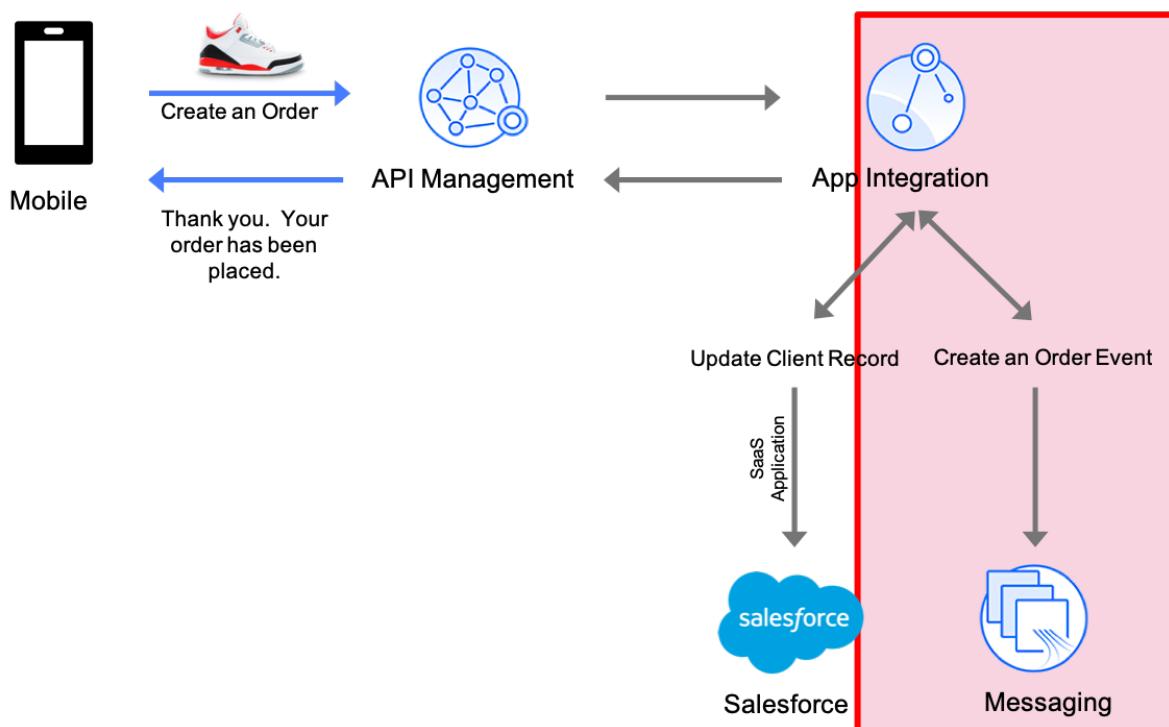


Connect an App Integration with Message Queue using the Cloud Pak for Integration

In this tutorial you will use the Cloud Pak for Integration to create an App Integration flow that will send messages to a queue.

Implement enterprise grade messaging that is secure and reliable for any application across your backend integration architecture

Modern applications and APIs all need the ability to communicate data reliably between mission critical systems across internal/external data sources, networks, and regions. In mission critical environments your messaging infrastructure must be robust, reliable, and secure with the ability to integrate into your applications and APIs at pace and scale. In this tutorial, you will create a message queue that will receive order data from an API call to a retail ordering system. The red box in the diagram shows what you will be creating and where it fits in the overall architecture of a mobile retail buying application.



In this tutorial, you will explore the following key capabilities:

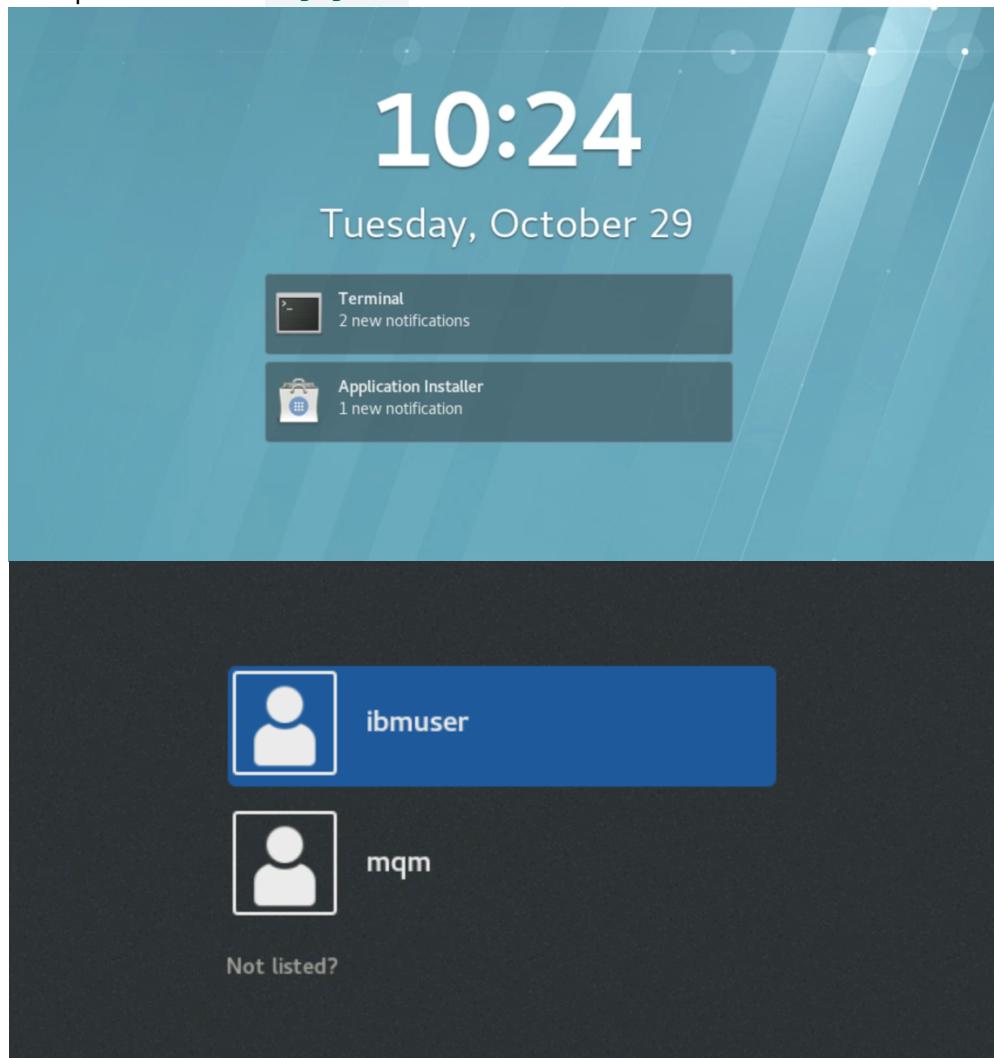
- Explore multiple integration capabilities within a single platform.
- Explore MQ Console
- Create an integration flow that connects to a message queue and sends data to a topic.
- Deploy the integration flow as a container in Kubernetes.
- Check the message using MQ Web Console
- How to delete App Connect Instance and Helm Releases

Task 1 -- Initial Setup Steps

As this is a new deployment of the IBM Cloud Pak for Integration, you must execute some steps to prepare the environment. Initial setup steps are only needed for a fresh installation of the platform. They do not need to be repeated.

Start the Environment

1. Log into the Linux desktop with the following steps and credentials
 - a. Click **ENTER** to view the Linux desktop.
 - b. userid: **ibmuser**
 - c. password: **engageibm**



Confirm the Cloud Pak for Integration is up and running

1. The next step is to check if the environment is done loading. Open a terminal console window by clicking the Terminal icon in the Desktop.
 - a. Type `./startup-scripts/oc-startup.sh`
 - b. Type `oc get pods -n integration`

```
[ibmuser@developer ~]$ oc get pods -n integration
NAME                               READY   STATUS    RESTARTS   AGE
assetrepo-1-asset-files-api-9946b849d-wwlg   1/1     Running   0          3m
assetrepo-1-catalog-api-76cf9d474f-wpn62   1/1     Running   0          3m
assetrepo-1-clt-db-0                   2/2     Running   0          3m
assetrepo-1-clt-haproxy-7856dfb8d6-2jp4r   2/2     Running   0          3m
assetrepo-1-dc-main-5c467f68f7-z62p8   1/1     Running   0          3m
assetrepo-1-portal-catalog-867f9594c5-dvwr5   1/1     Running   0          3m
assetrepo-1-portal-common-api-66498db67b-9gwd2   1/1     Running   0          3m
assetrepo-1-redis-ha-sentinel-86ddcb88dd-ll2l2   1/1     Running   0          3m
assetrepo-1-redis-ha-server-b5ccd65c8-z54ds   1/1     Running   0          3m
ibm-icp4i-prod-ibm-icp4i-prod-6d5648d965-pqksh   2/2     Running   8          37d
```

Note: The third command ensure the Cloud Pak is running. After the third command, once all the pods show 1/1 or equivalent proceed to the next step.

Sync Helm Repositories

1. The Helm repositories must be resynchronized between the repository and the server. Click on the **IBM Cloud Private** bookmark.

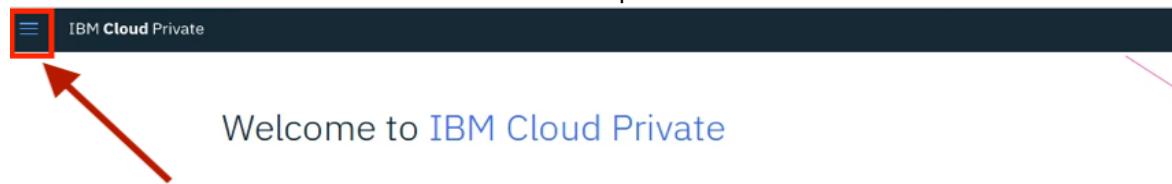


2. Log into IBM Cloud Private with the following credentials:

- a. username: **admin**
- b. password **admin**
- c. Click **Log in**.

A screenshot of the IBM Cloud Private login page. The page features a blue header bar with the 'IBM Cloud Private' logo. Below the header, there is a promotional tagline: 'Fast. Flexible. Intelligent. Open. Enterprise-grade.' A horizontal line separates the tagline from the login form. The login form contains fields for 'Username' (with 'admin' entered) and 'Password' (with '*****' entered). A 'Log in' button is at the bottom right of the form. A red arrow points from the 'IBM Cloud Private' bookmark in the previous image to the logo on this page. Another red box highlights the 'Log in' button.

3. Click the main menu icon at the top left.



4. Click **Manage -> Helm Repositories**

X IBM Cloud Private

Dashboard

Container Images

Workloads

Network Access

Configuration

Platform

Manage

- Identity & Access
- Resource Security
- Service Brokers

Helm Repositories

Name Helm Repositories

Command Line Tools

Getting started

5. Click **Sync all** and click **Sync** in the new window to confirm.

The screenshot shows the Helm Repositories page in the IBM Cloud Private interface. A modal dialog box is open, asking "Sync all Helm Repositories?". The dialog contains a message: "Are you sure you want to sync all repositories (6 items)? Only charts that are not syncing will be added to the queue. The process might take a few seconds to sync all of the charts." At the bottom of the dialog are two buttons: "Cancel" and "Sync", with "Sync" highlighted by a red circle labeled "1". In the background, the main table lists six repositories with their URLs. To the right of the table, a "Last Sync" column shows times ranging from 24 minutes ago. A red circle labeled "2" highlights the "Sync" button in the bottom right corner of the main interface area.

Name	Url	Last Sync
ibm-charts	https://	24 minutes ago
local-charts	https://	24 minutes ago
mgmt-charts	https://	24 minutes ago
ibm-charts-public	https://	24 minutes ago
ibm-community-charts	https://	24 minutes ago
ibm-entitled-charts	https://raw.githubusercontent.com/IBM/charts/master/repo/entitled/	Completed

Task 2 -- Explore the platform capabilities.

IBM Cloud Pak for Integration provides a single solution for all of your enterprise integration needs. The platform provides a comprehensive set of industry-leading capabilities. Combine the powerful integration capabilities to create, manage, and monitor all of your integrations across applications, messaging, events, APIs, and more.

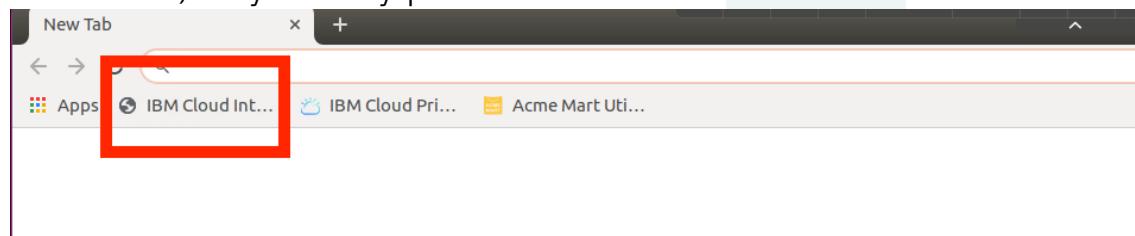
Unlock the power of your data and support the scale required to grow all of your integration and digital transformation initiatives.

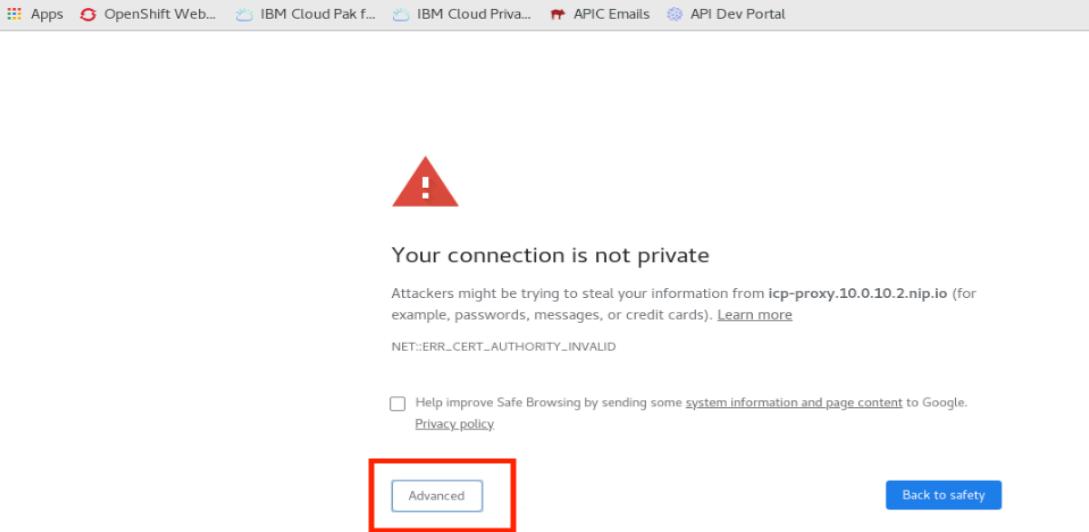
The home page of the **Cloud Pak for Integration** is referred to as the **Platform Navigator**. From the Platform Navigator you are able to navigate to all the integration and development technology contained within the platform.

As of today, the technology included is API management, application integration, message queues, and Kafka event streams. For this lab, we will work with IBM MQ and IBM App Connect.

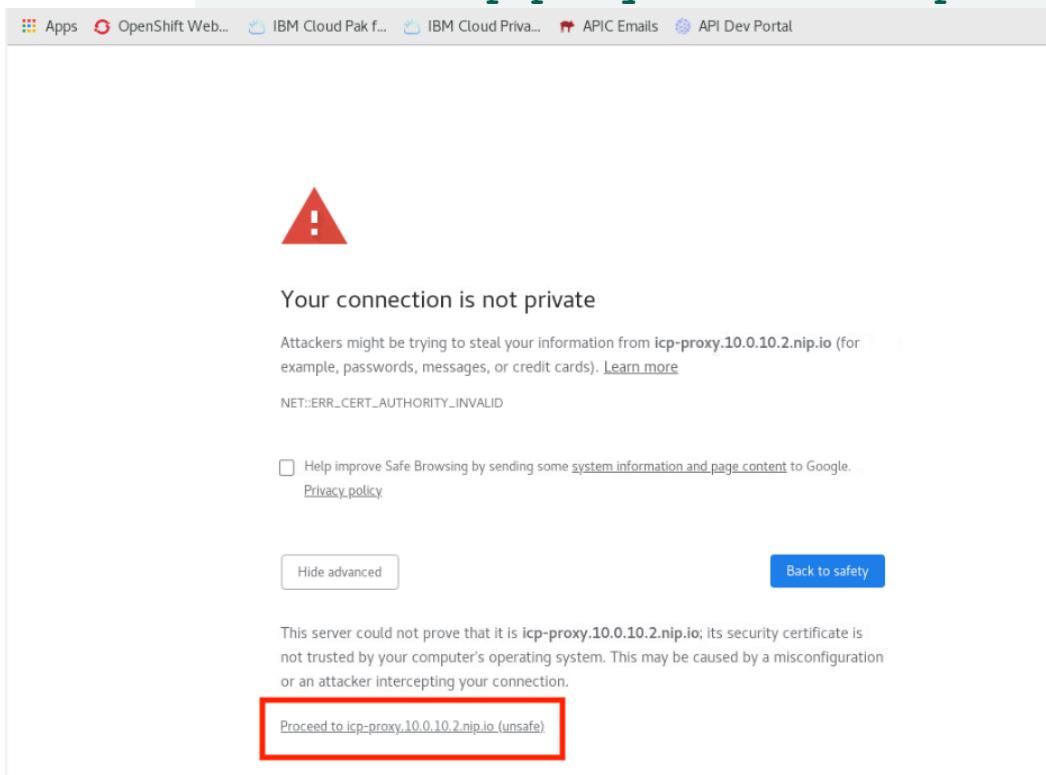
1. Click the **IBM Cloud Pak for Integration** bookmark in the bookmarks bar. If you see the login page, please continue to Task 3 (p. 10)

If instead you see the **Your connection is not private** page, there is an issue reading a certificate. The certificate will not affect this lab, so you may proceed and click **Advanced**.





2. Click **Proceed to icp-proxy.10.0.10.2.nip.io (unsafe)**.



Task 3 -- Verify the existing queue in IBM MQ using MQ Console

In this task you will navigate the MQ Console and check the MQ configuration. You will see the Queue which accepts the resulting data from the call to the "Orders" API.

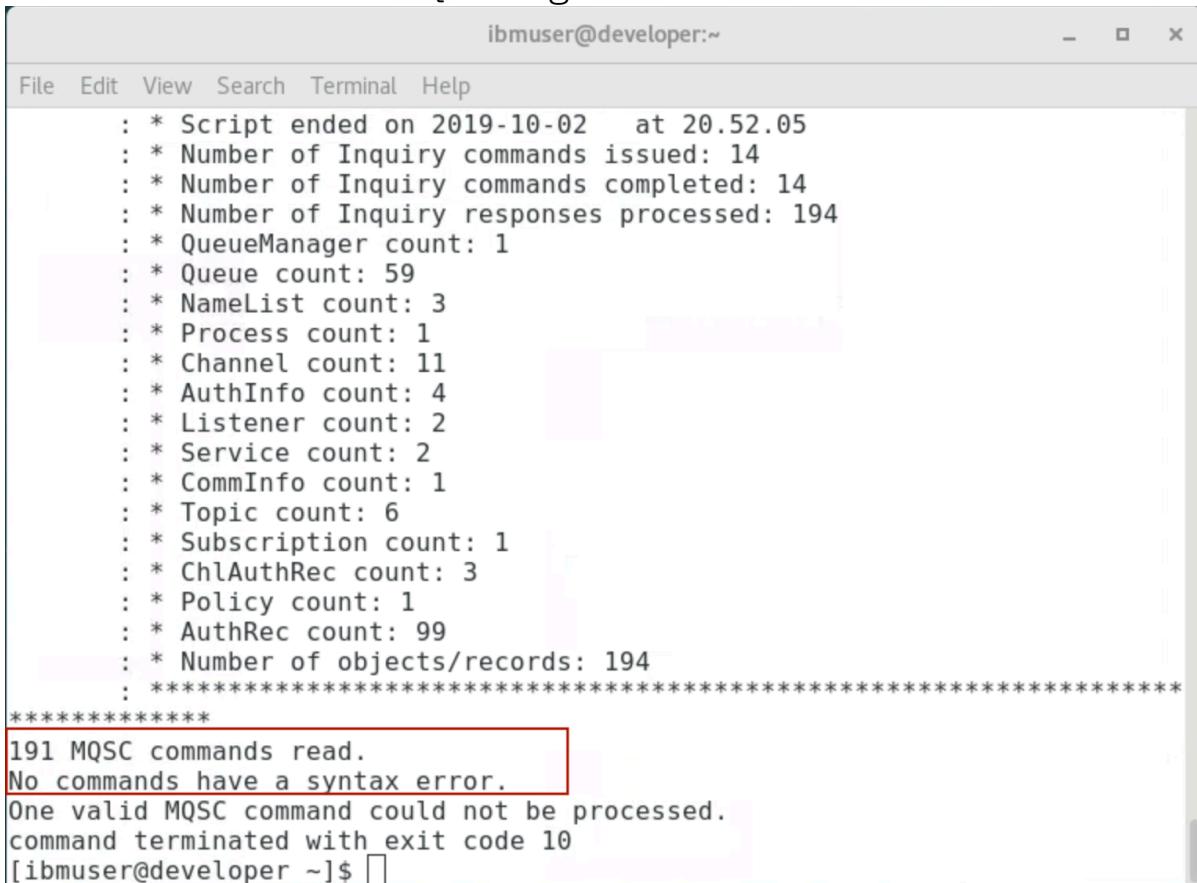
Configure Message Queue (MQ) to Authorize and Accept Data.

1. In this deployment of the Cloud Pak for Integration, all instances of integrations, message queues, and event streams are deployed as microservices. We need to authorize the Message Queue service to accept the incoming data from the integration running on a separate server. Navigate to the `/home/ibmuser` directory, type `./loadmqace.sh`



```
ibmuser@developer:~  
File Edit View Search Terminal Help  
[ibmuser@developer ~]$ ./loadmqace.sh
```

2. Make sure that the MQ configuration has been loaded.



```
ibmuser@developer:~  
File Edit View Search Terminal Help  
: * Script ended on 2019-10-02 at 20.52.05  
: * Number of Inquiry commands issued: 14  
: * Number of Inquiry commands completed: 14  
: * Number of Inquiry responses processed: 194  
: * QueueManager count: 1  
: * Queue count: 59  
: * NameList count: 3  
: * Process count: 1  
: * Channel count: 11  
: * AuthInfo count: 4  
: * Listener count: 2  
: * Service count: 2  
: * CommInfo count: 1  
: * Topic count: 6  
: * Subscription count: 1  
: * ChlAuthRec count: 3  
: * Policy count: 1  
: * AuthRec count: 99  
: * Number of objects/records: 194  
: *****  
*****  
191 MQSC commands read.  
No commands have a syntax error.  
One valid MQSC command could not be processed.  
command terminated with exit code 10  
[ibmuser@developer ~]$
```

3. Go back to the IBM Cloud Private browser.

Click **Menu -> Workloads -> Helm Releases**

The screenshot shows the IBM Cloud Private interface. The top navigation bar has 'IBM Cloud Private' and 'CLUSTER myCluster'. The left sidebar menu is open, showing 'Overview', 'Cluster Services', 'Configuration', 'Manage', 'Platform', 'Workloads' (which is expanded), 'Brokered Services', 'DaemonSets', 'Deployments', 'Helm Releases' (which is selected and highlighted with a red box), 'Jobs', 'StatefulSets', and 'ReplicaSets'. The main content area has a title 'Welcome to IBM Cloud Private'. Below it, there's a section about the platform, one about the catalog, and another about managing your cloud. A large blue box in the center says 'Start using IBM Cloud Private.' with a link 'Install CLI tools'. At the bottom right of the main content area, there's a small note 'Tired of this top'.

4. On the **Helm Releases** catalog, in the search bar at the top of the page: type **mq** and click on **mq-1** release.

The screenshot shows the Helm Releases catalog search results for 'mq'. The search bar at the top contains 'mq' with a magnifying glass icon and a clear button. The results table has columns: Name, Namespace, Status, Chart name, Current version, and Available version. One result is shown: 'mq-1' in namespace 'mq' is 'Deployed' using chart 'ibm-mqadvanced-server-integration-prod' with current version '3.0.0' and available version '4.0.2'. At the bottom, there are pagination controls 'items per page' set to '20' and '1-1 of 1 items'.

5. You will see the MQ configuration. To review the configurations, scroll down to **service**. Click on the MQ **NodePort** line and note the listener port number (32340).

The screenshot shows a table titled 'Service' with columns: Name, Type, Cluster IP, External IP, and Port(s). There are two entries: 'mq-1-ibm-mq-metrics' with Type 'ClusterIP', Cluster IP '172.30.173.4', and Port(s) '9157/TCP'. The second entry is 'mq-1-ibm-mq' with Type 'NodePort', Cluster IP '172.30.73.253', External IP '<none>', and Port(s) '9443:32558/TCP,1414:32340/TCP'. The '32340/TCP' part of the NodePort row is highlighted with a red box.

6. Return to the **IBM Cloud Pak for Integration** browser page.
Click the **mq-1** link in Message Queues.

The screenshot shows the IBM Cloud Pak for Integration Platform home page. It features three main service categories: API Connect, App Connect, and MQ. The MQ category is highlighted with a red box around the 'mq-1' instance. Each service category has a corresponding icon, a list of instances, and a 'Add new instance' button.

Note:

If you see the message "*mq-1 did not load correctly*", click **Open mq-1**. This issue occurs when using a self-signed certificate such as in this demo environment. This will load mq-1 in a new tab. Accept the self-sign certificate. Afterwards come back to the original tab and click **Try again**.

7. Take a moment and review the MQ Console. The MQ server may be administer from the MQ Console. Click **Add Widget**.

Select **Queues** to add queues to the MQ Console. Click **Close**.

The screenshot shows the 'Add a new widget' dialog for the MQ Console. It includes sections for 'Local Queue Managers' and 'Chart'. Below, it says 'Add a widget to display MQ object information for the specified queue manager'. A dropdown menu shows 'Queue manager: mq'. Underneath, there are several options: 'Queues' (highlighted with a red box), 'Topics', 'Listeners', 'Channels', and 'Client-connection Channels'. The 'Add widget' button is circled in red at the top right of the dialog.

8. You will see the **NEWORDER** queue. Verify that **Queue Depth** is 0.

Queues on mq		
Search		Create +
Name	Queue type	Queue depth
AMQ.5DB116AC233398C	Local	13
NEWORDER	Local	0
Total: 2	Last updated: 10:37:47 PM	

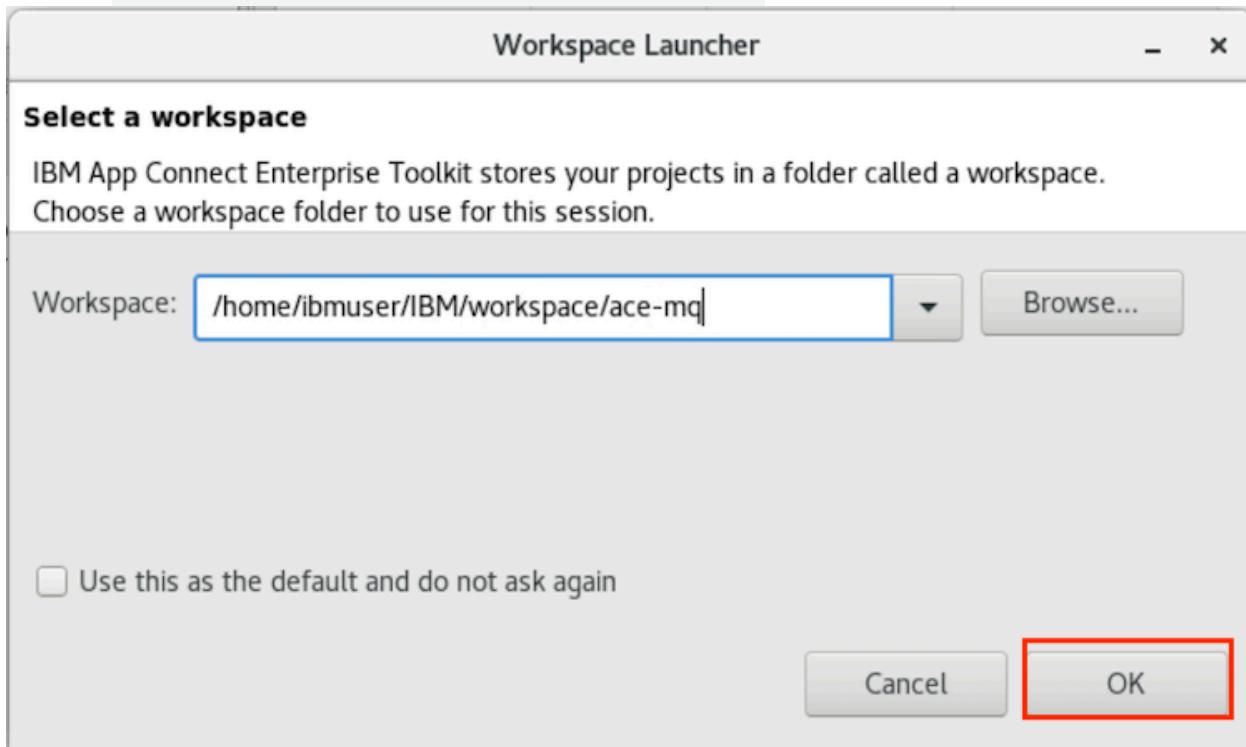
Task 4 -- Modify the Order ACE Flow

In this task you will take an existing integration flow in ACE and modify it to send only the payload data to the NEWORDER queue.

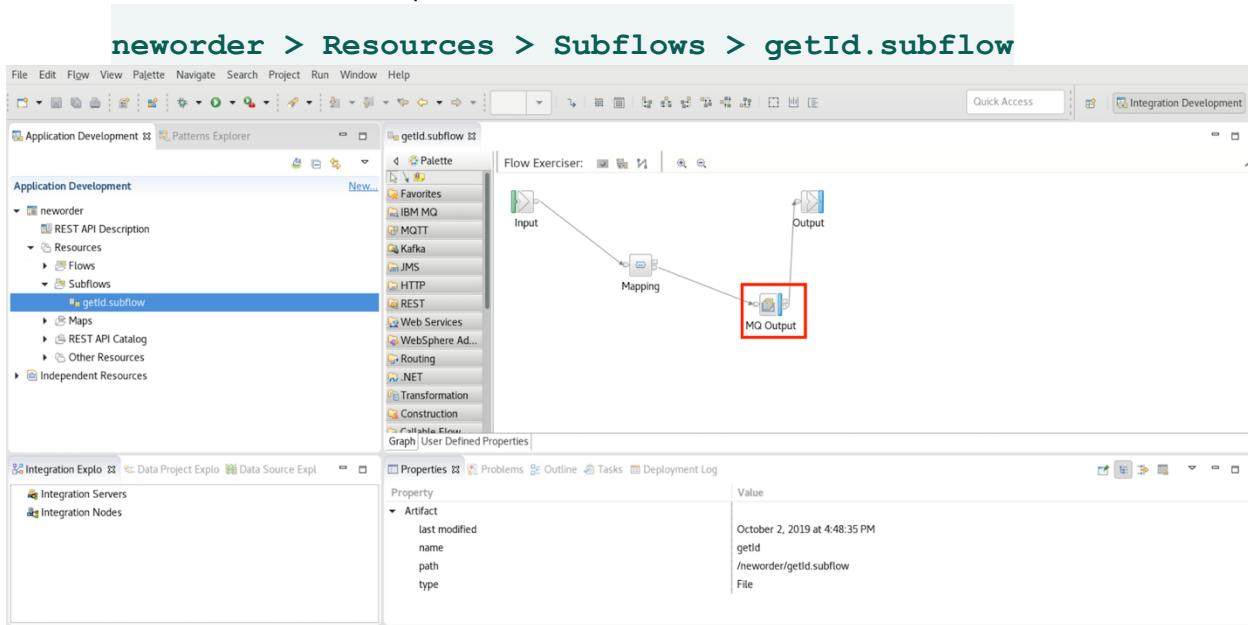
1. In the terminal window, type `ace toolkit` to open the App Connect Enterprise Toolkit.

```
[ibmuser@developer ~]$ ace toolkit  
Starting App Connect Enterprise Toolkit interactively  
[ibmuser@developer ~]$
```

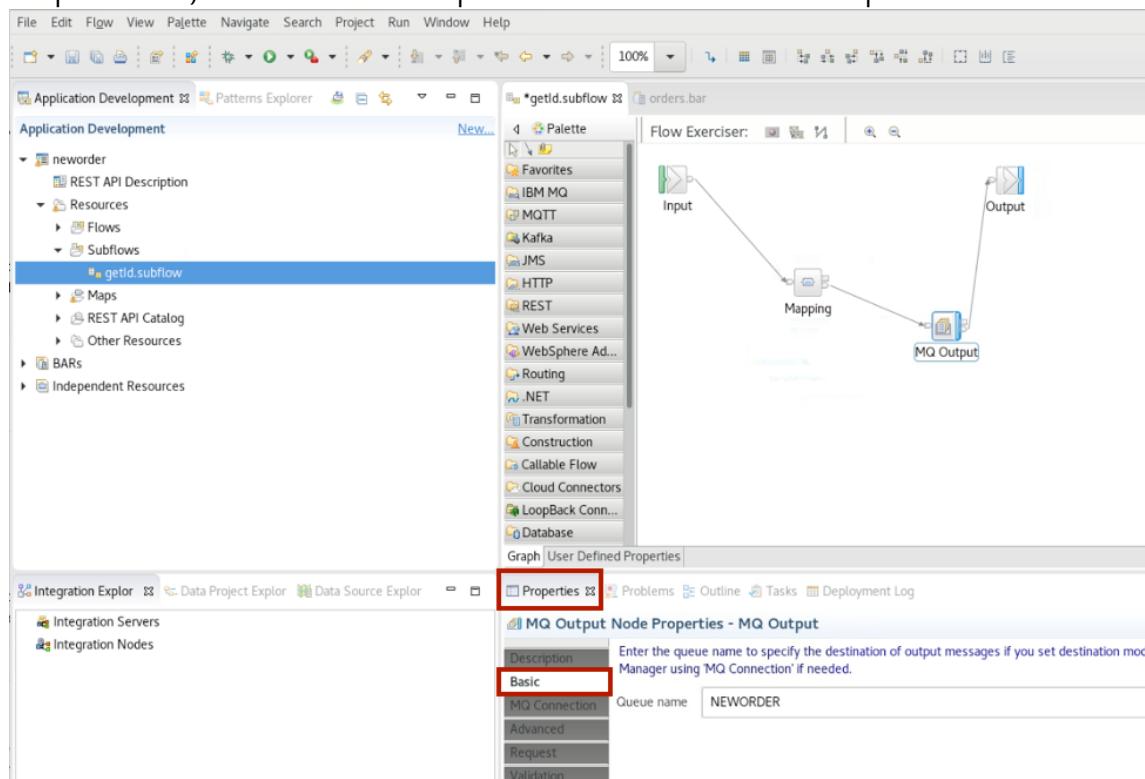
2. In the Workspace Launcher window, choose the workspace `/home/ibmuser/IBM/workspace/ace-mq`. Click OK.



3. On the left-hand panel, double-click

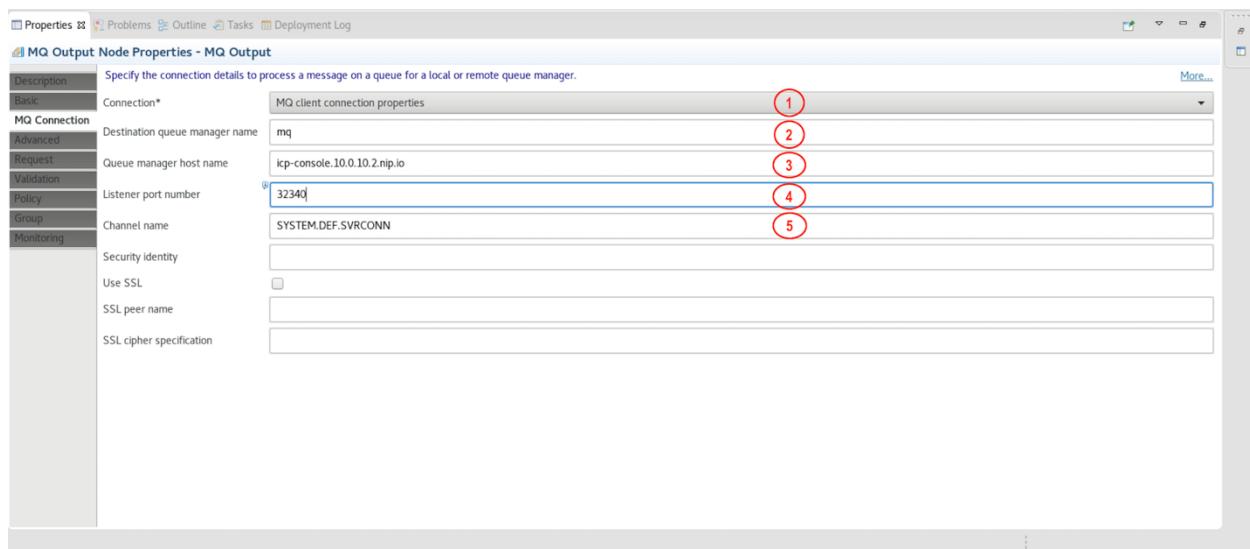


4. Click the **MQ Output** tile. In the Properties pane in the lower right, open the **Properties** tab and click **Basic**. Type the queue name: **NEWORDER** (case sensitive). The queue may already be present, in which case proceed to the next step.



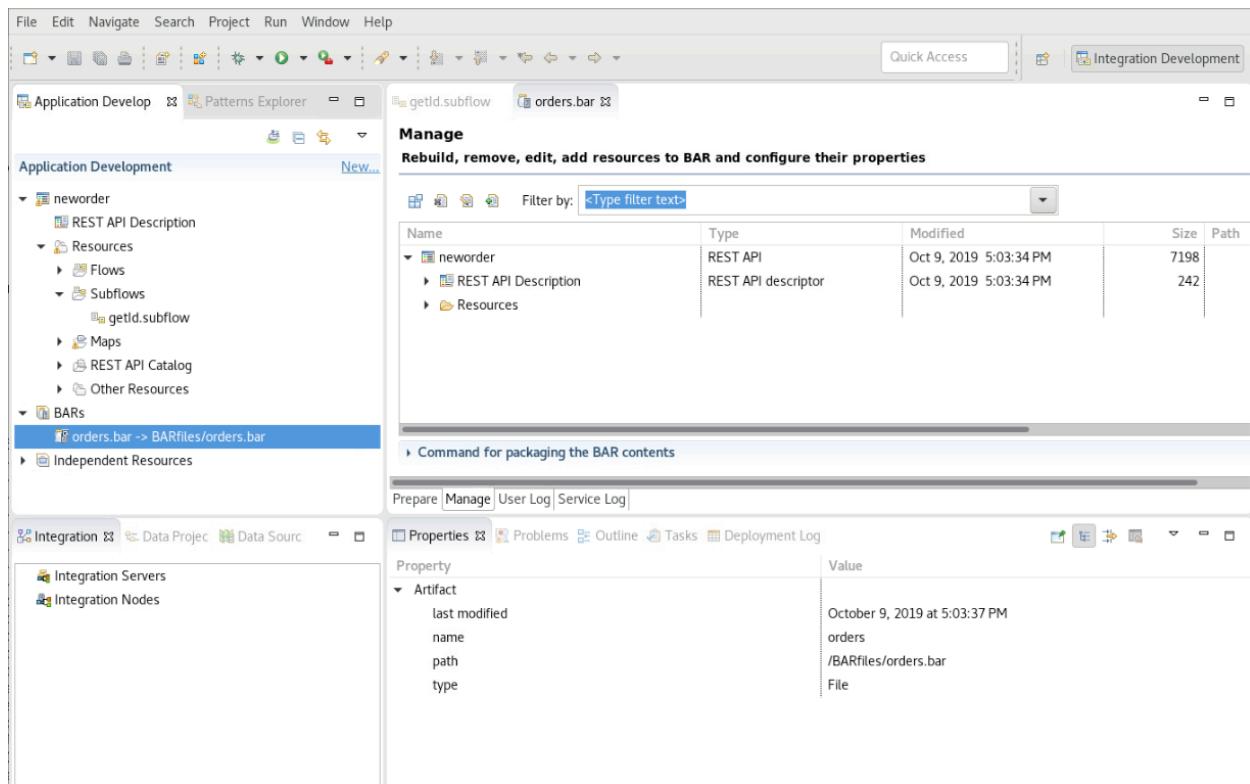
5. Click the **MQ Connection** parameter and verify the settings:

- 1) Connection*: Select **MQ client connection properties** from the drop-down
- 2) Destination queue manager name: **mq** (case sensitive)
- 3) Queue manager host name: **icp-console.10.0.10.2.nip.io**
- 4) Listener: **32340**. You copied this listener port from Step 5 of Task 3: Verify the existing queue in IBM MQ section.
- 5) Channel name: **SYSTEM.DEF.SRVCONN**
- 6) Save your flow (CRTL+S)



6. The App Connect Enterprise server uses BAR files, also known as broker archive, to save compiled message flows, libraries, etc. We have created the file **orders.bar** for this lab.

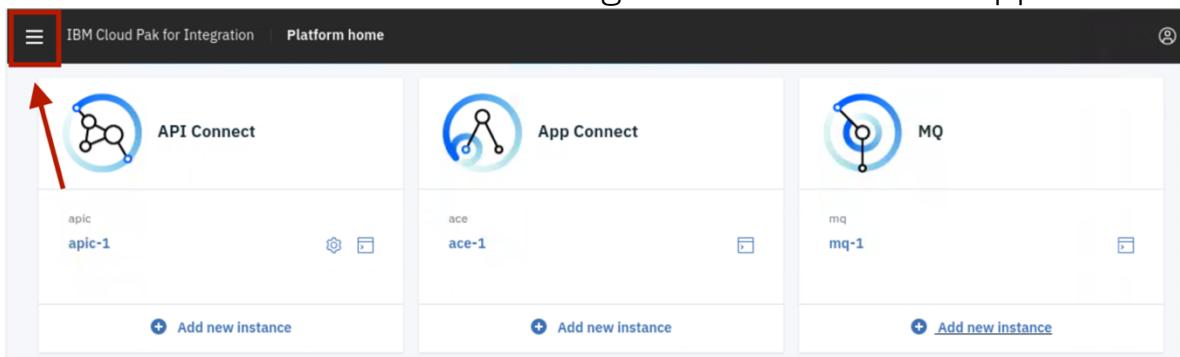
In the **Application Development** window in the upper left, click **BARs** then double-click on **orders.bar**. You can review the details in the upper-right and lower-right panes.



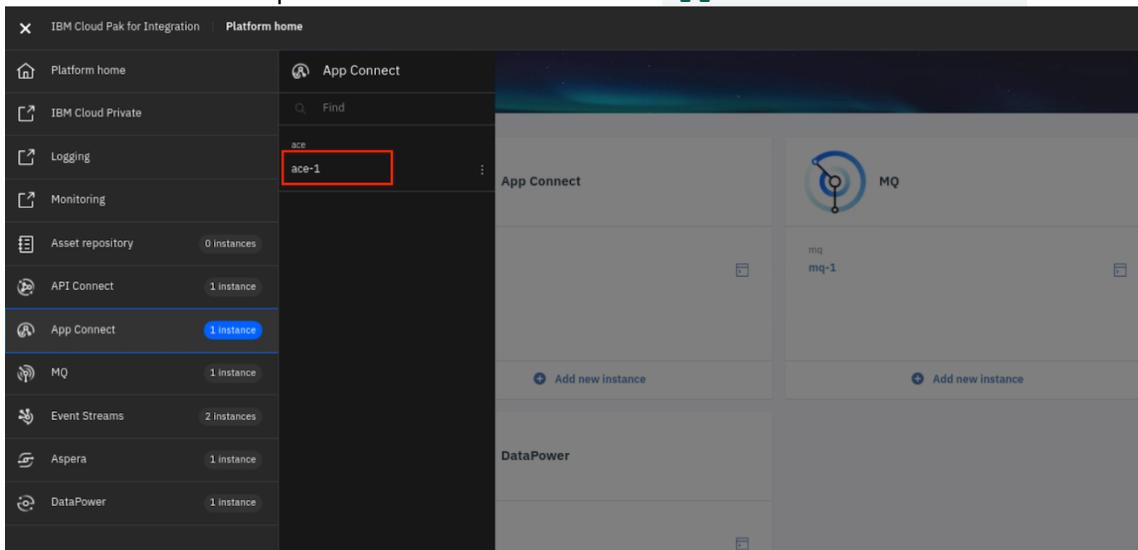
Task 5 -- Deploy the neworder integration flow

In this task you will deploy the orders integration flow as App Connect Enterprise containers running in Kubernetes on the Cloud Pak for Integration. You will then test the integration API by calling the API to create a new order and confirm the response payload and data in the queue.

1. In your browser, return to the [IBM Cloud Pak for Integration](#) bookmark. Click the hamburger menu icon in the upper-left.



2. In the drop-down menu. Select [App Connect > ace-1](#)



Note:

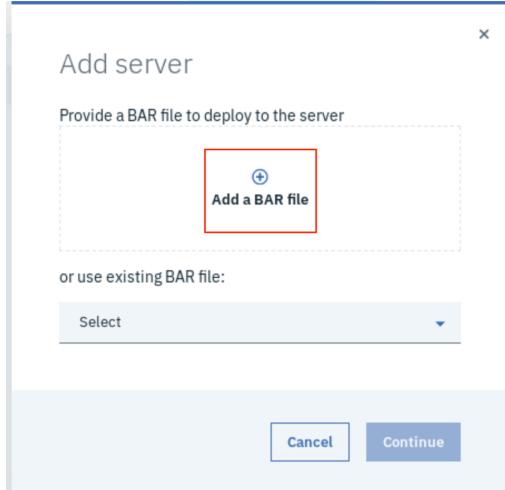
If you see the message "*ace-1 did not load correctly*", click [Open ace-1](#). This issue occurs when using a self-signed certificate such as in this demo environment. This will load ace-1 in a new tab. Accept the self-

sign certificate. Afterwards come back to the original tab and click **Try again**.

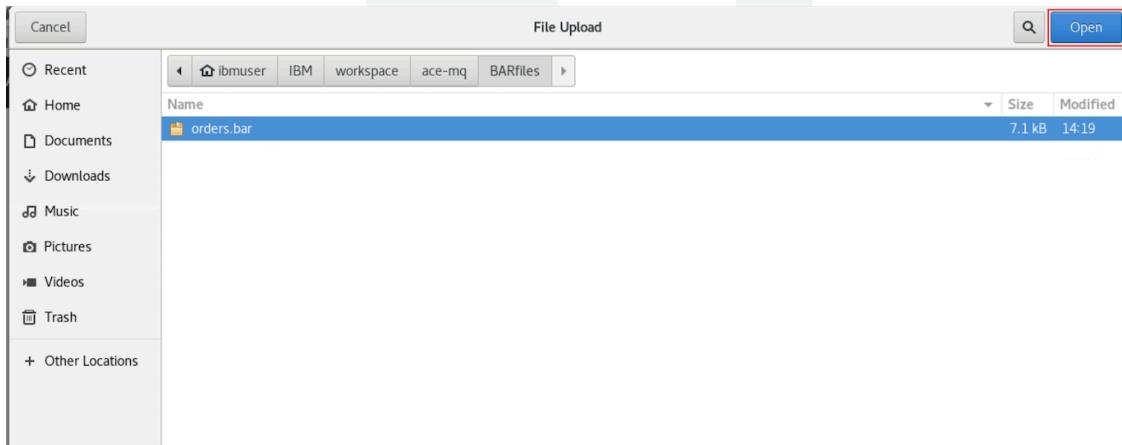
3. When you deploy a BAR file, you create a new instance of App Connect Enterprise. In the top right corner of the, click **Add Server**.



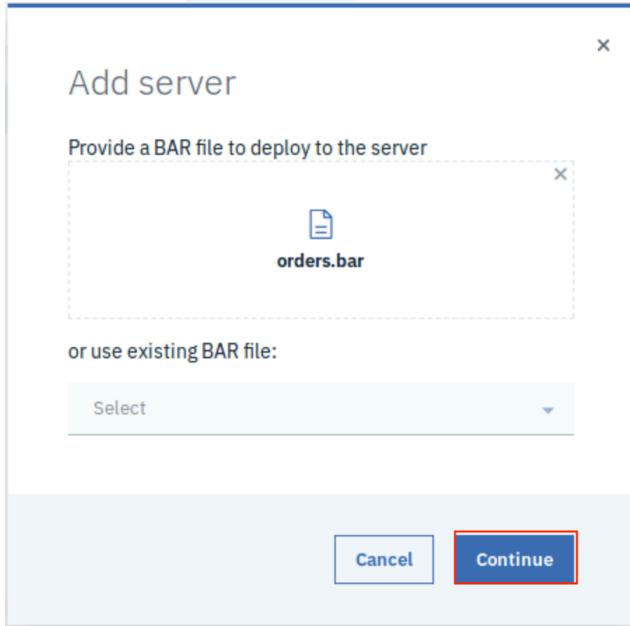
4. Click **Add a BAR file**.



5. Navigate to the `/home/ibmuser/IBM/workspace/ace-mq/BARfiles` directory. Find `orders.bar` and click **Open**.

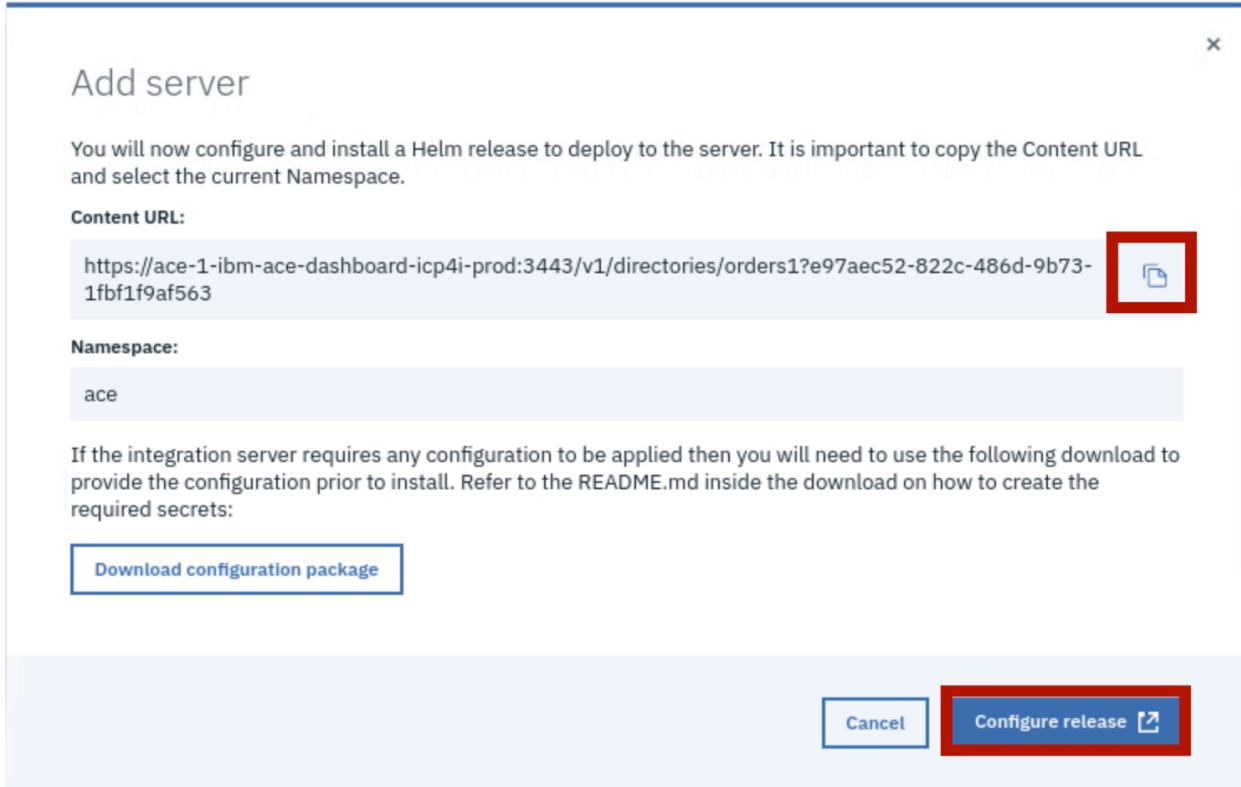


6. Click **Continue**.



7. A pop up with a content URL will appear. Click the clipboard icon to copy the Content URL. Click **Configure release**.

Note: Your URL will be different than the one displayed below.



8. This is the ACE installation server main page. Verify the ACE version. Click **Configure** in the bottom right or the **Configuration** tab at upper left to continue.

The screenshot shows the IBM Cloud Private interface with the cluster 'mycluster'. A specific chart named 'ibm-ace-server-icp4i-prod V 2.0.0' is selected. The 'Configuration' tab is highlighted with a red box. The page content includes:

- IBM APP CONNECT ENTERPRISE**: The chart icon and title.
- Introduction**: A brief description of IBM App Connect Enterprise.
- Chart Details**: Information about the chart's deployment.
- Prerequisites**: Requirements for deploying the chart.
- A prominent **Configure** button at the bottom right.

9. Configuring ACE

- 1) Helm Release name: **orders**
- 2) Target Namespace: **ace**
- 3) Check as Target Cluster: **local-cluster**
- 4) Check the **License agreement** checkbox.

The configuration page for the ACE helm chart includes the following sections:

- Configuration**: A warning about Pod Security Conflict and a warning about Pod Security Warning.
- Helm release name ***: The field contains 'orders' with circle 1 around it.
- Target namespace ***: The field contains 'ace' with circle 2 around it.
- Target Cluster **i***: The field contains 'local-cluster' with circle 3 around it. A dropdown menu shows 'local-cluster' selected.
- License ***: A checkbox labeled 'I have read and agreed to the License agreement' is checked with circle 4 around it.
- Pod Security**: A note about deploying correctly with a required Namespace.

10. Scroll down and expand the **Quick start** to open:
- 1) Content Server URL: paste the URL from step 7 above
 - 2) Type the Proxy Node IP: **icp-proxy.10.0.10.2.nip.io**

Parameters

To install this chart, additional configuration is needed in Quick start. To customize installation, view and edit All parameters.

Quick start
Required and recommended parameters to view and edit.

Content Server URL *
https://ace-1-ibm-ace-dashboard-icp4i-prod:3443/v1/directories/orders?d3e5cfe2-8dce-4467-8755-76f48c71b837

Service
Service settings

Proxy Node IP or FQDN *
icp-proxy.10.0.10.2.nip.io

> **All parameters**
Other required, optional, and read-only parameters to view and edit.

11. Scroll down and click **All parameters**:
- 1) Check the **Production Usage** checkbox.
 - 2) Check the **Local Default Queue Manager** checkbox

All parameters
Other required, optional, and read-only parameters to view and edit.

Content Server URL *
https://ace-1-ibm-ace-dashboard-icp4i-prod:3443/v1/directories/orders?d3e5cfe2-8dce-4467-8755-76f48c71b837

Production usage

Local default Queue Manager

Architecture scheduling preference ⓘ
amd64

Enable IBM App Connect Designer flows

12. Scroll down until Docker Image settings:

- 1) Change Image pull policy: **Always**
- 2) Type Image Pull Secret: **deployer-dockercfg-bnjx2**

Enable IBM App Connect Designer flows

Docker image
Specify the image to run. The image is selected based on whether MQ is enabled.

Docker image for ACE only *	Docker image for ACE with MQ *
docker-registry.default.svc:5000/ace/ibm-ace-server-prod	docker-registry.default.svc:5000/ace/ibm-ace-mq-server-prod
Configurator docker image *	Docker image for ACE with Designer flows *
docker-registry.default.svc:5000/ace/ibm-ace-icp-configurator-prod	docker-registry.default.svc:5000/ace/ibm-ace-designer-flows-prod
Image tag *	Image pull policy
11.0.0.5-amd64	Always 1
Image pull secret <small>?</small>	2
deployer-dockercfg-bnjx2	

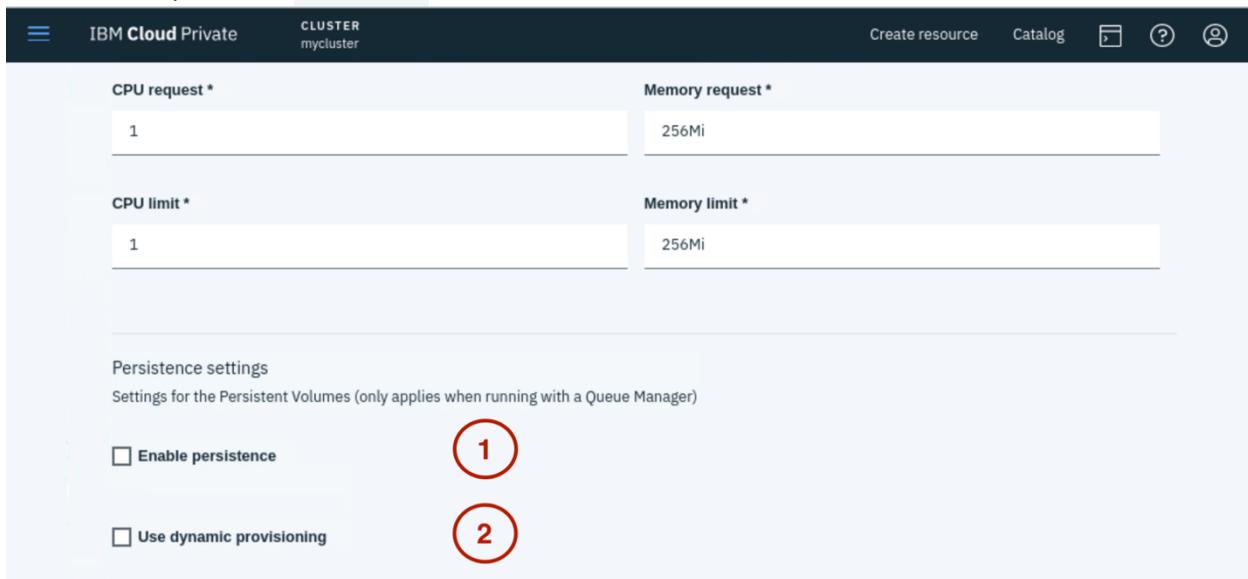
13. Scroll down and set **Replica Count** to 1.

Configuration for ACE (without MQ) deployments
Configuration settings for specifying required resources when running ACE without MQ

CPU request *	Memory request *
1	1024Mi
CPU limit *	Memory limit <small>?</small>
1	1024Mi
Replica count	
1	

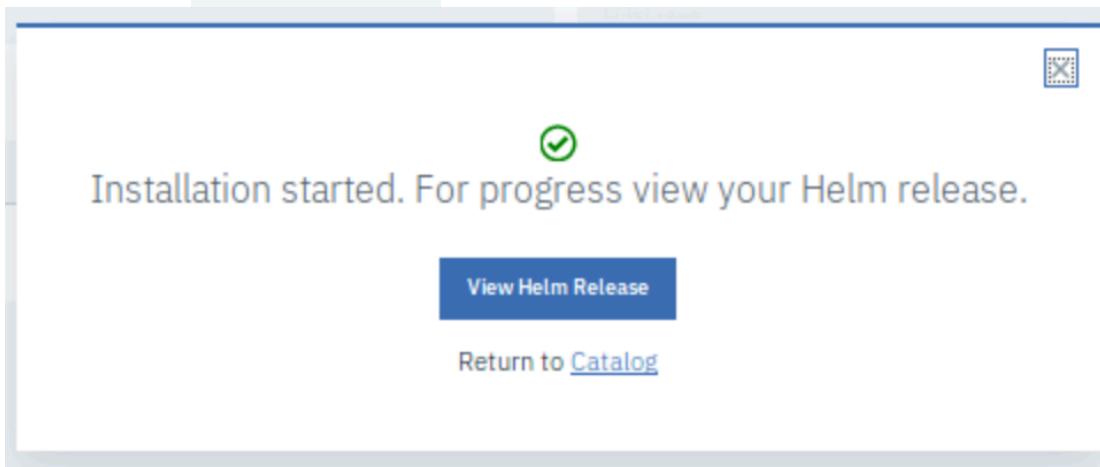
14. Scroll down to Persistence settings

- 1) Uncheck **Enable persistence**
- 2) Uncheck **Dynamic provisioning**
- 3) Click **Install**



The screenshot shows the IBM Cloud Private interface for creating a new resource. At the top, it says "IBM Cloud Private" and "CLUSTER mycluster". On the right, there are buttons for "Create resource", "Catalog", and help/space icons. Below this, there are sections for "CPU request *" (1), "Memory request *" (256Mi), "CPU limit *" (1), and "Memory limit *" (256Mi). A section titled "Persistence settings" follows, with a note about applying to Queue Managers. It contains two checkboxes: "Enable persistence" (circled in red) and "Use dynamic provisioning" (circled in red).

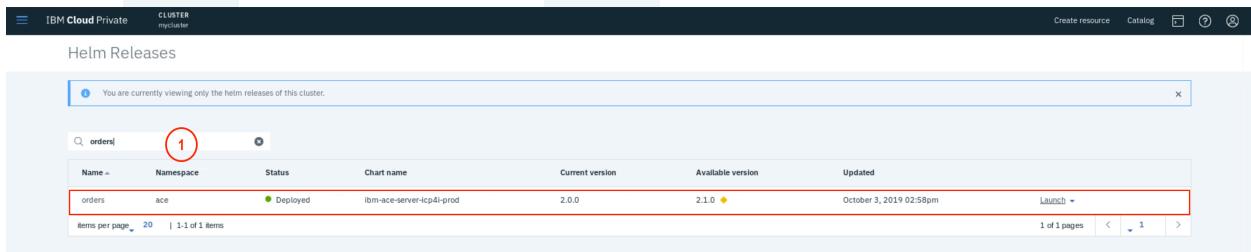
15. View the progress of the install via Helm Releases as prompted.
Click on **View Release**.



The screenshot shows a progress message for a Helm release. It features a green checkmark icon and the text "Installation started. For progress view your Helm release." Below this, there are two buttons: "View Helm Release" (in blue) and "Return to Catalog".

16. The ACE installation server may take some time to deploy.
(Approximately 5 mins) To verify the installation, go to IBM Cloud Private. Click the **Menu-->Workloads-->Helm Releases**.

17. You will see the Helm Releases. Type on search releases:
orders and click on the **orders** line.

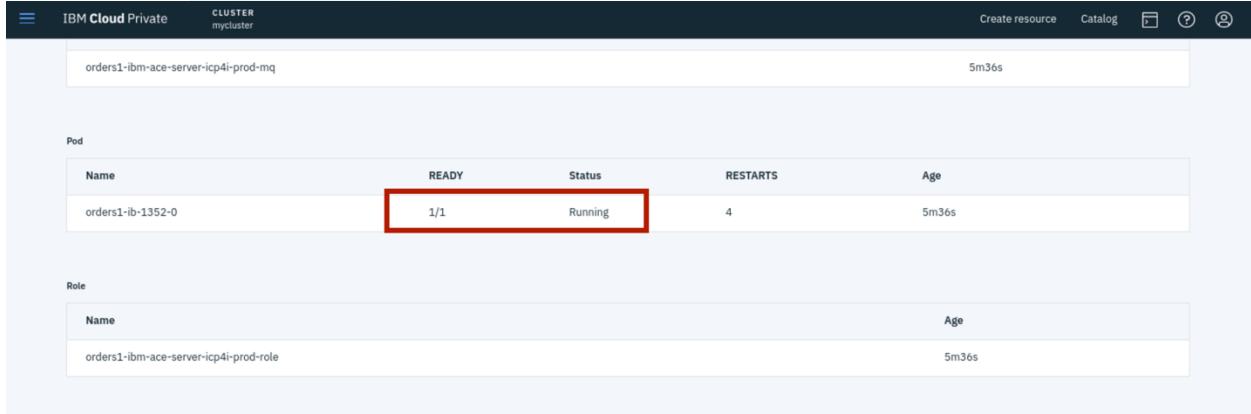


The screenshot shows the Helm Releases page in the IBM Cloud Private interface. A search bar at the top contains the text "orders". Below it is a table with the following data:

Name	Namespace	Status	Chart name	Current version	Available version	Updated
orders	ace	Deployed	ibm-ace-server-icp4i-prod	2.0.0	2.1.0	October 3, 2019 02:58pm

A red box highlights the entire table row for the 'orders' release. A circled '1' is positioned above the search bar.

18. Locate Pod services and check the Pod **Status:Running** and **Ready: 1/1**. Approximate deployment time 5 minutes.



The screenshot shows the Pod details page in the IBM Cloud Private interface. The pod name is "orders1-ibm-ace-server-icp4i-prod-mq". The pod status table is as follows:

Name	READY	Status	RESTARTS	Age
orders1-ib-1352-0	1/1	Running	4	5m36s

A red box highlights the "READY" column for the pod. Below the pod table is a role table:

Name	Age
orders1-ibm-ace-server-icp4i-prod-role	5m36s

19. Return to the Cloud Pak for Integration browser. Click **Done** to view the ACE server that was installed.

The screenshot shows the 'IBM Cloud Pak for Integration' browser window. In the center, a modal dialog titled 'Add server' is open. It contains fields for 'Content URL' (set to <https://ace-1-ibm-ace-dashboard-icp4i-prod:3443/v1/directories/orders?cee1e1e1-a7cd-408a-824e-11ca9df35918>) and 'Namespace' (set to 'ace'). Below these fields, a note states: 'If the integration server requires any configuration to be applied then you will need to use the following download to provide the configuration prior to install. Refer to the README.md inside the download on how to create the required secrets:'. At the bottom right of the dialog is a blue 'Done' button, which is highlighted with a red border.

20. You see the ACE Server running. Select the **orders** server. Click on the **neworder API**.

The image consists of two side-by-side screenshots of the 'IBM Cloud Pak for Integration' interface.

Left Screenshot: The title bar says 'IBM Cloud Pak for Integration | App Connect ace | ace-1'. Below it, a 'Servers' section lists one item: 'orders1' (Server (with MQ)). Below the server name, it says 'Release name: orders1 1/1 replicas' and 'Started'. This entire card is highlighted with a red box. An arrow points from the right side of this card to the right screenshot.

Right Screenshot: The title bar says 'IBM Cloud Pak for Integration | App Connect ace | ace-1'. Below it, the path 'Dashboard / Server: orders1' is shown. The 'orders1' server card is displayed, featuring a yellow icon, the name 'orders1', and the text 'Release name: orders1'. Below the card, the text 'neworder API' is also highlighted with a red box.

21. You will see the REST API base URL and OpenAPI document created by ACE. Click **OpenAPI document** to see the swagger json.

The screenshot shows the IBM Cloud Pak for Integration interface. In the top navigation bar, there are tabs for 'Dashboard', 'Server: orders', and 'API: neworder'. Below the navigation, there's a section titled 'neworder' with tabs for 'Documentation', 'Contents', 'Properties', and 'Other resources'. Under 'Documentation', the 'REST API Base URL' is listed as <http://icp-proxy.10.0.10.2.nip.io:31263/neworder/v1>. Next to it is the 'OpenAPI document' link, which is highlighted with a red border. Below these links is a search bar with the placeholder 'Search' and a dropdown menu showing the path '/[id]'. A status indicator 'Started' is visible in the top right corner.

22. Observe the json document created.

The screenshot shows a browser window displaying the generated OpenAPI JSON document. The URL in the address bar is <http://icp-proxy.10.0.10.2.nip.io:31263/neworder/v1/swagger.json>. The JSON content is as follows:

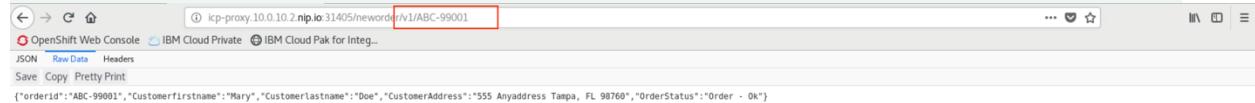
```
{
  "swagger": "2.0",
  "info": {
    "title": "neworder",
    "version": "1.0.0",
    "description": "neworder"
  },
  "paths": {
    "/[id)": {
      "get": {
        "operationId": "getId",
        "responses": {
          "200": {
            "description": "The operation was successful.",
            "schema": {
              "type": "string"
            }
          }
        },
        "produces": [
          "application/json"
        ],
        "parameters": [
          {
            "required": true,
            "name": "id",
            "in": "path",
            "type": "string"
          }
        ]
      }
    }
  },
  "basePath": "/neworder/v1",
  "definitions": {
    "order": {
      "type": "object",
      "properties": {
        "orderId": {
          "type": "string"
        },
        "Customerfirstname": {
          "type": "string"
        },
        "Customerlastname": {
          "type": "string"
        },
        "CustomerAddress": {
          "type": "string"
        },
        "OrderStatus": {
          "type": "string"
        }
      }
    }
  },
  "schemes": [
    "http"
  ]
}
```

23. To invoke the ACE API, click the **REST API Base URL** link.

The screenshot shows the IBM Cloud Pak for Integration interface again. The 'REST API Base URL' link is highlighted with a red border. The rest of the interface is identical to the previous screenshot, showing the 'neworder' API details and the search bar.

24. A new window browser will open. Complete the URL by appending: ABC-99001 after v1. Hit **Enter** to see the results.

<http://icp-proxy.10.0.10.2.nip.io:31405/neworder/v1/ABC-99001>



25. Check whether the order message has arrived in the **NEWORDER** queue. You can use MQ console. Open a new tab on your browser and click the **IBM Cloud Pak for Integration** bookmark then click **mq-1** link.

The screenshot shows the "Platform home" page of the IBM Cloud Pak for Integration interface. It lists several components:

- API Connect**: Instances: apic, apic-1. Action: Add new instance.
- App Connect**: Instances: ace, ace-1. Action: Add new instance.
- MQ**: Instances: mq, mq-1. Action: Add new instance.
- Event Streams**: Instances: eventstreams, es-1. Action: Add new instance.
- Aspera**: Instances: aspera, aspera-1. Action: Add new instance.
- DataPower**: Instances: datapower, dp-1. Action: Add new instance.

A red arrow points from the text "click the **IBM Cloud Pak for Integration** bookmark then click **mq-1** link." to the "mq-1" entry in the MQ section.

26. You will see a new message has arrived in the **NEWORDER** queue.

The screenshot shows a table titled "Queues on mq". The columns are "Name", "Queue type", and "Queue depth". There are two rows. The first row is for a queue named "AMQ.5D95E32E24C63005" which is of type "Local" and has a queue depth of "0". The second row is for a queue named "NEWORDER" which is also of type "Local" and has a queue depth of "1". A red box highlights the "1" in the "Queue depth" column for the "NEWORDER" row. At the bottom of the table, it says "Total: 2" and "Last updated: 3:44:09 PM".

Name	Queue type	Queue depth
AMQ.5D95E32E24C63005	Local	0
NEWORDER	Local	1

Total: 2 Last updated: 3:44:09 PM

Task 6 -- Delete a BAR file on the Cloud Pak for Integration

If you want to delete an ACE instance, follow these steps. First delete the Helm release located in IBM Cloud Private, then delete the BAR file located in ACE.

1. Click on the **IBM Cloud Private** bookmark in your browser. From the main menu choose **Workloads > Helm Releases**

The screenshot shows the IBM Cloud Private interface. At the top, there is a navigation bar with links for Apps, OpenShift Web..., IBM Cloud Pak f..., and IBM Cloud Priva... On the left, there is a sidebar with several menu items: Overview, Cluster Services, Configuration, Manage, Platform, and Workloads. The Workloads item is expanded, and Helm Releases is selected, with a red box highlighting it. The main content area has a large blue banner with the text "to IBM Cloud Private". Below the banner, there is descriptive text about the platform being built on Kubernetes and its integration capabilities. A sidebar on the right lists Brokered Services, DaemonSets, Deployments, and Jobs. The Helm Releases section contains a detailed description of management services and their integration with other tools. A blue footer bar at the bottom encourages users to check out reference links and look around IBM Cloud.

2. On the search bar, type **orders**. Click on the ellipsis(. . .) and click **Delete**

Name	Namespace	Status	Chart name	Current version	Available version	Updated	Launch	...
orders	ace	● Deployed	ibm-ace-server-icp4i-prod	2.0.0	2.1.0	October 5, 2019 09:48am	Launch	Delete

3. Click on the **IBM Cloud Pak for Integration** bookmark. Choose the **ace-1** link.

API Connect	App Connect	MQ
apic apic-1	ace ace-1	mq mq-1
Add new instance	Add new instance	Add new instance

Event Streams	Aspera	DataPower
eventstreams es-1	aspera aspera-1	datapower dp-1
Add new instance	Add new instance	Add new instance

4. On the **Servers** dashboard in the top right corner, click the settings icon. Choose **Bar files** from the drop-down menu.

5. Select which BAR files you wish to delete. Click the ellipsis (...).
Click **Delete BAR**. Make sure the BAR file is not deployed.

The screenshot shows a table with two columns: 'Name' and 'Deployed to'. The 'Name' column contains 'orders'. The 'Deployed to' column contains 'Not deployed'. To the right of the table is a three-dot menu icon. A red box surrounds this icon. A larger red box surrounds the 'Delete BAR...' option in the dropdown menu that appears when the icon is clicked.

Summary

Congratulations! You've completed this tutorial. In this lab you learned how to:

- Increase efficiency by creating, testing, and debugging an integration flow with a message queue within a single, unified experience
- Increase scale by deploying integration flows and message queues as containers on Kubernetes.
- Utilize Helm, the industry standard for Kubernetes package manager, to deploy the integration.

To try out more labs, go to [Cloud Pak for Integration Demos](#). For more information about the Cloud Pak for Integration, go to <https://www.ibm.com/cloud/cloud-pak-for-integration>.