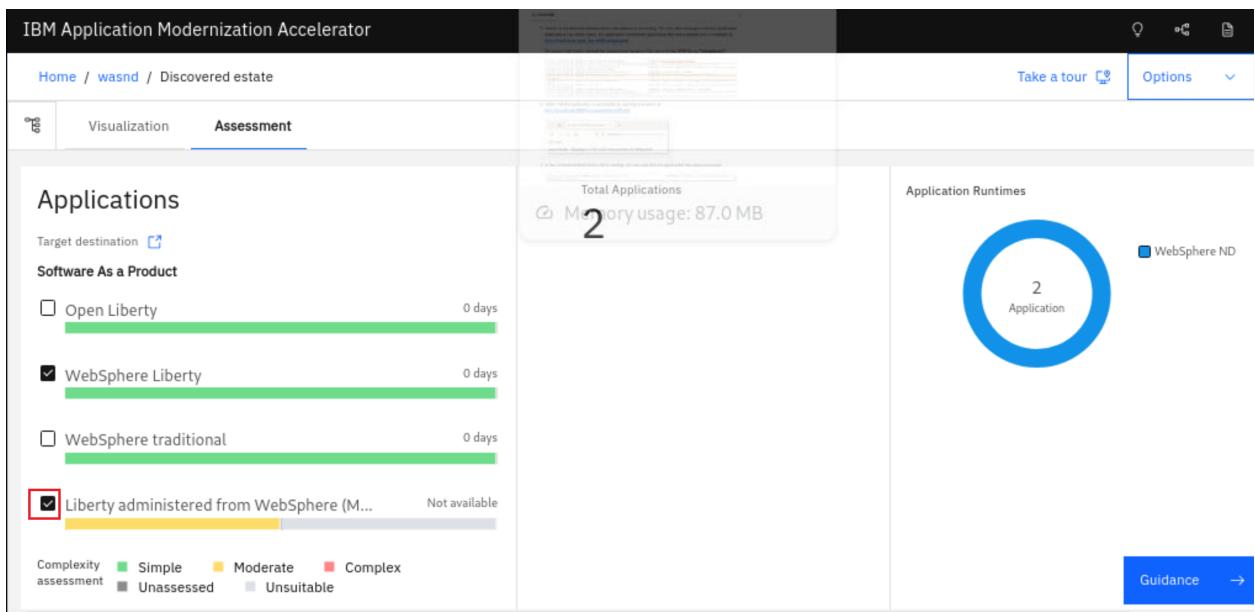
In the wasnd workspace, you should find the visualization of the two applications DefaultApplication and modresorts.



Switch to **Assessment** to see the assessment results.

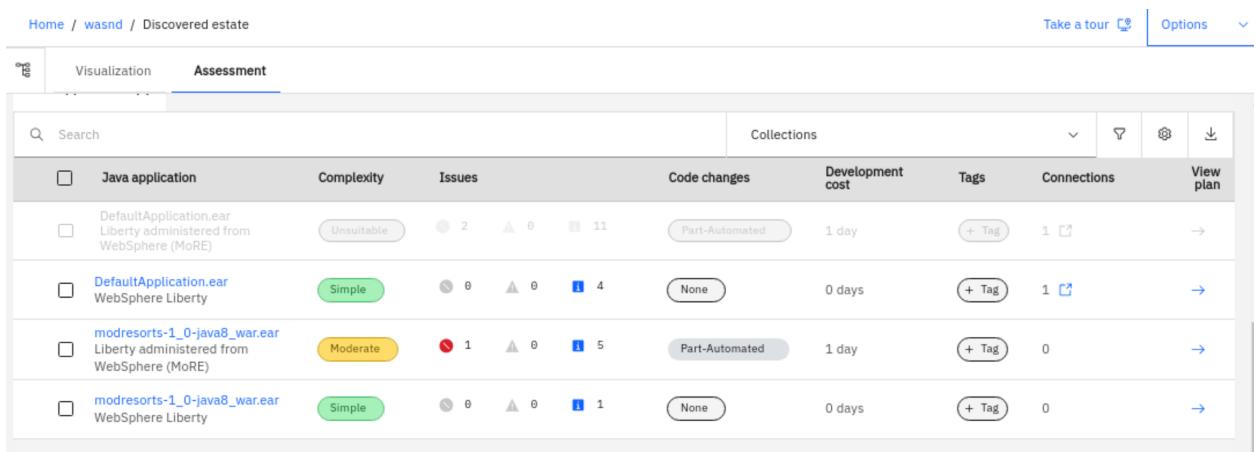


Select **Liberty administered from WebSphere (MoRE)** to see the assessment for target MoRE.



The screenshot shows the 'Assessment' tab of the IBM Application Modernization Accelerator. The left panel displays a list of applications under 'Software As a Product': Open Liberty, WebSphere Liberty, WebSphere traditional, and Liberty administered from WebSphere (MoRE). The 'WebSphere Liberty' row is selected. The right panel shows a summary: 'Total Applications' (2), 'Memory usage: 87.0 MB', and 'Application Runtimes' (2 Application, WebSphere ND). A legend at the bottom defines complexity levels: Simple (green), Moderate (yellow), Complex (red), Unassessed (grey), and Unsuitable (light grey).

Scroll down to see for each application the efforts by modernization target.



The screenshot shows the 'Assessment' tab of the IBM Application Modernization Accelerator. The table lists Java applications with their details: DefaultApplication.ear (Liberty administered from WebSphere (MoRE)), modresorts-1\_0-java8\_war.ear (Liberty administered from WebSphere (MoRE)), and another entry for modresorts-1\_0-java8\_war.ear (WebSphere Liberty). Each row includes columns for Complexity, Issues, Code changes, Development cost, Tags, Connections, and a 'View plan' button.

Feel free to investigate further into the applications.

**This concludes the lab.**