

AM0200ST - Modernize a Java Application for Container and OpenShift with Transformation Advisor



Contents

MODERNIZE A JAVA APPLICATION FOR CONTAINER AND OPENSHIFT WITH TRANSFORMATION ADVISOR.....	3
1. BUSINESS SCENARIO	3
2. OBJECTIVE.....	4
3. PREREQUISITES.....	4
4. WHAT IS ALREADY COMPLETED	5
5. LAB TASKS.....	6
6. EXECUTE LAB TASKS	6
7. SUMMARY	28
APPENDIX A. NOTICES	29
APPENDIX B. TRADEMARKS AND COPYRIGHTS	31

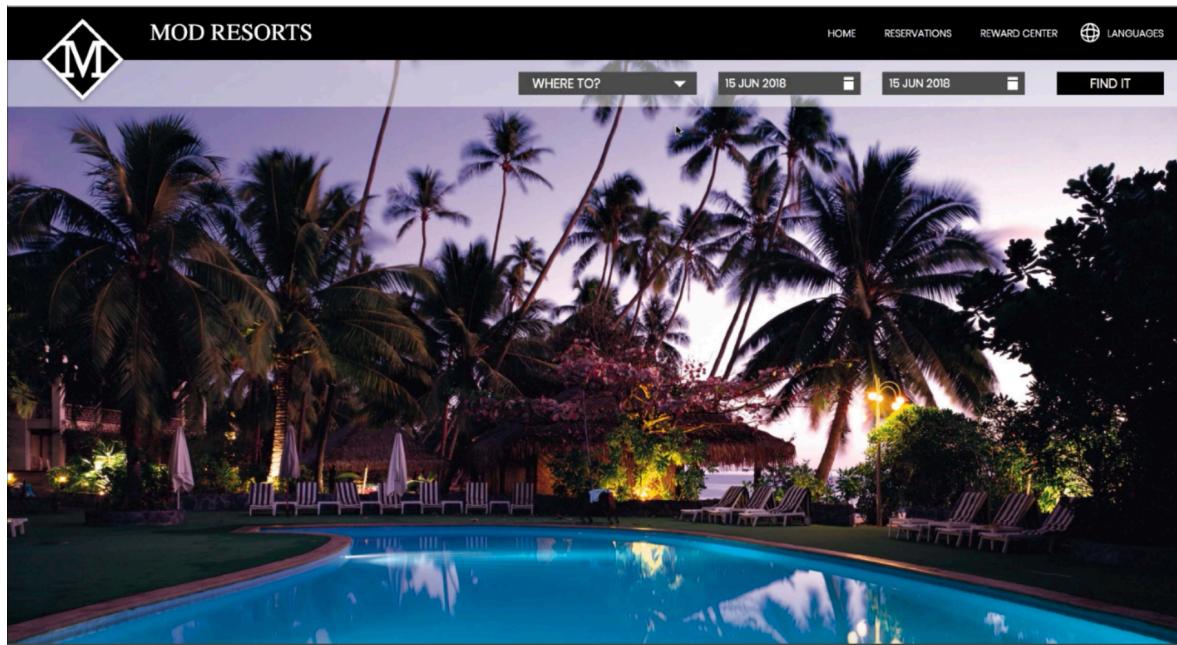
Modernize a Java Application for Container and OpenShift with Transformation Advisor

On the journey to cloud, enterprise customers are facing challenges moving their existing on-premises applications to cloud quickly and cost-effectively. The **IBM Cloud Pak for Applications (CP4Apps)** provides a complete and consistent experience and solution to modernize enterprise applications for cloud-native deployments. Customers can easily modernize their existing applications with IBM's integrated tools and develop new cloud-native applications faster for deployment on any cloud. One of the tools included in the Pak is the **IBM Cloud Transformation Advisor** (Transformation Advisor), a developer tool that is available at no charge to help you quickly evaluate on-premises Java EE applications and to prepare the apps for a rapid cloud deployment.

This lab exercise is a part of the Application Modernization lab series which focus on evaluation, re-platforming, and rehosting modernization approaches and solutions. In Part 1 of the Application Modernization Journey (Lab **AM0100ST**), you go through the process to evaluate the existing On-Prem Java applications and to identify the candidate to be moved to the cloud using the IBM Cloud Transformation Advisor. This lab showcases the re-platforming process. Re-platform uses Lift, Modify and Shift approach to move an existing application to the cloud. You learn how to move a selected candidate Java application to Liberty container and deploy it to **Red Hat OpenShift Container Platform (RHOCP)** cluster using the **Transformation Advisor Migration Bundle** tool. The tool identifies the Java EE programming models in an Java application, helps to determine its complexity by listing a high-level inventory of the content and structure of the application, highlights Java EE programming models and WebSphere API differences between the WebSphere profile type and shows any Java EE specification implementation differences that could affect the application. You can use this tool in your move in the cloud journey to quickly analyze on-premise WebSphere applications without accessing source code and to prepare the applications for a rapid cloud deployment.

1. Business Scenario

As shown in the image below, your company has a web application called **Mod Resorts**, a WebSphere application showing the weather in various locations. Your company wants to move this application from on-premises to the cloud.



As a tech lead, you have already analyzed the application using the **Transformation Advisor** tool in Part 1 of the lab series. Based on the analysis you know that you can move this application from the traditional WebSphere Server environment to a light-weighted Liberty server environment without any code change. Now you are planning to use the **Transformation Advisor** migration plan to create the migration bundle, to containerize the application, and to deploy the Docker container to an OpenShift Kubernetes cluster environment.

2. Objective

The objectives of this lab are to:

- Learn how Transformation Advisor can accelerate application migration to cloud process, minimize errors and risks and reduce time to market
- Learn how to deploy the application to an OpenShift cluster environment using the migration bundle created by Transformation Advisor

3. Prerequisites

The following prerequisites must be completed prior to beginning this lab:

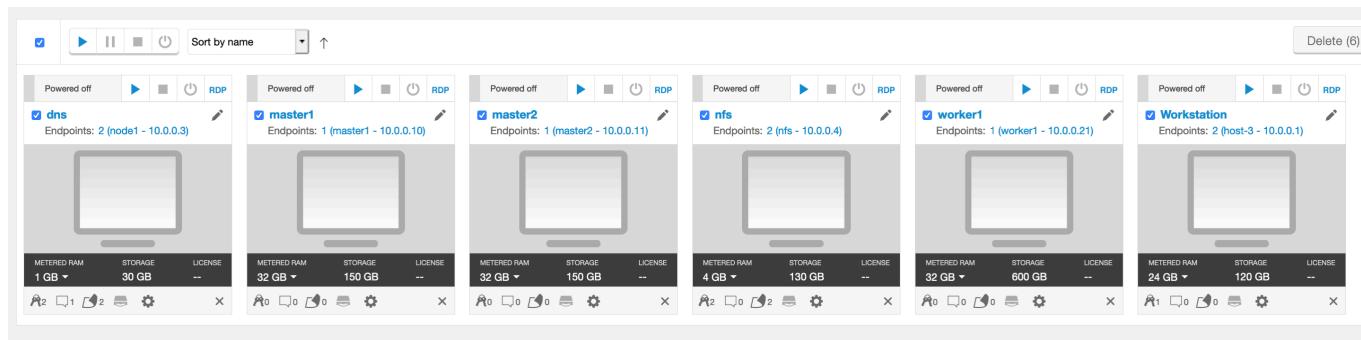
- Familiarity with basic Linux commands
- Have Internet Access

The following symbols appear in this document at places where additional guidance is available.

Icon	Purpose	Explanation
	Important!	This symbol calls attention to a particular step or command. For example, it might alert you to type a command carefully because it is case sensitive.
	Information	This symbol indicates information that might not be necessary to complete a step, but is helpful or good to know.
	Trouble-shooting	This symbol indicates that you can fix a specific problem by completing the associated troubleshooting information.

4. What is Already Completed

Six Linux VMs have been provided for this lab.



- The Red Hat OpenShift Container Platform (**RHOCP**) v4.3, is installed in 4 VMs, the **master1** VM, the **master2** VM, the **dns** VM, and the **worker2** VM, with 2 master nodes and 3 compute nodes (the master nodes are serving as computer nodes as well).
- The **IBM Cloud Pak for Applications (CP4Apps)** v4.1 is installed on the **RHOCP**.
- The **Transformation Advisor** has been deployed to **RHOCP** cluster as a part of **IBM Cloud Pak for Applications** installation. For information on how to install IBM Cloud Pak for Applications on OpenShift, please visit:
https://www.ibm.com/support/knowledgecenter/SSCSJL_4.x/welcome.html
- The **Workstation** VM is the one you use to access and work with the **RHOCP** cluster in this lab. The login credentials for the **Workstation** VM are:

User ID: **ibmdemo**

Password: **passw0rd**

Note: Use the Password above in the **Workstation** VM Terminal for **sudo** in the Lab.

- The CLI commands used in this lab are listed in the **Commands.txt** file located at the **/home/ibmdemo/cp4a-labs/am0200st** directory of the **Workstation** VM for you to copy and paste these commands to the Terminal window during the lab.

5. Lab Tasks

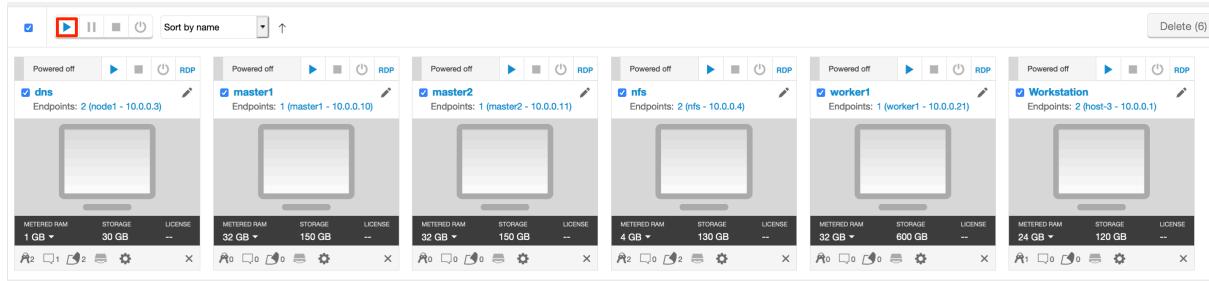
During this lab, you complete the following tasks:

- Review the on-premises WebSphere application
- Upload Application Data into Transformation Advisor
- Create Migration Bundle
- Update the Migration Bundle
- Testing the Migration Bundle Locally (Optional)
- Containerize Liberty Application
- Deploy the Application to OpenShift

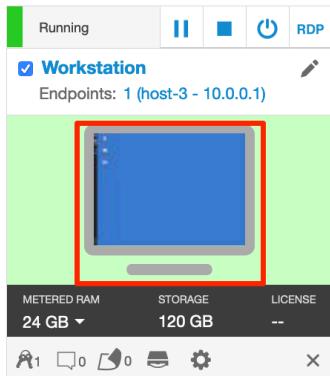
6. Execute Lab Tasks

6.1 Log in to the **Workstation** VM and get started

1. Launch the lab VMs by clicking the **play** button.



2. After the VMs are started, click the **Workstation** VM icon to access it.



The **Workstation Linux Desktop** is displayed. You execute all the lab tasks on this VM.

6.2 Review the on-premises WebSphere application

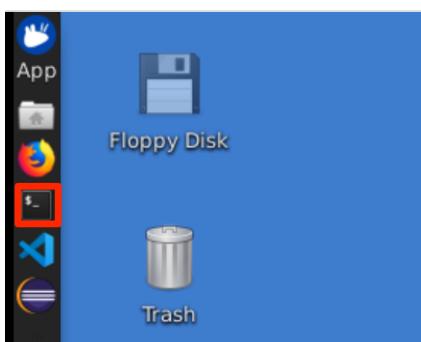
In this task, you take a look at **Mod Resorts** application deployed to the local WebSphere Application Server (WAS) environment. You are going to move this application to the cloud using Open Liberty Operator later.

1. Start WebSphere Application Server

In the **workstation** VM, you have a local WebSphere Application Server V9 which hosts the **Mod Resorts** application.

To start the WAS server:

a. Open a terminal window by clicking its icon on the **Workstation** VM desktop tool bar.



b. In the terminal window, issue the command below to start the WAS server (You can copy and paste the command from the **Commands.txt** file in the **/home/ibmdemo/cp4a-labs/am0200st** directory).

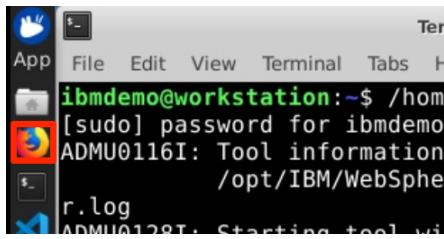
/home/ibmdemo/cp4a-labs/shared/startWAS.sh

when prompted, enter the **sudo** user password as: **passw0rd**.

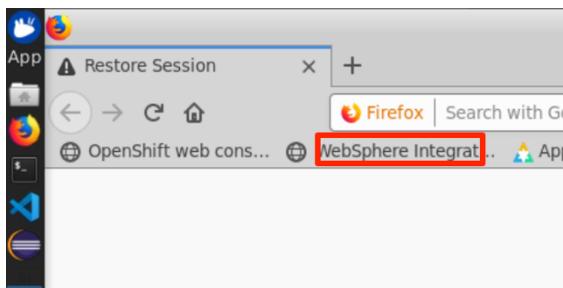
```
ibmdemo@workstation:~$ /home/ibmdemo/cp4a-labs/shared/startWAS.sh
[sudo] password for ibmdemo:
ADMU0116I: Tool information is being logged in file
          /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/logs/server1/startServer.log
ADMU0128I: Starting tool with the AppSrv01 profile
ADMU3100I: Reading configuration for server: server1
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process id is 4509
db2-pbw
ibmdemo@workstation:~$
```

Within a few minutes the WAS server is ready.

- _c. Access the WAS Admin Console to view the application deployed by clicking the web browser icon desktop tool bar to open a browser window.

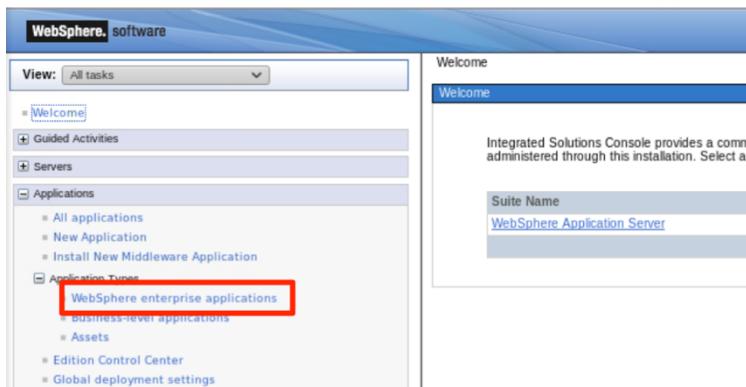


- _d. From the web browser window and click **WebSphere Integrated Solution Console** bookmark to launch the WAS console.



- _e. In the WAS Admin Console login page, enter the User ID and Password as: **wsadmin/passw0rd** and click **Login**.

- _f. On the WAS Console page, click **Applications -> Application Types -> WebSphere enterprise applications** to view the apps deployed.



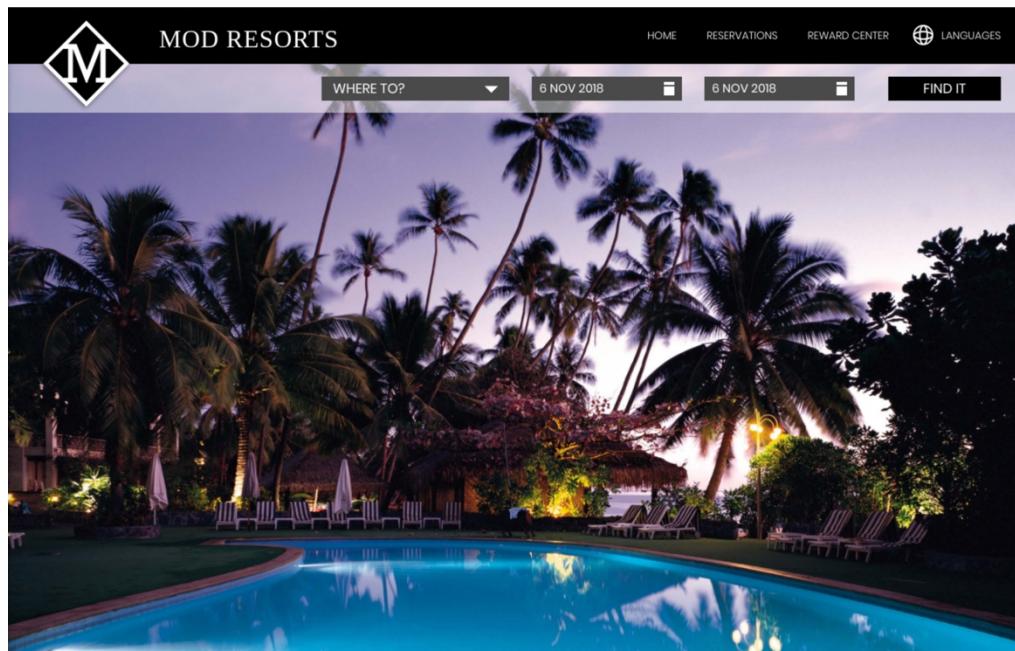
In the **Enterprise Applications** list, you can see all applications deployed. The **Mod Resorts** application is in the list, currently it is running.

Select	Name	Application Status
<input type="checkbox"/>	DefaultApplication	
<input type="checkbox"/>	PlantsByWebSphereEE6	
<input type="checkbox"/>	ivtApp	
<input type="checkbox"/>	modresorts-1_0.war	
<input type="checkbox"/>	query	

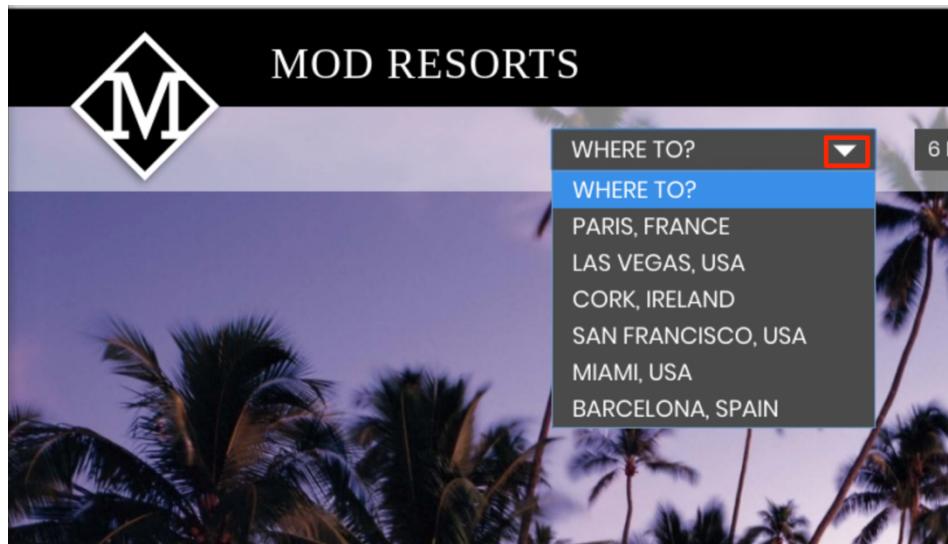
2. View the **Mod Resorts** application

- a. From the web browser window, click new Tab to open a new browser window. Type the **Mod Resorts** application URL: <http://localhost:9080/resorts/> and press **Enter**.

The **Mod Resorts** application home page is displayed.



- ___b. Click **WHERE TO?** dropdown menu to see the city list.



- ___c. Click **PARIS, FRANCE** from the list, it shows the weather of the city.

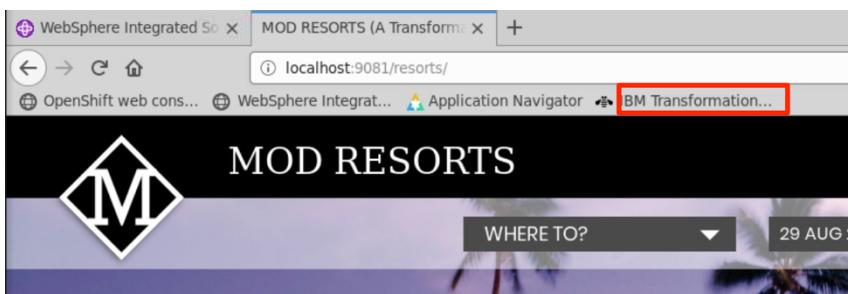


Now you have reviewed the application, next you use Transformation Advisor to analyze this application to see if it is good candidate to be moved to the cloud.

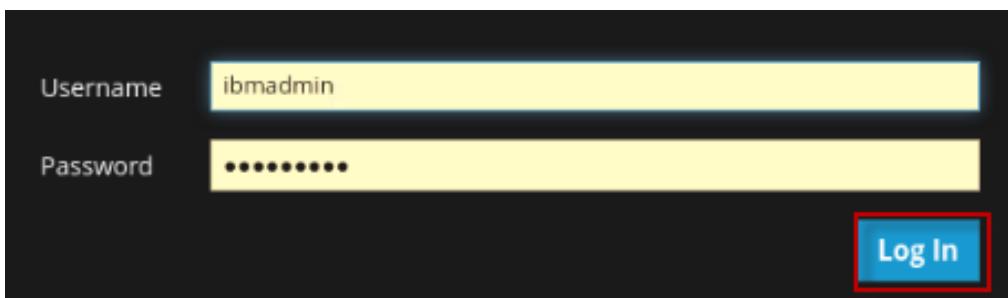
6.3 Upload Application Data into Transformation Advisor

The Transformation Advisor can evaluate any Java based applications and help to package the good candidate application to move to cloud. In Lab1 of this lab series, you have demonstrated how to use the tool to evaluate the on-premises WebSphere application, **Mod Resorts**, and have determined that it is suitable to move this application to cloud without any code change. Now you need to move forward to create the application migration bundle with the Transformation Advisor. If you have completed Lab1 with this lab environment, you already have the application data loaded to the Transformation Advisor, you can skip this task and move to the next one. Otherwise continue to execute the procedures below to load the application data to Transformation Advisor first.

- From web browser window, click the **IBM Transformation Advisor** bookmark and launch it.



- a. Since the **Transformation Advisor** is deployed to **RHOCP** as an operator, you are re-directed to the **RHOCP Login** page. Click **htpasswd** field. Then login with **ibmadmin / engageibm** as the username and password.



The **Transformation Advisor Home** page is displayed.

IBMCLOUD Transformation Advisor

Welcome to Transformation Advisor

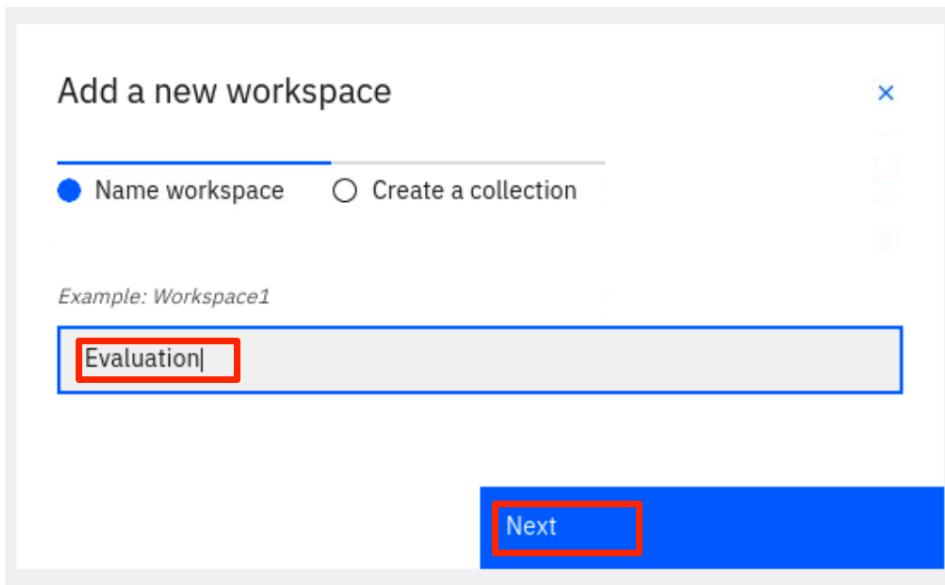
We know that modernizing your existing middleware deployments can take you into unfamiliar territory. IBM Cloud Transformation Advisor will help you take your first steps toward getting you onto Cloud Pak for Apps.

Let's get started.

Add a new workspace +

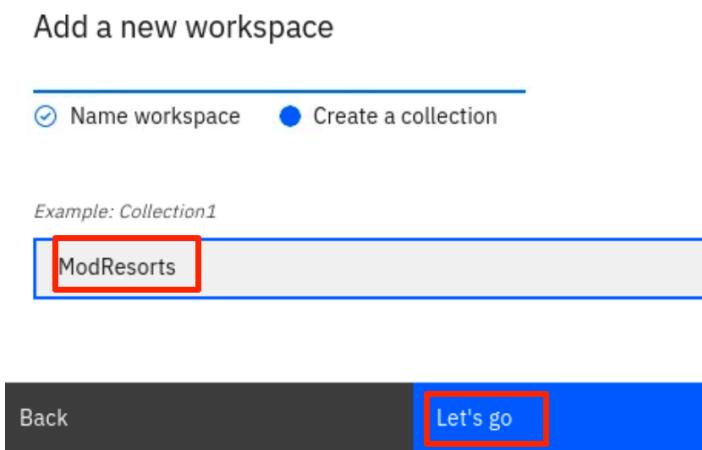
You have no workspace, try creating one!

- 2. Create a new workspace by entering the workspace name as **Evaluation** and then clicking **Next**.



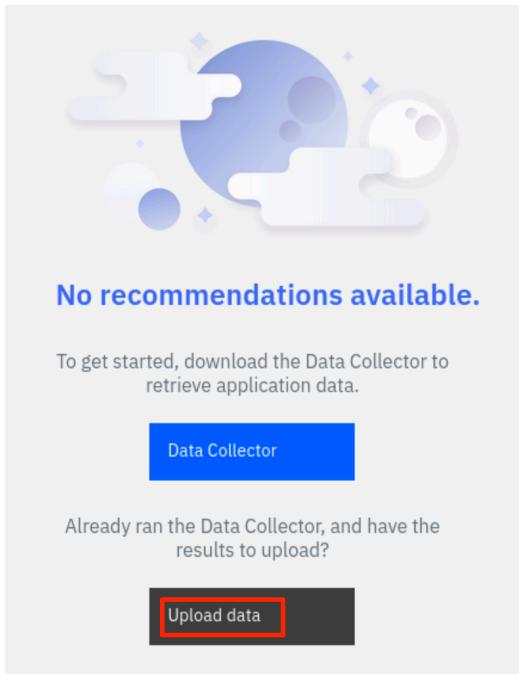
Note: A workspace is a designated area that houses the migration recommendations provided by **Transformation Advisor** against your application server environment. You can name and organize these however you want, whether it's by business application, location or teams.

- 3. Enter the collection name as **ModResorts** and click **Let's go**.



Note: Each workspace can be divided into collections for more focused assessment and planning. Like workspaces, collections can be named and organized in whatever way you want.

- 4. Once the Workspace and Collection are created, you get the data collection options page. You can either download the Data Collector utility and run it against your application server, or upload an existing data file. In this lab, you are going to use the upload option. Click **Upload data** to go to the upload page.



5. Click **Drop or Add File**.

Upload data

When you run the Data Collector on a profile, it creates a zip file with the same name as the profile. You can find the zip file(s) in the /path/to/transformationadvisor directory that is created by the Data Collector.

The following example shows the zip file for a profile named AppSrv01:

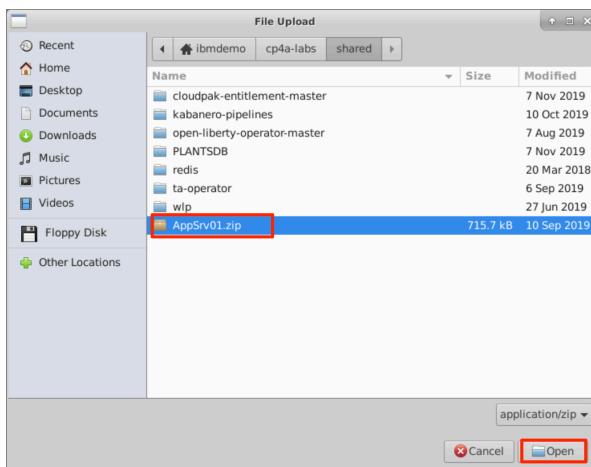
```
$ pwd  
/path/to/transformationadvisor  
$ ls  
AppSrv01 AppSrv01.zip bin  
conf environment.json jre lib
```

Copy the zip file(s) to a location where you can access them with this browser and select them using the Drop or add file button below.

Please upload 1 file at a time.



6. Navigate to **/home/ibmdemo/cp4a-labs/shared** directory, select **AppSrv01.zip** file and click **Open**.



7. Click **Upload**.

Upload data

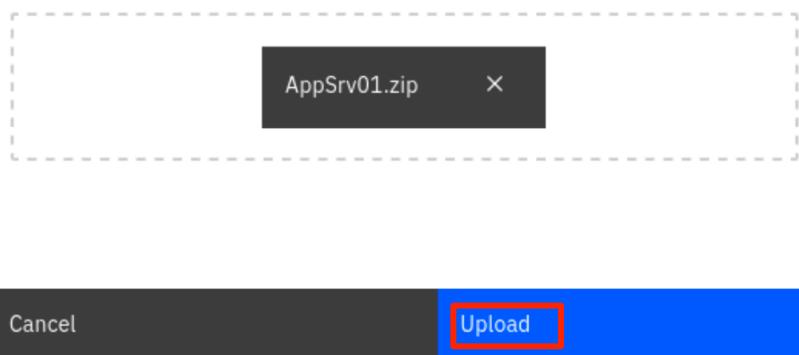
When you run the Data Collector on a profile, it creates a zip file with the same name as the profile. You can find the zip file(s) in the /path/to/transformationadvisor directory that is created by the Data Collector.

The following example shows the zip file for a profile named AppSrv01:

```
$ pwd
/path/to/transformationadvisor
$ ls
AppSrv01 AppSrv01.zip bin
conf environment.json jre lib
```

Copy the zip file(s) to a location where you can access them with this browser and select them using the Drop or add file button below.

Please upload 1 file at a time.



Within a few minutes, the application data is uploaded to the **Transformation Advisor**. You see all applications deployed to the WAS server listed.

Name	Tech match	Dependencies	Issues	Estimated dev cost in days
DefaultApplication.ear	Complex (Complex)	85%	3	▲ 4 ■ 1 ● 4 14
lvApp.ear	Moderate (Moderate)	100%	2	▲ 2 ■ 1 ● 2 3
modresorts-1_0_war.ear	Simple (Simple)	100%	None	● 1 0
PlantsByWebSphereEE7.ear	Simple (Simple)	100%	3	● 5 0
query.ear	Moderate (Moderate)	100%	2	▲ 2 ■ 1 ● 3 3

The Transformation Advisor provides all migration recommendations for all applications deployed to the WAS server based on the specified source and target environments.

On the **Recommendations** page, the identified migration source environment is shown in the Profile section, and the target environment is shown in the Preferred migration section. The data collector tool detects that the source environment is your WebSphere Application Server ND **AppSrv01** profile. The target environment is **Liberty on OpenShift**, which is the default target environment.

The Recommendations page also shows the summary analysis results for all the apps in the **AppSrv01** environment to be moved to a Liberty on OpenShift environment. For each app, you can see these results:

- Name
- Complexity level
- Technology match
- Dependencies
- Issues
- Estimated development cost in days

For example, if you want to move the **modresorts-1_0_war.ear** application to **Liberty on OpenShift**, the complexity level is **Simple** and the Tech match is **100 percent**. A Simple complexity score indicates that the application code does not need to be changed before it can be moved to cloud. The application has no dependency, has 1 minor level issue and the estimated development effort is 0 day with no code change.

6.4 Create Migration Bundle

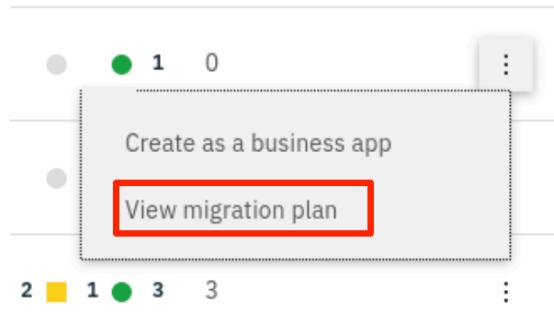
Transformation Advisor has the ability to use the imported application analysis data to generate a migration bundle for the specified application and target environment to accelerate the migration

process while minimizing the errors and risks. You use it to create our migration package for moving the Mod Resorts application to Liberty container on OpenShift cluster environment.

- 1. Click the action icon in **modresorts-1_0_war.ear** application row.

Java applications (5)						Export	Upload options
Name	Tech match	Dependencies	Issues	Estimated dev cost in days			
DefaultApplication.ear	Complex	85%	3	4	1	4	14
ivtApp.ear	Moderate	100%	2	2	1	2	3
modresorts-1_0_war.ear	Simple	100%	None	1	0	1	0
PlantsByWebSphereEE7.ear	Simple	100%	3	1	0	5	0
query.ear	Moderate	100%	2	2	1	3	3

- 2. Select the **View migration plan** action.



- 3. Transformation Advisor is now starting to prepare the migration bundle package for the application. It quickly prepares a migration bundle package with several required key files created from the application data collected from the WAS server, including server.xml, pom.xml, OpenShift Operator resource files and Dockerfile. However, it needs a few more application specific dependencies like the application runtime binary file and other library files, like database driver file the application is using. For the Mod Resorts application, it only needs the application runtime binary file. Select **Binary** option and **Manual upload** option, then click **Drag or add file** in **Application binary** row.

Migration bundle

The files included in your migration bundle help you migrate to IBM WebSphere Liberty, create an image, and package your application as a Kubernetes Operator for easy deployment.

Build type (i)

Select the type of application you want to build to help Transformation Advisor determine what files to include in the bundle.

- Source code
- Binary

Use Accelerator for Teams Collection (i)

Keep "Use" selected to have your files set up to use Accelerator for Teams, or deselect to manually set up your Git repository. If you do not select "Use" here, your files will still include instructions to set up Accelerator for Teams later.

- Don't use ⚠

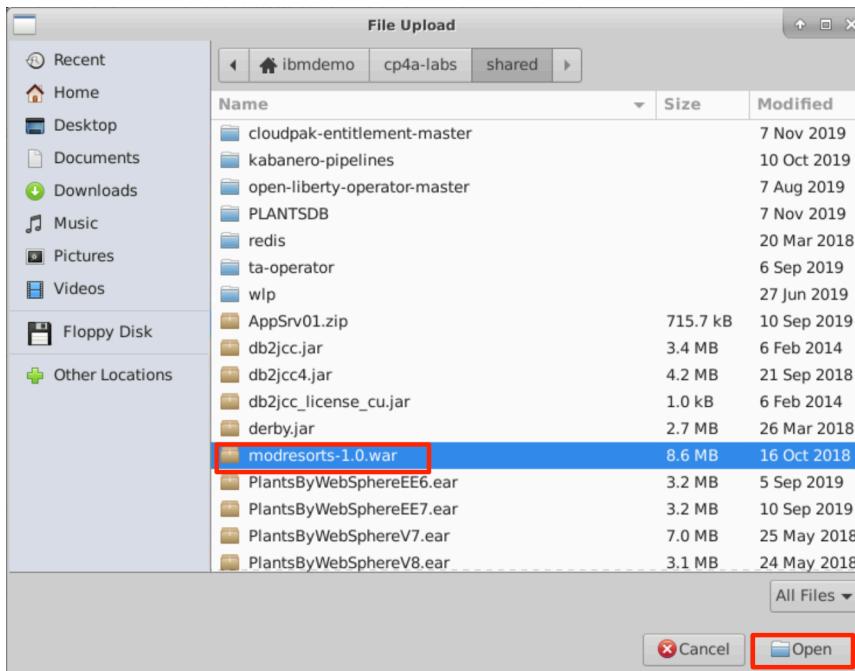
Application dependencies

Transformation Advisor detected some application dependencies. Please Provide the dependencies in addition to the application binary by manually uploading or supplying your Maven coordinates:

- Manual upload
- Maven repository

Detected dependencies	Uploaded files
Application binary	<input style="width: 100%; height: 100%;" type="button" value="Drag or add file"/>

4. Navigate to **/home/ibmdemo/cp4a-labs/shared** directory, select **modresorts-1.0.war** file and click **Open** to add the file to the bundle.



Once the file is added, your application migration bundle is completed and ready for use.

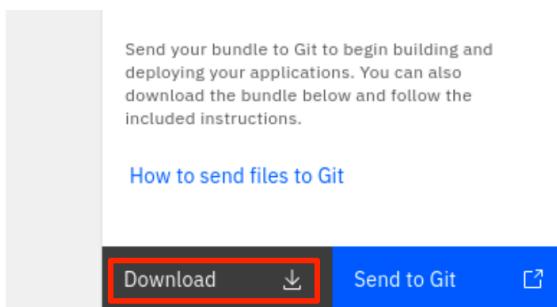
If you are sure that no more changes are needed for the migration bundle, you can push it to your GitHub repository and use your delivery pipeline to deploy it to cloud.

In this lab, you are going to make a few changes and deploy the bundle manually.

6.5 Update the Migration Bundle

After the migration bundle is created, you need to do one additional update.

1. Click **Download** to download it to the local machine.



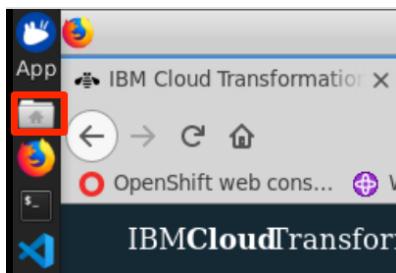
Your migration bundle is downloaded to **/home/ibmdemo/Downloads** directory.

```
Error 404: javax.servlet.ServletException: java.io.FileNotFoundException: SRVE0190E: File not found: /e57304c6-9c72-4238-951e-cc19143509db/40ef1302-f81e-4994-8395-05c1b72b2376/AppSrv01/modresorts10war/modresorts10war_migrationBundle.zip
```

Note: If you see the following error, that means that there is an issue in the **modresorts-1.0.war** uploading process. You need to remove the **modresorts-1.0.war** added and re-add it. That will fix the issue.

```
Error 404: javax.servlet.ServletException: java.io.FileNotFoundException: SRVE0190E: File not found: /e57304c6-9c72-4238-951e-cc19143509db/40ef1302-f81e-4994-8395-05c1b72b2376/AppSrv01/modresorts10war/modresorts10war_migrationBundle.zip
```

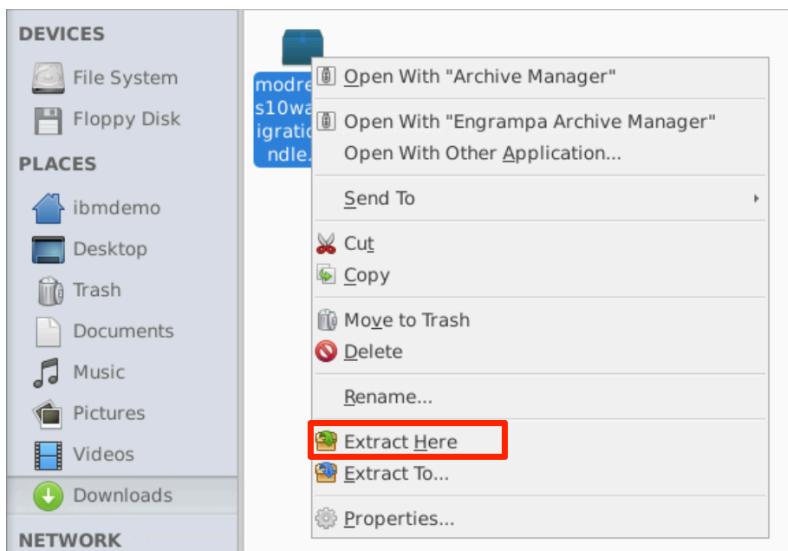
2. Click **File Manager** icon in the Workstation VM desktop toolbar.



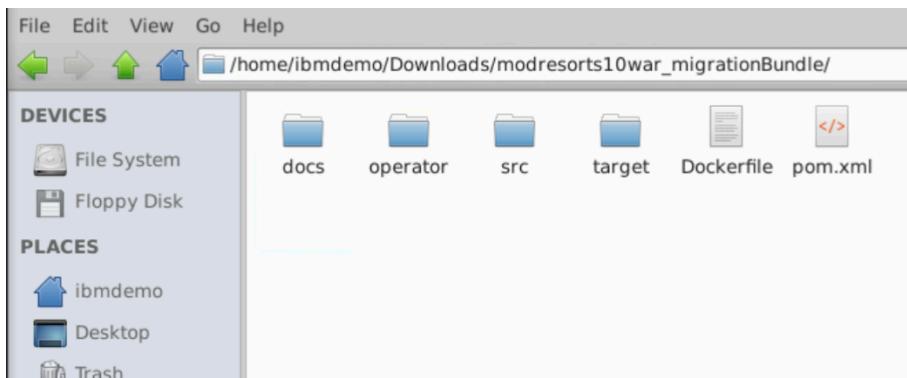
- _3. Navigate to **/home/ibmdemo/Downloads** directory.



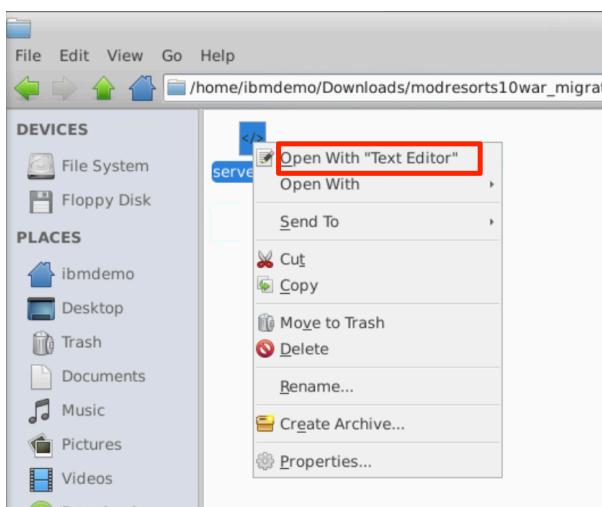
- _4. Right click the migration bundle zip file and select **Extract Here** to unzip the bundle pack in the **/home/ibmdemo/Downloads** directory



5. Navigate to the **modresorts10war_migrationBundle** directory, you see the all files in the migration bundle package. The step-by-step instruction on how to deploy the migration bundle is stored in the **docs** directory.



6. Navigate to **modresorts10war_migrationBundle/src/main/liberty/config** directory, update the Liberty **server.xml** file by selecting **server.xml** file, right clicking and opening it with **Text Editor**.



7. Change the servlet feature from **3.0** to **3.1**, because the application requires servlet 3.1 on Liberty server and the default highest supported version in WAS v855 is servlet 3.0. Click **Save** to save the changes.

```
*server.xml
<?xml version="1.0" encoding="UTF-8"?><!--Generated by IBM TransformationAdvisor
Sun Sep 08 19:57:29 GMT 2019--><server>
<!!--These elements have been identified from this application's configuration.-->
<featureManager>
<feature>Servlet-3.1</feature>
</featureManager>
<!-- <variable name="DERBY_JDBC_DRIVER_PATH" value="${WAS_INSTALL_ROOT}/derby/lib"/> -->
<httpEndpoint host="*" httpPort="9080" httpsPort="9443" id="defaultHttpEndpoint"/>
<!-- <variable name="DB2UNIVERSAL_JDBC_DRIVER_NATIVEPATH" value="/" /> -->
<variable name="DERBY_JDBC_DRIVER_PATH" value="${shared.config.dir}/lib/global"/>
<variable name="DB2UNIVERSAL_JDBC_DRIVER_NATIVEPATH" value="${shared.config.dir}/lib/global"/>
<applicationManager autoExpand="true"/>
<!!--These elements are from the profile level configuration. Not all of them may be necessary for
your application.-->
<!!--<dataSource containerAuthDataRef="workstationNode01/db2User" id="pbwDBS"
jdbcDriverRef="DB2_Universal_JDBC_Driver_Provider_(XA)" jndiName="jdbc/
n1-->
```

6.6 Testing the Migration Bundle Locally (Optional)

Now you have the **Mod Resorts** application migration package. In general, before deploying any application to the OpenShift cluster, you need to deploy it to a local Liberty server for testing first. To save time, you can skip this task, but the task steps are documented here for your reference.

- 1. Go back to the terminal window and issue the command below to stop the WAS server.

```
sudo /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/bin/stopServer.sh server1
```

if prompted, enter the **sudo** password as **passw0rd**, and WAS admin user/password as **wsadmin/passw0rd**.

- 2. Create a Liberty server named **server1** in the Liberty installation directory to test the application:

```
/home/ibmdemo/cp4a-labs/shared/wlp/bin/server create server1
```

- 3. Copy the application binary file in the migration bundle **target** directory to the **apps** directory of Liberty:

```
cp /home/ibmdemo/Downloads/modresorts10war_migrationBundle/target/*.war  
/home/ibmdemo/cp4a-labs/shared/wlp/usr/servers/server1/apps
```

- 4. Copy the **server.xml** file to local Liberty server:

```
cp  
/home/ibmdemo/Downloads/modresorts10war_migrationBundle/src/main/liberty/config/server.xml /home/ibmdemo/cp4a-labs/shared/wlp/usr/servers/server1/server.xml
```

- 5. Start the Liberty server:

```
/home/ibmdemo/cp4a-labs/shared/wlp/bin/server start server1
```

- 6. Check the Liberty logs to confirm that your application has started correctly and to find the application URL for access:

```
cat /home/ibmdemo/cp4a-labs/shared/wlp/usr/servers/server1/logs/messages.log
```

You see from the log message the application is available with a URL like **http://xxx.xxx.xxx.xxx:9080/resorts**.

- 7. In the web browser window, type the application as: **localhost:9080/resorts**, and press **Enter**. You are able to access the application from the local Liberty server.

6.7 Containerize Liberty Application

In this task, you containerize the application. You first create a Liberty Docker image that has the **Mod Resorts** application installed and configured, and then you test the image to confirm that it is operating correctly.

- 1. If you do not do tasks in section 6.6, you can skip this step. Otherwise from the terminal window issue the command below to stop the Liberty server and to free the port:

```
/home/ibmdemo/cp4a-labs/shared/wlp/bin/server stop server1
```

- 2. Go to where your migration artifacts are located and build your image from the docker file.

```
cd /home/ibmdemo/Downloads/modresorts10war_migrationBundle
```

```
docker build --no-cache -t default-route-openshift-image-
registry.apps.demo.ibmdte.net/modresorts10war/modresorts10war:latest .
```

where

- **default-route-openshift-image-registry.apps.demo.ibmdte.net** is the OpenShift Docker image registry
- **modresorts10war** is the name of namespace in OpenShift you are using to host the image.
- **modresorts10war:latest** is the application container image.

The base Liberty image is pulled down and used to create the image that includes your migrated application.

- 3. Once the Docker image is built, create a container instance from the image and confirm that it is working correctly:

```
docker run -p 9080:9080 default-route-openshift-image-
registry.apps.demo.ibmdte.net/modresorts10war/modresorts10war:latest
```

- 4. If everything looks good, the container has been started and mapped to the port 9080. You can access it from your browser with this link: **localhost:9080/resorts/**.

- 5. After testing, press **Ctrl-C** to stop the container.

6.8 Deploy the Application to OpenShift

In this step you deploy the docker image you have created to Red Hat OpenShift and create an instance of it. Before you begin, you need push the image to the OpenShift image registry.

Note: The migration artifacts generated by Transformation Advisor (specifically the **operator/application/application-cr.yaml** file) assume that the default Docker Registry is being used. If you choose to use a different registry, remember to update the image property in the YAML file appropriately.

- __1. From the terminal window, issue the command below to login to the **RHOCP** cluster.

```
oc login -u ibmadmin -p engageibm https://api.demo.ibmdte.net:6443
```

- __2. Create a new project (namespace) as **modresorts10war**.

```
oc new-project modresorts10war
```

You see the message to confirm that the **modresorts10war** project is created.

```
ibmdemo@workstation:~/Downloads/modresorts10war_migrationBundle$ oc new-project
modresorts10war
Now using project "modresorts10war" on server "https://10.0.0.4:6443".
You can add applications to this project with the 'new-app' command. For example
, try:
  oc new-app django-psql-example
to build a new example application in Python. Or use kubectl to deploy a simple
Kubernetes application:
  kubectl create deployment hello-node --image=gcr.io/hello-minikube-zero-instant/hello-node
```

- __3. Log in to the OpenShift Docker registry with the command:

```
docker login -u ibmadmin -p $(oc whoami -t) default-route-openshift-image-
registry.apps.demo.ibmdte.net
```

- __4. Execute the following command to push your docker image to OpenShift image repository.

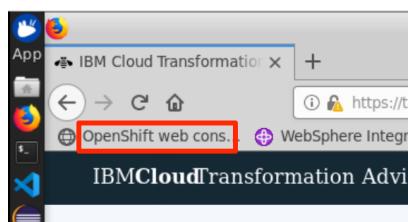
```
docker push default-route-openshift-image-
registry.apps.demo.ibmdte.net/modresorts10war/modresorts10war:latest
```

When it is done, your application docker image **default-route-openshift-image-registry.apps.demo.ibmdte.net/modresorts10war/modresorts10war:latest** is pushed to the **RHOCP** image registry.

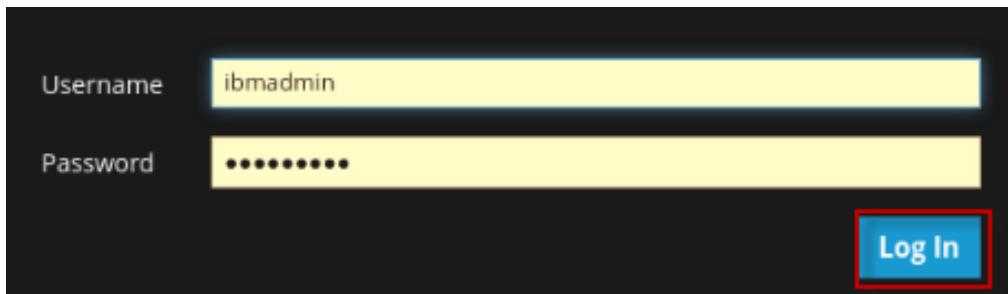
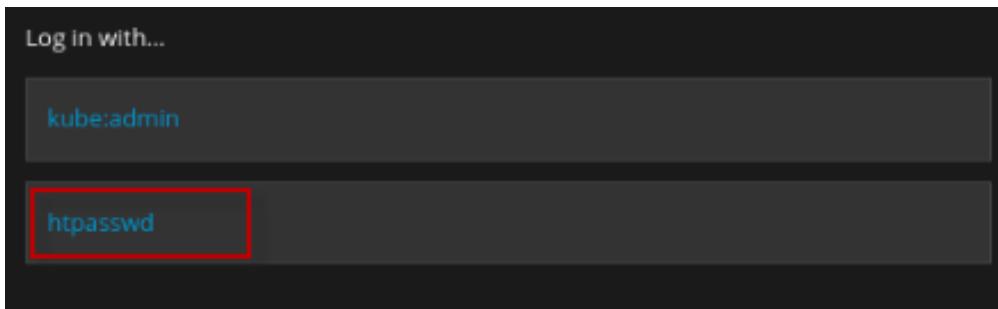
- __5. Verify the pushed Docker image in **RHOCP** cluster.

__a. Click the web browser icon on the desktop toolbar to go back to the browser window.

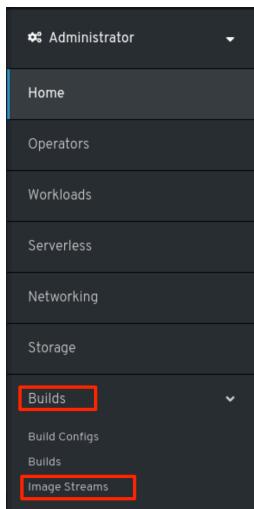
__b. Click **OpenShift web console** bookmark to open the **RHOCP** console.



__c. If the **RHOCP Web Console** Login page is displayed, click **htpasswd** field. Then login with **ibmadmin / engageibm** as the username and password.



__d. From the **RHOCP** Web Console Home page, click **Builds>Images Streams**.

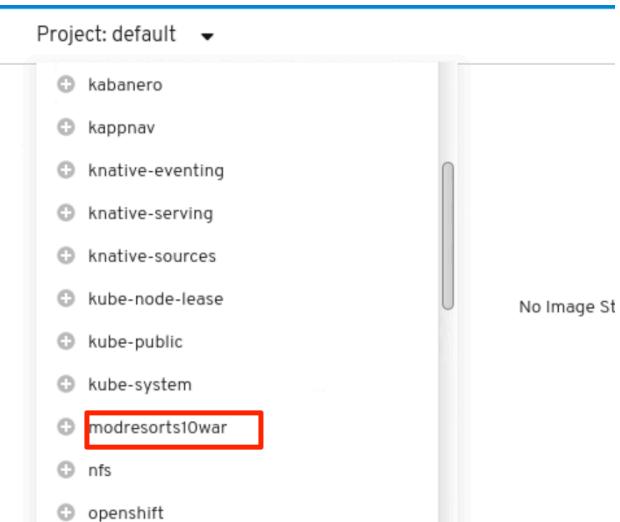


__e. Change project (namespace) from **default** to **modresorts10war**.

Project: default ▾

- ⊕ kabanero
- ⊕ kappnav
- ⊕ knative-eventing
- ⊕ knative-serving
- ⊕ knative-sources
- ⊕ kube-node-lease
- ⊕ kube-public
- ⊕ kube-system
- ⊕ modresorts10war
- ⊕ nfs
- ⊕ openshift

No Image St



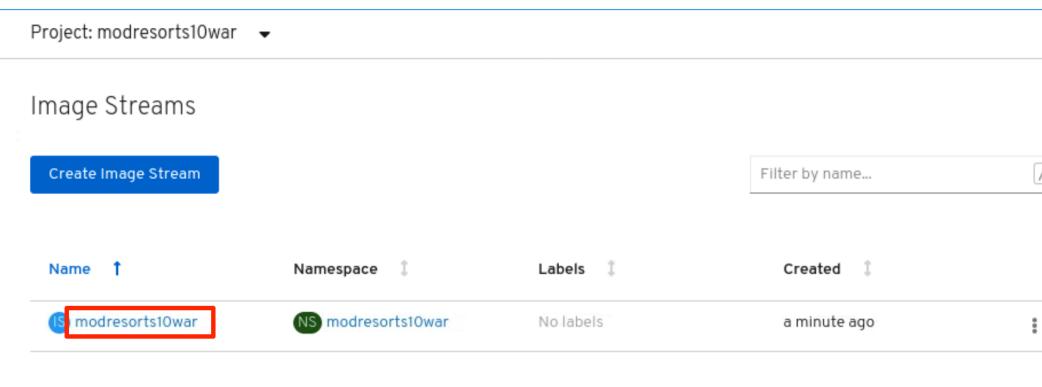
f. You can see the image you just pushed is listed. Click its link to view its details.

Project: modresorts10war ▾

Image Streams

[Create Image Stream](#)

Name ↑	Namespace	Labels	Created	⋮
modresorts10war 	NS modresorts10war	No labels	a minute ago	⋮



In the **Image Stream Details Page Overview** section, you see the public image repository you used to push the image. Notice that the public image repository is mapped to an internal image repository which is used to deploy the application. The internal image repository is: **image-registry.openshift-image-registry.svc:5000/modresorts10war/modresorts10war**.

Project: modresorts10war ▾

Image Streams > Image Stream Details

IS modresorts10war Actions ▾

Overview YAML History

Image Stream Overview

Name
modresorts10war

Namespace
NS modresorts10war

Labels
No labels

Annotations
0 Annotations

Image Repository
image-registry.openshift-image-registry.svc:5000/modresorts10war/modresorts10war

Public Image Repository
default-route-openshift-image-registry.apps.demo.ibmdte.net/modresorts10war/modresorts10war

6. From the terminal window, change to the **operator** directory under the **modresorts10war_migrationBundle** directory.

```
cd /home/ibmdemo/Downloads/modresorts10war_migrationBundle/operator
```

7. Create the custom resource definition (CRD) for the application using the artifact generated by the Transformation Advisor:

```
oc apply -f application/application-crd.yaml -n modresorts10war
```

8. The Liberty operator requires the ServiceAccount, Role, and RoleBinding Kubernetes resources to be created. Run the following commands to create them:

```
oc apply -f deploy/service_account.yaml -n modresorts10war
```

```
oc apply -f deploy/role.yaml -n modresorts10war
```

```
oc apply -f deploy/role_binding.yaml -n modresorts10war
```

9. Create an instance of the Liberty operator:

```
oc apply -f deploy/operator.yaml -n modresorts10war
```

10. Wait for the Liberty operator installation to complete before going to the next step. You can check the status using **oc get pods** command and wait until the **modresorts-operator** pod is ready.

```
oc get pods -n modresorts10war
```

```
ibmdemo@workstation:~/Downloads/modresorts10war_migrationBundle/operator$ oc get
pods -n modresorts10war
NAME                                READY   STATUS    RESTARTS   AGE
modresorts10war-operator-8564f77596-s4tvx  1/1    Running   0          22s
ibmdemo@workstation:~/Downloads/modresorts10war_migrationBundle/operator$
```

- __11. Now deploy the application using the provided custom resource artifact.

oc apply -f application/application-cr.yaml -n modresorts10war

- __12. You can view the status of your deployment by running **oc get pods** command.

- __13. When you see the pod is ready, issue the command below to get its service port:

oc get svc -n modresorts10war

```
ibmdemo@workstation:~/Downloads/modresorts10war_migrationBundle/operator$ oc get
svc -n modresorts10war
NAME           TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)
AGE
modresorts10war-dub3qmf8  NodePort  172.30.238.139  <none>        9443:32164/
TCP   2s
ibmdemo@workstation:~/Downloads/modresorts10war_migrationBundle/operator$
```

As you can see the application https service NodePort is exposed (for example 32164 in this screenshot), therefore the containerized application in the OpenShift cluster can be accessed through URL: <https://master1.demo.ibmdte.net:<NodePort>/resorts> (like <https://master1.demo.ibmdte.net:32164/resorts>).

- __14. Open a new browser window, type in the application URL you get from the step above and press **Enter**.
- __15. If you see the message “Warning: Potential Security Risk Ahead”, click **Advanced>Accept the Risk and Continue** to continue. The **Mod Resorts** application home page shows up.
- __16. Navigate through the application web pages as you did in the WAS version to verify the application deployment.

7. Summary

In this lab, you have learned how to use Transformation Advisor to prepare a migration bundle for your application and to deploy it to cloud. As a part of IBM Application Modernization solutions in **IBM Cloud Pak for Applications**, the **Transformation Advisor** accelerates application migrating to cloud process, minimize errors and risks and reduce time to market. To learn more about IBM Application Modernization solutions, please visit [Cloud Pak for Applications](#).

Congratulations! You have successfully completed the lab “Modernize a Java Application for Container and OpenShift with Transformation Advisor”

Appendix A. Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have

been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental. All references to fictitious companies or individuals are used for illustration purposes only.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Appendix B. Trademarks and copyrights

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	AIX	CICS	ClearCase	ClearQuest	Cloudscape
Cube Views	DB2	developerWorks	DRDA	IMS	IMS/ESA
Informix	Lotus	Lotus Workflow	MQSeries	OmniFind	
Rational	Redbooks	Red Brick	RequisitePro	StrongLoop	System i
<i>System z</i>	<i>Tivoli</i>	<i>WebSphere</i>	<i>Workplace</i>	<i>System p</i>	

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of The Minister for the Cabinet Office, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

NOTES

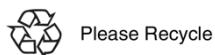
NOTES



© Copyright IBM Corporation 2020.

The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. This information is based on current IBM product plans and strategy, which are subject to change by IBM without notice. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way.

IBM, the IBM logo and ibm.com are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.



Please Recycle
