Traditional MQ/IIB

Based ESB

On

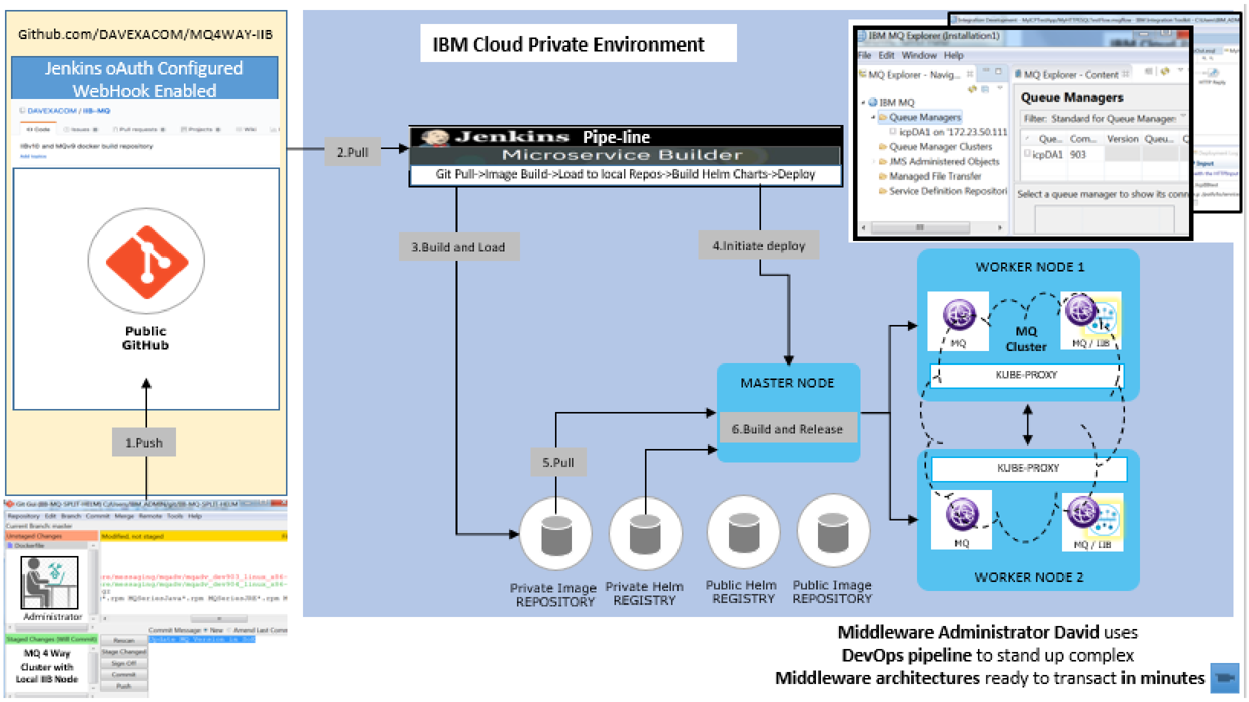
IBM Cloud Private

Contents

Introduction

This setup and demo guide explores delivering a complex integration and messaging pattern on IBM Cloud Private using Helm, Github and Microclimate/Jenkins DevOps pipelines.

Overview diagram - A Traditional ESB pattern on IBM Cloud Private



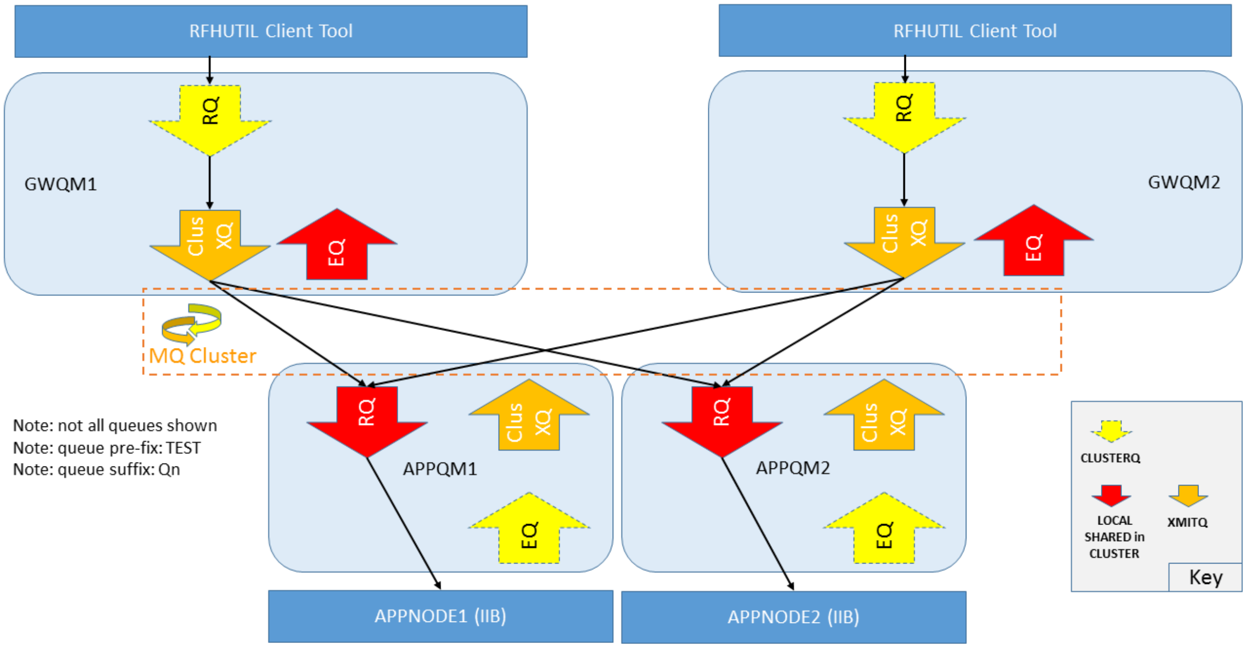
Middleware Administrator David uses
DevOps pipeline to stand up complex
Middleware architectures ready to transact **in minutes**

Scenario Description

The scenario shown above explores using a single Helm Release on IBM Cloud Private to deploy a 4 Way MQ cluster with 2 active/active Gateway queue managers and active/active back office application queue managers each with a locally connected IBM Integration Bus node. The docker images and helm charts are stored on Public Github and the ICP Devops tooling, Micro Climate (Jenkins) is used to build the images and deploy the Helm Release. This 4 container deployment initializes on ICP with all IBM MQ channels resolved and started plus IBM Integration Bus message flows deployed, started and ready to service messages for back office.

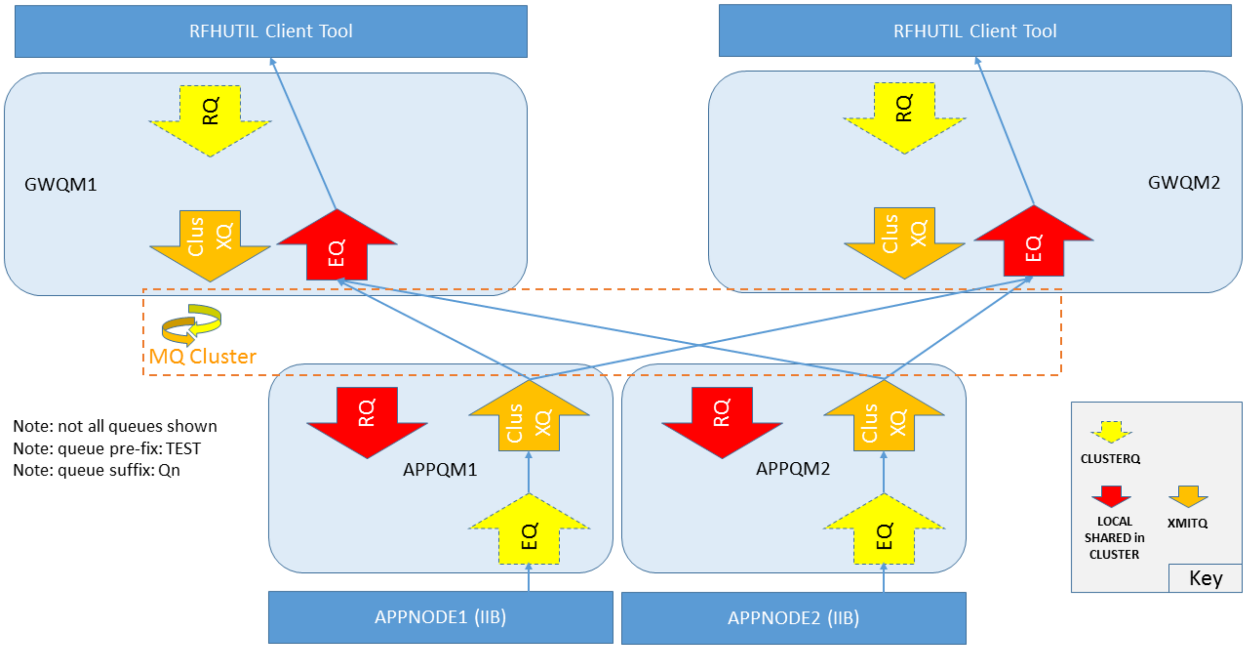
The following two diagrams show how the MQ Cluster and IIB nodes process inbound (receipt) and outbound (emission) messages.

MQ cluster – Receipt Processing (inbound to back office)



Note: not all queues shown
Note: queue pre-fix: TEST
Note: queue suffix: Qn

MQ cluster – Emission Processing (Outbound from Back Office)



Note: not all queues shown
Note: queue pre-fix: TEST
Note: queue suffix: Qn

Supporting Collateral on Github

Helm Release Repository for 4-Way MQ Cluster with Local IIB

https://github.com/DAVEXACOM/MQ4WAYCLUS-LCIIB

The MQ4WAYCLUS-LCIIB repository delivers:

- The MQ Docker Image build used by GWQM1 and GWQM2 containers
- Helm charts for the full GWQM1 (MQ only), GWQM2 (MQ Only), APPQM1(MQ with IIB) and APPQM2(MQ with IIB) helm release.

The Helm release relies on two Github repositories because we have two different images to deploy as part of the release. Two containers running the MQ Only Image and two containers running the MQ with IIB image. Therefore both images will need to be build and pushed to the ICP image repository

Container Build, Load (optional Helm release) IIB and MQ container

https://github.com/DAVEXACOM/IIB-MQ-SPLIT-HELM

The IIB-MQ-SPLIT-HELM repository delivers the MQ+IIB Docker Image build, which is used by APPQM1 and APPQM2 in the helm charts in MQ4WAYCLUS-LCIIB.

IIB-MQ-SPLIT-HELM does container helm charts such that this repository can also be used standalone to deliver a single MQ+IIB container build that will be leveraged by the Helm charts in MQ4WAYCLUS-LCIIB. Therefore, we must ensure it is built and available in the ICP image repository before initiating the Helm release MQ4WAYCLUS-LCIIB.

Pre-Req work – Customization of Microclimate/Jenkins build scripts on ICP

Documentation

The following documentation for this pre-req work is kept up to date here:

https://github.com/cloudnativedemo/icp-notes/blob/master/microclimate_notes.md

Customization Instruction guide

1. Create pipeline deployment namespace

```
kubectl create namespace microclimate-pipeline-deployments
```

2. Edit ClusterImagePolicy

```
kubectl edit clusterimagepolicies ibmcloud-default-cluster-image-policy
```

To add the following:

```
  - name: mycluster.icp:8500:*
  - name: docker.io/maven:*
  - name: docker.io/lachlanevenson/k8s-helm:*
  - name: docker.io/jenkins/*
```

3. Create Docker Registry secret to microclimate namespace

```
kubectl create secret docker-registry microclimate-registry-secret \
  --docker-server=mycluster.icp:8500 \
  --docker-username=admin \
  --docker-password=admin \
  --docker-email=null
```

4. Initialise Helm and login

```
helm init --client-only --skip-refresh
cloudctl login -a https://mycluster.icp:8443 -u admin -p admin -c id-mycluster-account -n
default --skip-ssl-validation
```

5. Create Helm secret

```
export HELM_HOME=$HOME/.helm
kubectl create secret generic microclimate-helm-secret --from-
file=cert.pem=$HELM_HOME/cert.pem --from-file=ca.pem=$HELM_HOME/ca.pem --from-
file=key.pem=$HELM_HOME/key.pem
```

6. Create Docker Registry secret for ***`microclimate-pipeline-deployments`*** *namespace*

```
kubectl create secret docker-registry microclimate-pipeline-secret \
  --docker-server=mycluster.icp:8500 \
  --docker-username=admin \
  --docker-password=admin \
  --docker-email=null \
  --namespace=microclimate-pipeline-deployments
```

7. Update ImagePullSecret for **`microclimate-pipeline-deployments`** *namespace*

```
kubectl patch serviceaccount default --namespace microclimate-pipeline-deployments -p
'{"imagePullSecrets": [{"name": "microclimate-pipeline-secret"}]}'
```

8. Customise Jenkins library

By default, the Jenkins library parameter is pointing to https://github.com/microclimate-dev2ops/jenkins-library This Jenkins library was a part of the Microclimate DevOps process. When a pipeline is created within a project in Microclimate, microclimate will create a Jenkins pipeline. The pipeline uses this library to .. 1. Pull the code from github repo . .. 2. Build a Docker image based on a Dockerfile found in the repo . .. 3. Authenticate and push the image into ICP's private registry . .. 4. Notify Microclimate to move to the next stage (e.g. deploy) . .. 5. Microclimate 'helm deploy' the helm chart found in the repo (by default it's under the /chart directory) .

Unfortunately, Microclimate only deploy it's supported project types e.g. Swift, NodeJS, Java/Liberty or Springboot. The easiest way to address this limitation is to fork and update the Jenkins library and inject the 'helm deploy' scriptlet onto step 4 (line 400 of microserviceBuilderPipeline.groovy)

```
            container ('helm') {
            echo "Attempting to deploy the test release"
            def deployCommand = "helm install ${realChartFolder} --values pipeline.yaml --
namespace ${namespace} --name ${helmRelease}"
            if (fileExists("chart/overrides.yaml")) {
              deployCommand += " --values chart/overrides.yaml"
            }
            if (helmSecret) {
              echo "Adding --tls to your deploy command"
              deployCommand += helmTlsOptions
            }
            testDeployAttempt = sh(script: "${deployCommand} > deploy_attempt.txt",
returnStatus: true)
            if (testDeployAttempt != 0) {
              echo "Warning, did not deploy the test release into the test namespace
successfully, error code is: ${testDeployAttempt}"
              echo "This build will be marked as a failure: halting after the deletion of the
test namespace."
            }
            printFromFile("deploy_attempt.txt")
         }
```

- ***Note:*** in my deployCommand, I've created one new variable `${helmRelease}`. The variable is defined on the top of the script (line 56 of the microserviceBuilderPipeline.groovy). Alternatively, you can just reuse `${image}` as your helm release name .

```
  def helmRelease = (config.releaseName ?: config.image ?: "").trim()
```

- My forked updated Jenkins library repo can be found here .

9. Deploy Microclimate helm chart

Via Helm command line

- **Add ibm-charts Helm repo**

```
helm repo add ibm-charts https://raw.githubusercontent.com/IBM/charts/master/repo/stable/
```

- **Deploy microclimate Helm chart**

```
helm install --name microclimate --namespace <target namespace> --set
global.rbac.serviceAccountName=micro-sa,jenkins.rbac.serviceAccountName=pipeline-
sa,global.ingressDomain=172.23.52.247.nip.io,jenkins.Pipeline.Template.RepositoryUrl=https://g
ithub.com/cloudnativedemo/jenkins-library.git,jenkins.Pipeline.Template.Version=master ibm-
charts/ibm-microclimate --tls
```

Note: Replace <172.23.52.247> with your <PROXY_IP>

Via ICP catalog

- Select ibm-microclimate from ICP catalog > click `Configure`
- Provide values for the following parameters:
  - `Helm release name`: your-microclimate-release-name
  - `Namespace`: default (or your preferred namespace)
  - `Microclimate hostname`: microclimate.172.23.52.247.nip.io (replace with your <microclimate.PROXY_IP.nip.io> or your own hostname)
  - Ensure that you've already created Persistent Volumes for Microclimate and Jenkins
  - `Service account name for Portal`: micro-sa
  - `Jenkins library repository`: https://github.com/cloudnativedemo/jenkins-library.git
  - `Jenkins hostname`: jenkins.172.23.52.247.nip.io (replace with your <jenkins.PROXY_IP.nip.io> or your own hostname)
  - `Service account name`: pipeline-sa
  - Click `deploy`

Create a project in Microclimate and Deploy

Once the Microclimate helm deployment completed, you can start to deploy your custom project

1. Make sure that your project contains a Dockerfile, Jenkinsfile and a chart directory (for helm chart)

2. Launch Microclimate (https://microclimate.172.23.52.247.nip.io - replace with your own microclimate hostname) and accept licensing agreement (for first launch only)

3. Select Projects > Click `New Project`

4. Select `Java` project type and provide a `project name` > click `Next`

5. Select `Microprofile/J2EE` and keep default value for `Context root` > click `Create`

6. Once the project is created, select `Pipeline` on the left menu

7. Click `Create pipeline`, and provide name and github repo of the pipeline > click `Create pipeline` to create a Jenkins pipeline

8. Switch to Jenkins (https://jenkins.172.23.52.247.nip.io - replace with your Jenkins hostname) to see if the pipeline has been created and built (refer to the troubleshooting section below if you have to wait for too long)

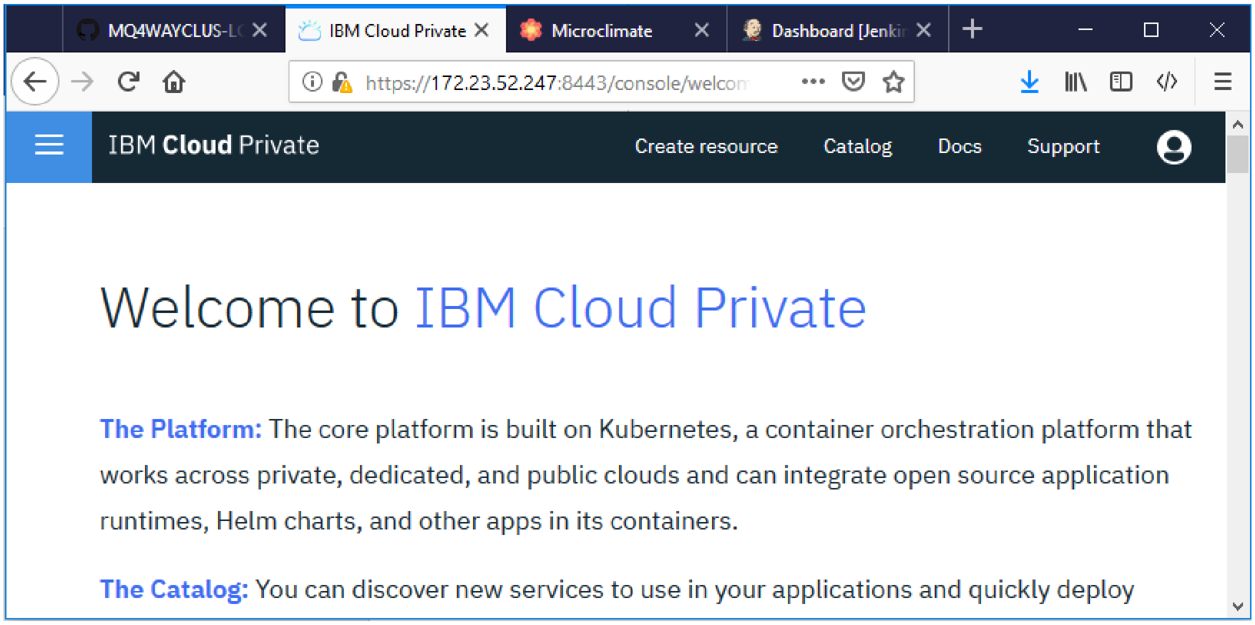
Troubleshooting

- When your Jenkins pipeline keeps looking for an executor for too long, there's probably an error occurred within your Jenkins containers. To identify the issue:
  - Identify the Jenkins pod name: `kubectl get pods -n <NAME_SPACE_WHERE_JENKINS_INSTALLED> | grep jenkins`
  - View the log: `kubectl log -n <NAME_SPACE_WHERE_JENKINS_INSTALLED> <JENKINS_POD_NAME> -f`
  - Most of the case I found caused by cluster image policy is not defined, you might need to update the default `clusterimagepolicy`
    - `kubectl edit clusterimagepolicies ibmcloud-default-cluster-image-policy`

Setting up the ESB on ICP Demo – Part 1 the MQ and IIB Image

Create Microclimate project to build the MQ with IIB image

Log into IBM Cloud Private

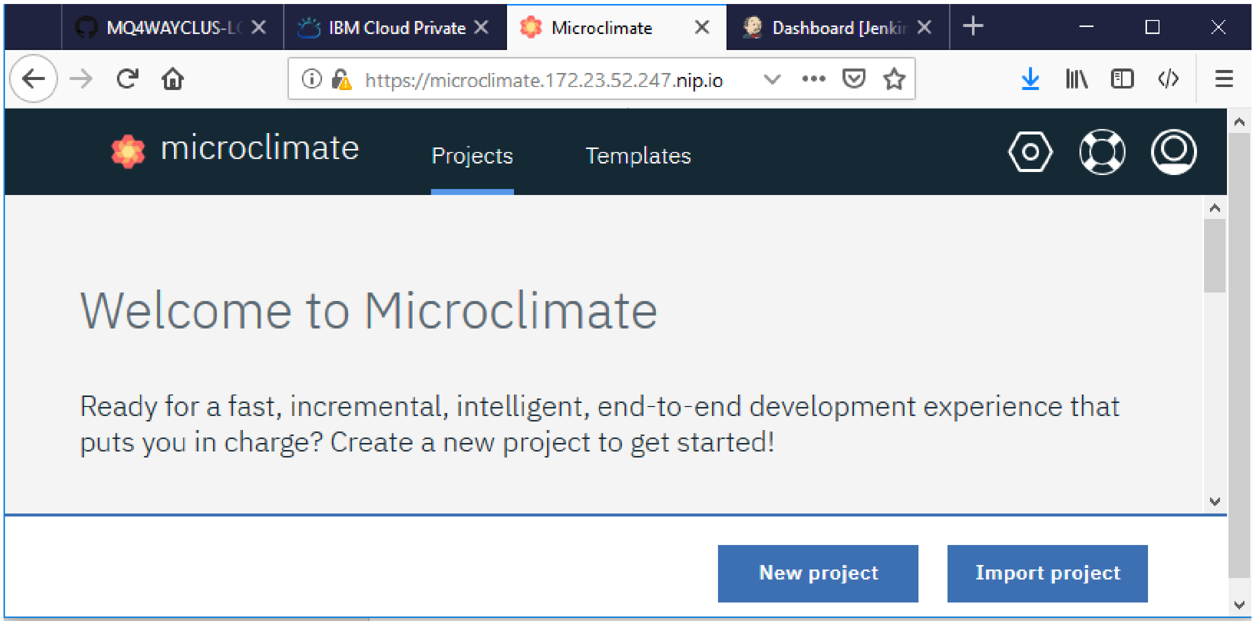


Welcome to IBM Cloud Private

The Platform: The core platform is built on Kubernetes, a container orchestration platform that works across private, dedicated, and public clouds and can integrate open source application runtimes, Helm charts, and other apps in its containers.

The Catalog: You can discover new services to use in your applications and quickly deploy

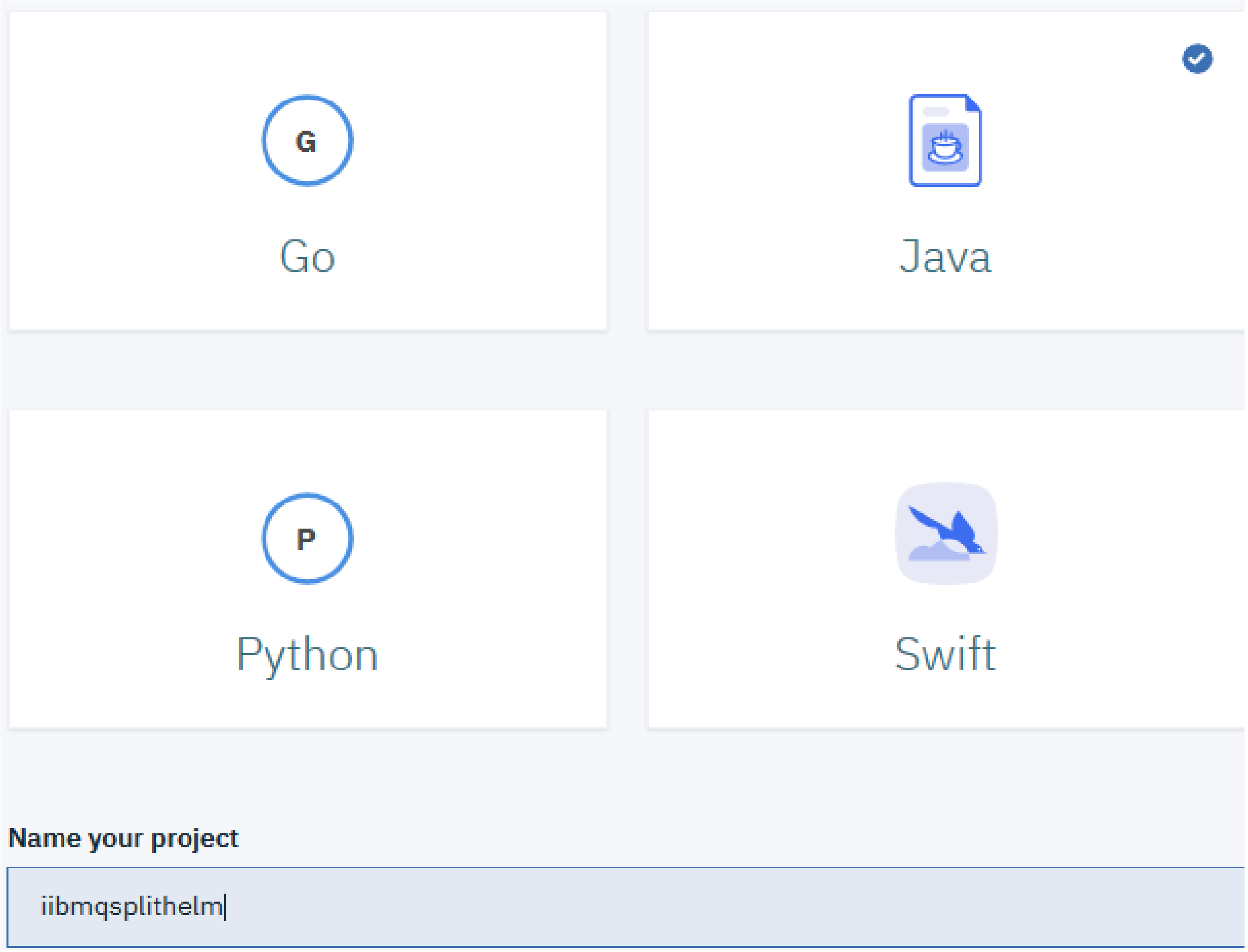
Connect to the Microclimate service https://microclimate.ICPipAddress.nip.io



Welcome to Microclimate

Ready for a fast, incremental, intelligent, end-to-end development experience that puts you in charge? Create a new project to get started!

Create a new Java project and name it to represent the github repository you wish to build from



Go

Java ✓

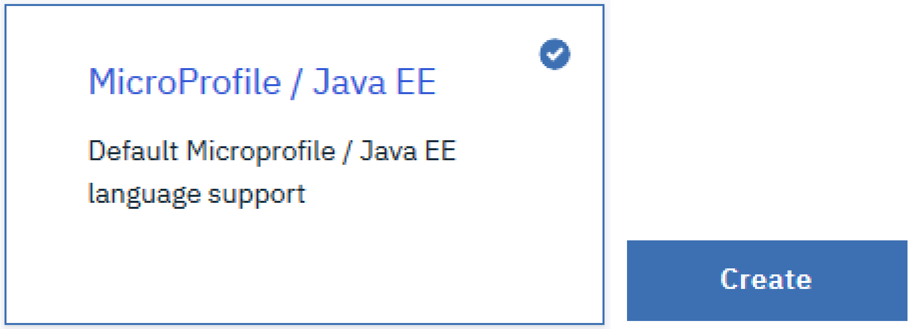
Python

Swift

Name your project

iibmqsplithelm

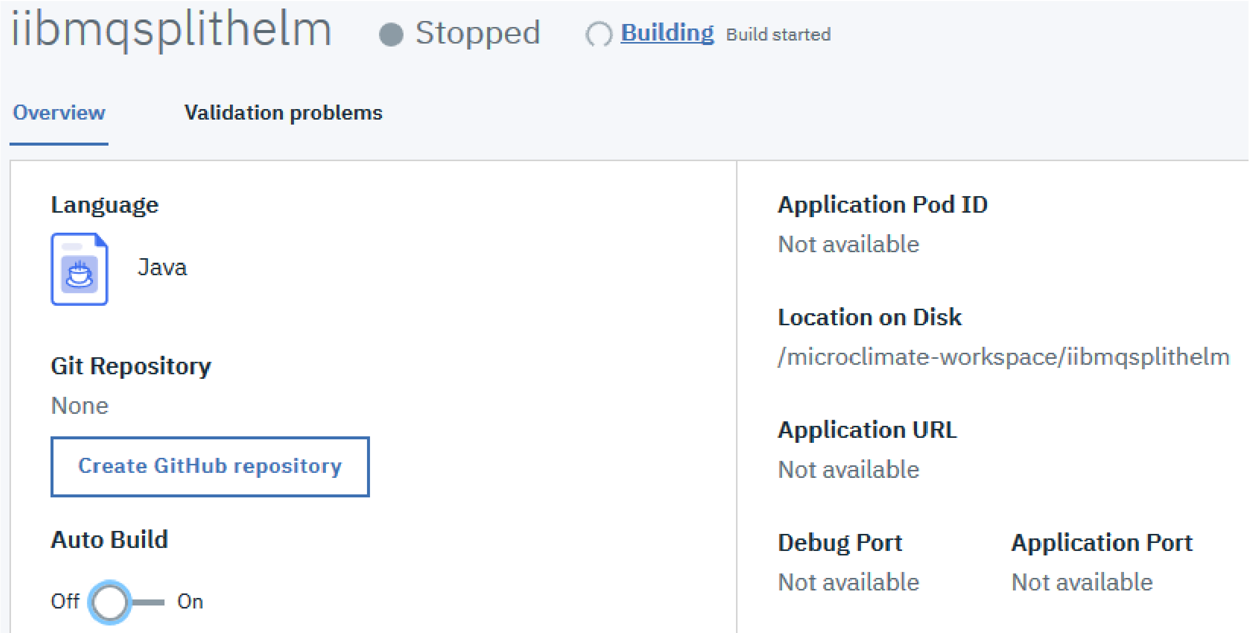
Use the MicroProfile / Java EE (IIB/MQ will work fine with these settings) and hit create.



MicroProfile / Java EE ✓

Default Microprofile / Java EE language support

Turn Auto Build to OFF. We will primarily interact with the Jenkins pipeline directly.



iibmqsplithelm ● Stopped Building Build started

Language
Java

Git Repository
None

Auto Build
Off On

Application Pod ID
Not available

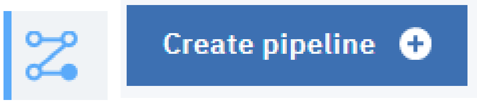
Location on Disk
/microclimate-workspace/iibmqsplithelm

Application URL
Not available

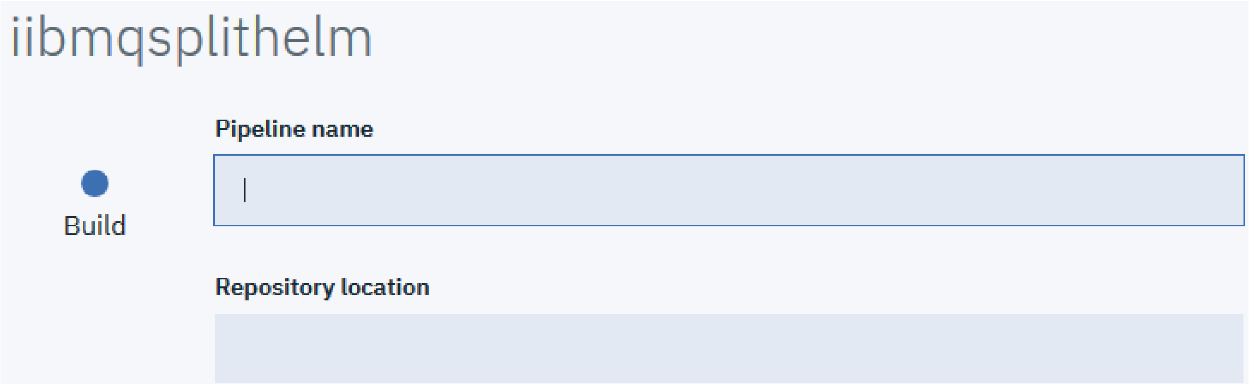
Debug Port
Not available

Application Port
Not available

Select the Pipeline icon and then select Create pipeline



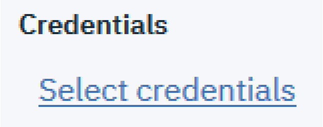
Leave the pipeline name and repos location blank for now and click on Select credentials



iibmqsplithelm

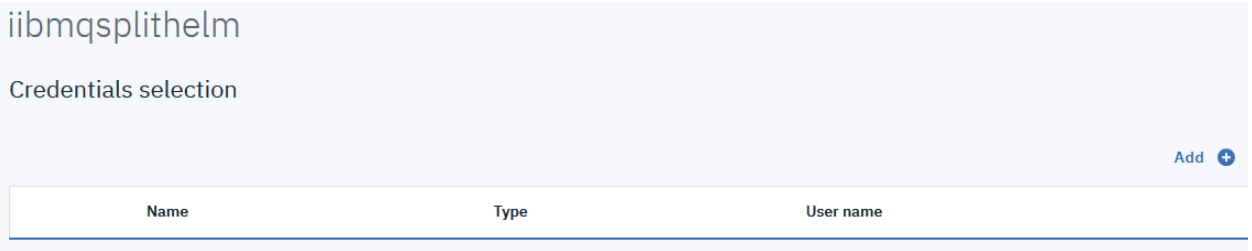
Pipeline name

Repository location



Credentials

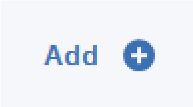
If this is your first project you will need to hit Add and add credentials



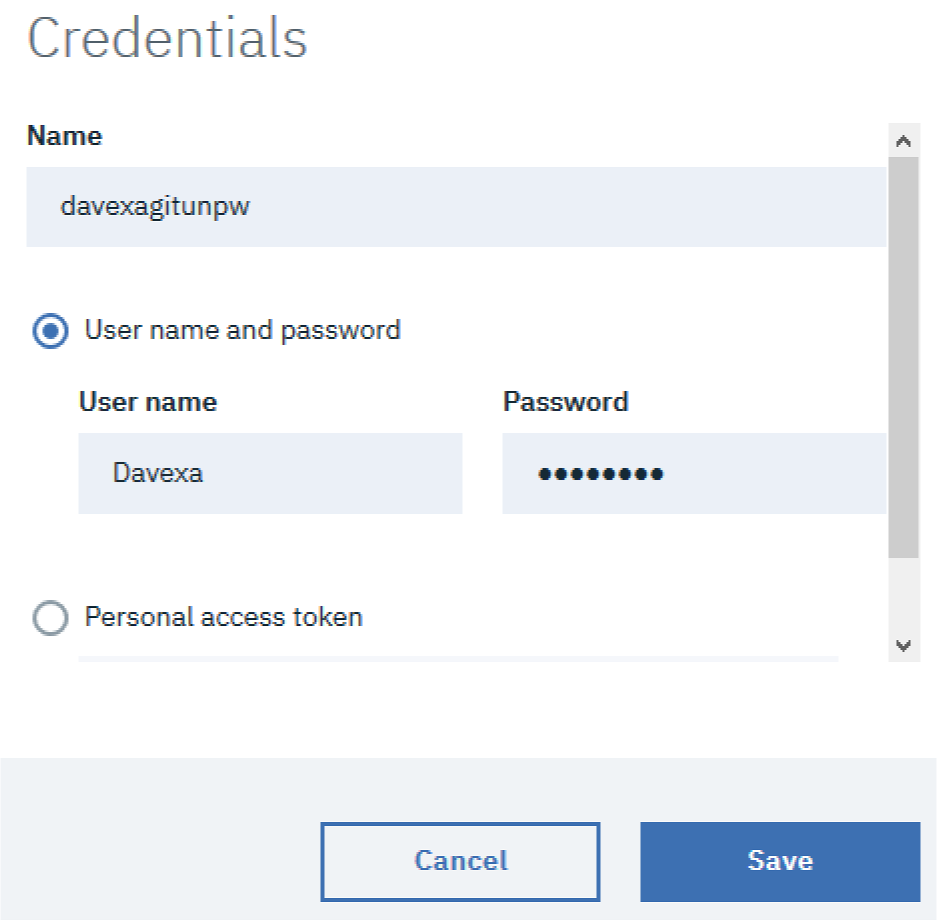
iibmqsplithelm

Credentials selection

Name	Type	User name



Enter your Github User name and password or leverage a Personal Access Token from github



Credentials

Name

davexagitunpw

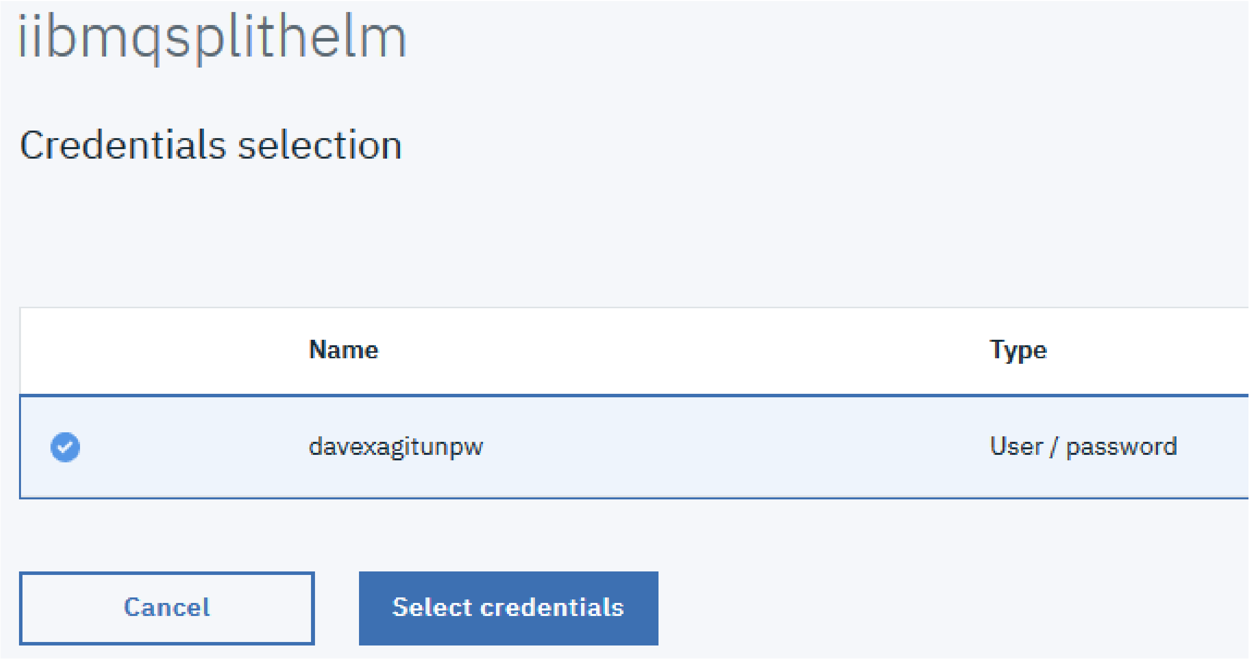
● User name and password

User name: Davexa

Password: ••••••••

○ Personal access token

Save the credentials. You can then click on them and hit select.

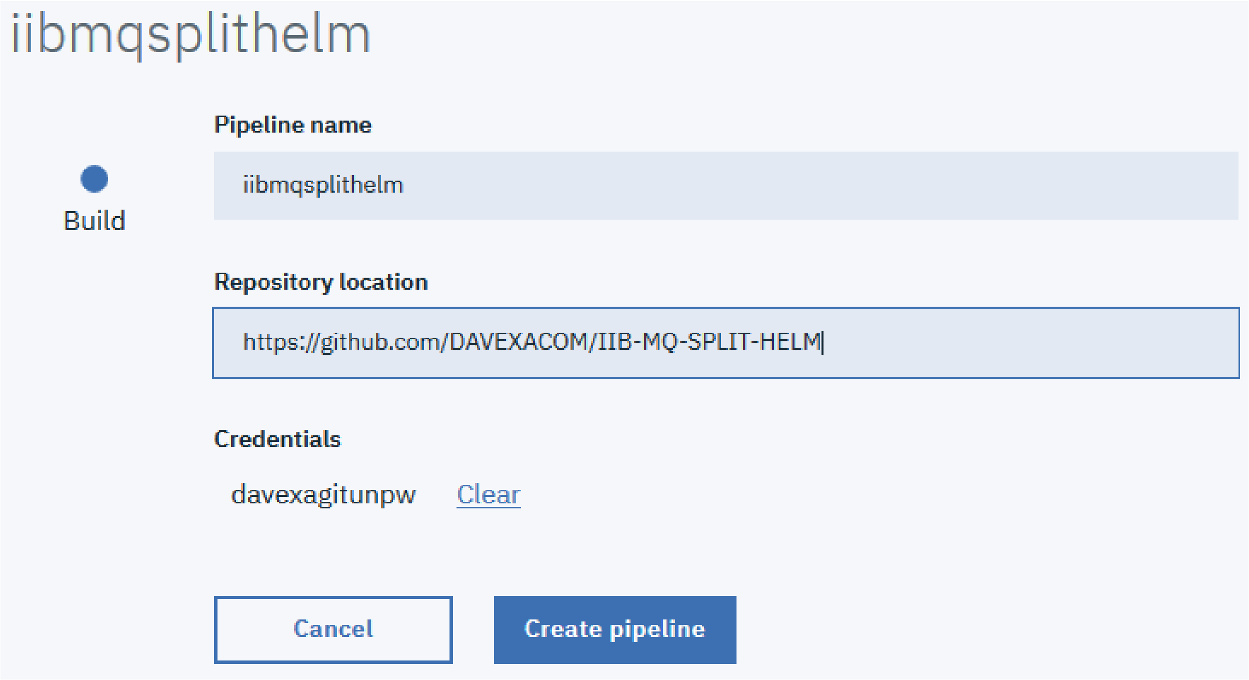


iibmqsplithelm

Credentials selection

	Name	Type
✓	davexagitunpw	User / password

Next add a pipeline name and repos location



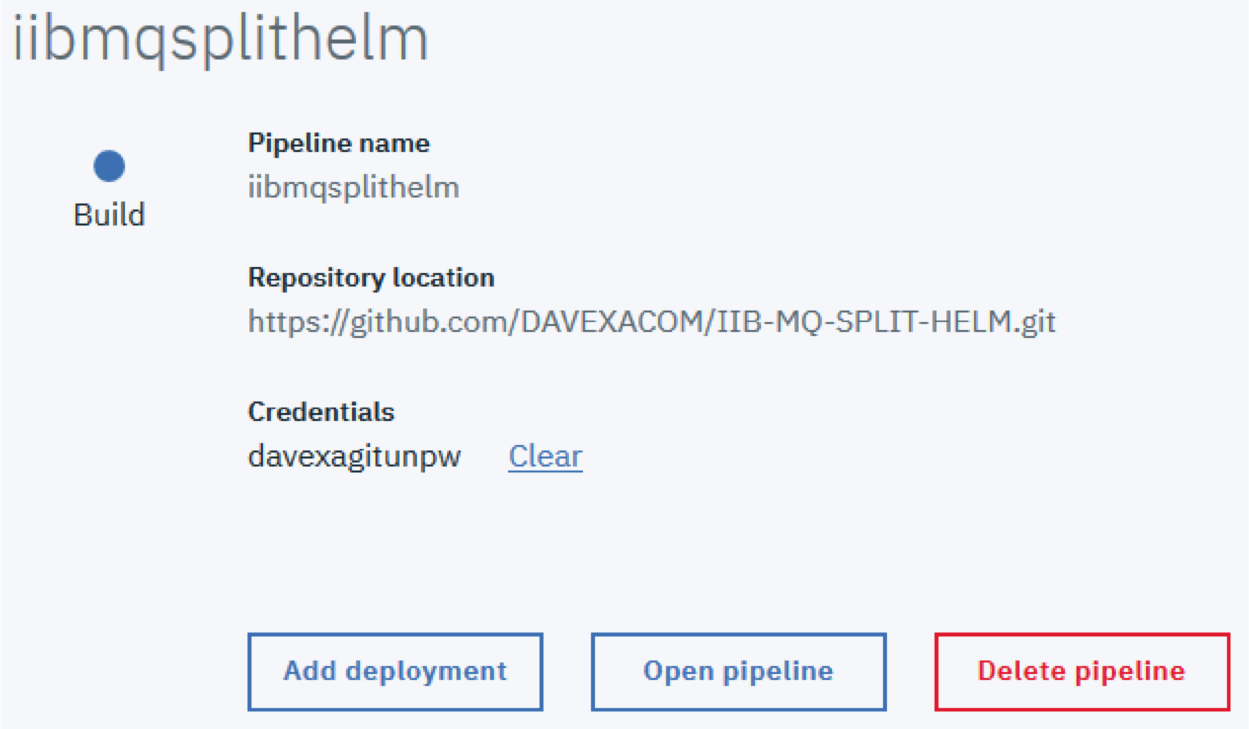
iibmqsplithelm

Pipeline name: iibmqsplithelm

Repository location: https://github.com/DAVEXACOM/IIB-MQ-SPLIT-HELM

Credentials: davexagitunpw

Hit Create pipeline



iibmqsplithelm

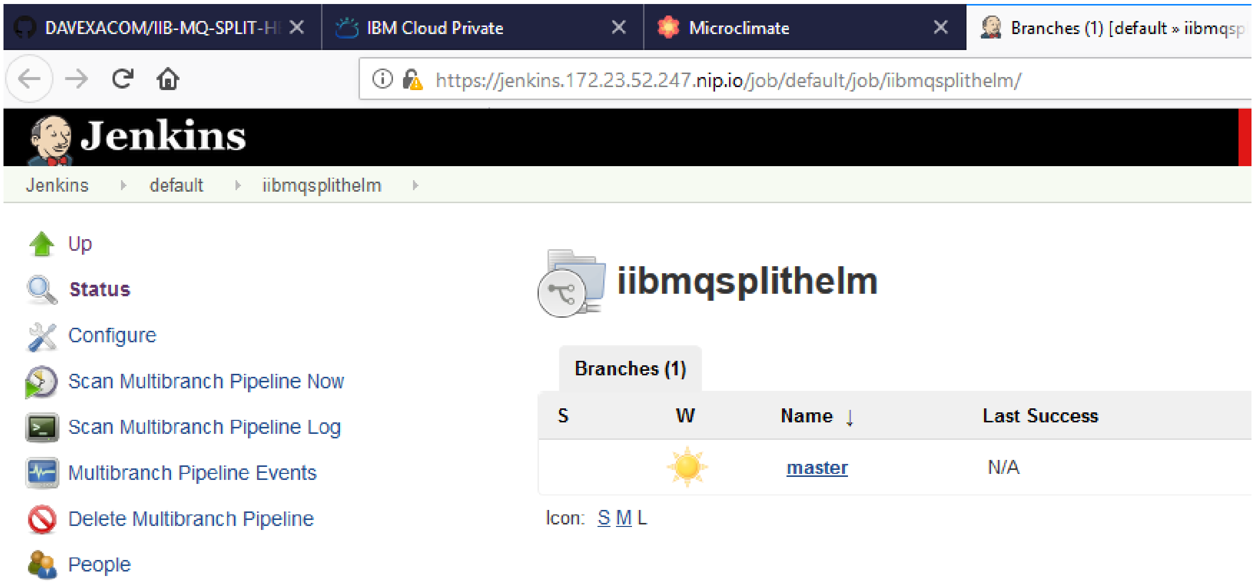
● Build

Pipeline name
iibmqsplithelm

Repository location
https://github.com/DAVEXACOM/IIB-MQ-SPLIT-HELM.git

Credentials
davexagitunpw Clear

Hit open pipeline and Jenkins will be launched and opened in a new browser window



DAVEXACOM/IIB-MQ-SPLIT-H
IBM Cloud Private
Microclimate
Branches (1) [default » iibmqsp
https://jenkins.172.23.52.247.nip.io/job/default/job/iibmqsplithelm/
Jenkins
Jenkins › default › iibmqsplithelm ›
Up
Status
Configure
Scan Multibranch Pipeline Now
Scan Multibranch Pipeline Log
Multibranch Pipeline Events
Delete Multibranch Pipeline
People
iibmqsplithelm
Branches (1)
Icon: S M L

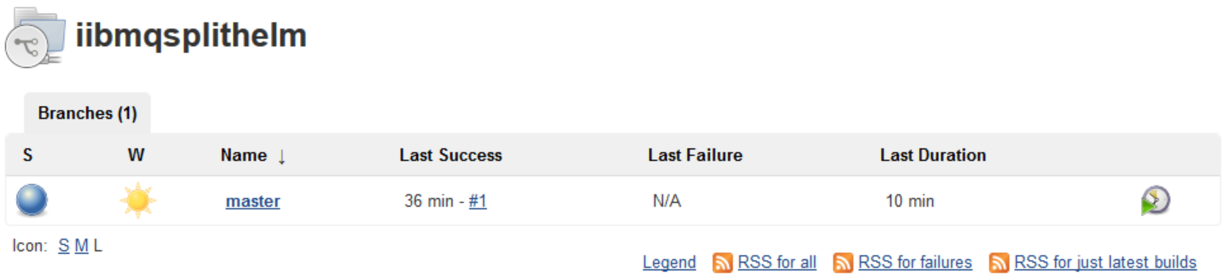
S	W	Name ↓	Last Success
	☀	master	N/A

The Jenkins pipeline is created and the first build is initiated

Microclimate project/Jenkins pipeline builds the MQ with IIB image

Following the Jenkins build process as it happens

The following screen shot is a capture of the Jenkins pipeline after the first initiated build is finished. This screen shot is for your reference only. If you want to start subsequent builds after the initial build is finished. For example you make changes to files in the github repository you can start a new build by hitting the circle with the green triangle on it. **You do not need to hit this button at this point**



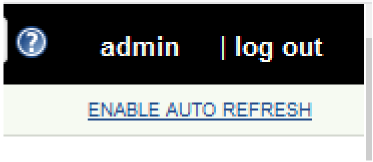
iibmqsplithelm

Branches (1)

S	W	Name ↓	Last Success	Last Failure	Last Duration	
●	☀	master	36 min - #1	N/A	10 min	

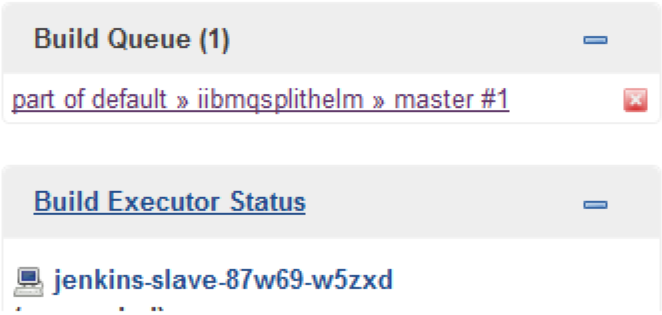
Icon: S M L

Legend RSS for all RSS for failures RSS for just latest builds



admin | log out

ENABLE AUTO REFRESH

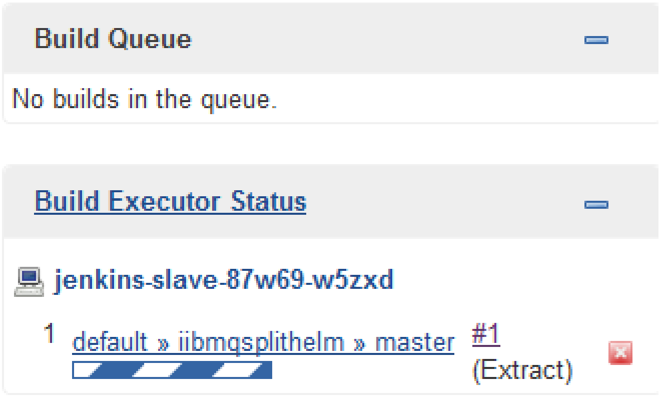


Build Queue (1)

part of default » iibmqsplithelm » master #1

Build Executor Status

jenkins-slave-87w69-w5zxd



Build Queue

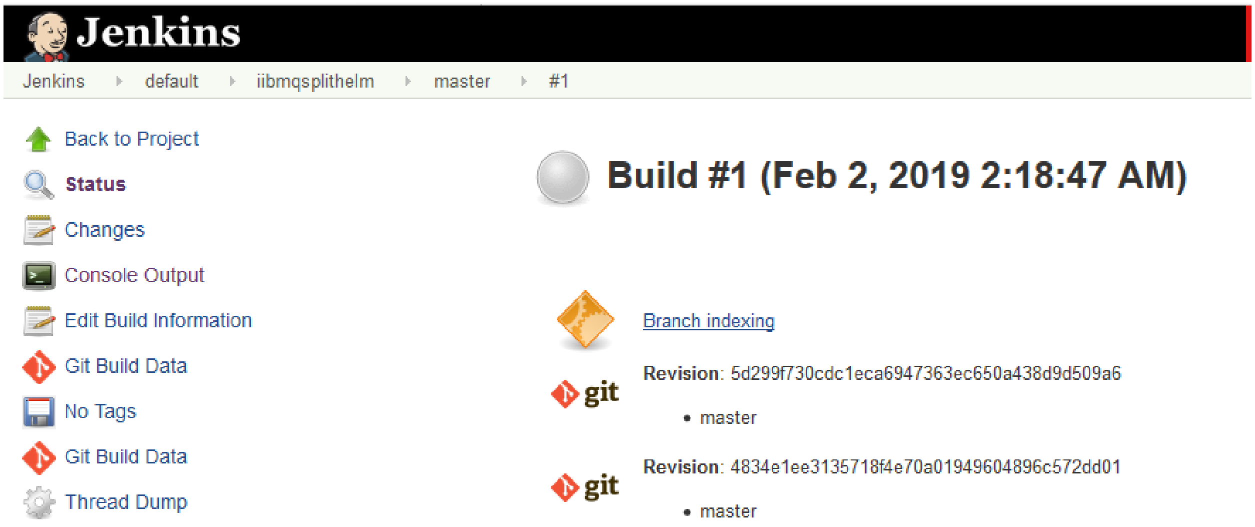
No builds in the queue.

Build Executor Status

jenkins-slave-87w69-w5zxd

1 default » iibmqsplithelm » master #1 (Extract)

Click on the master #1 link



Build #1 (Feb 2, 2019 2:18:47 AM)

Branch indexing

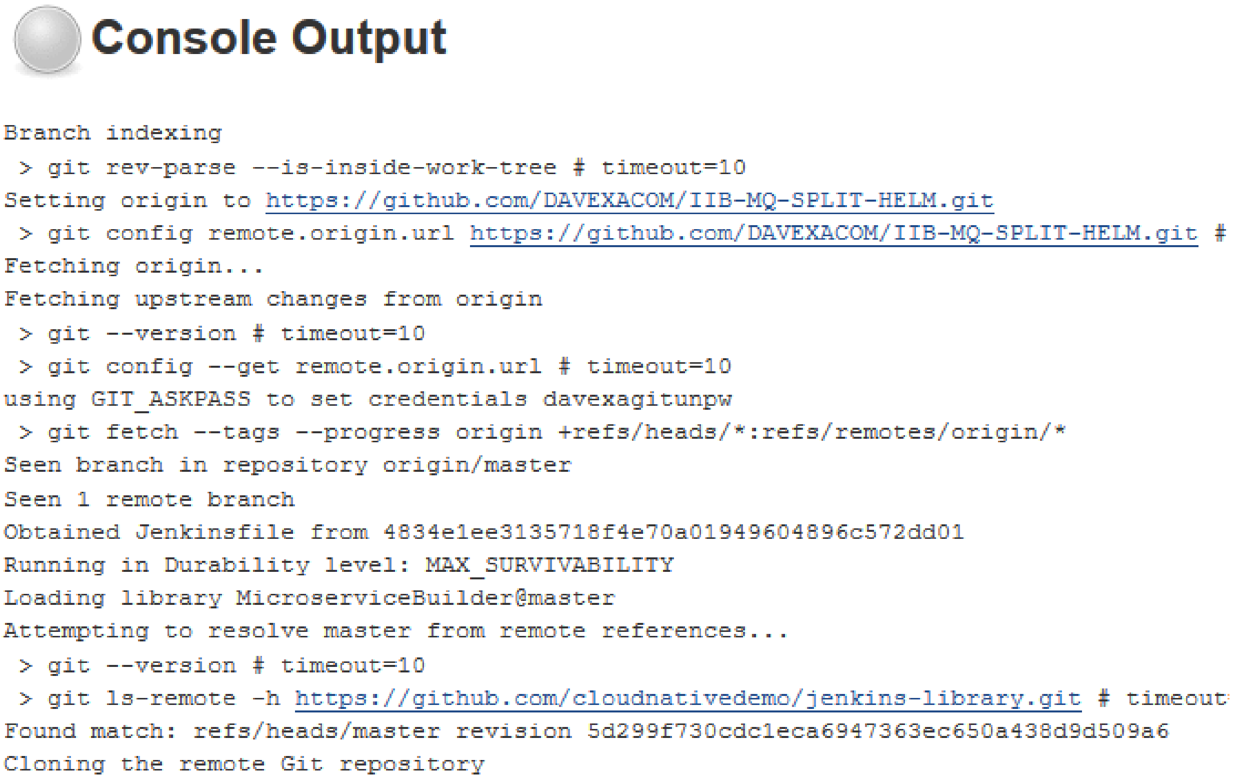
Revision: 5d299f730cdc1eca6947363ec650a438d9d509a6

- master

Revision: 4834e1ee3135718f4e70a01949604896c572dd01

- master

Click on Console Output

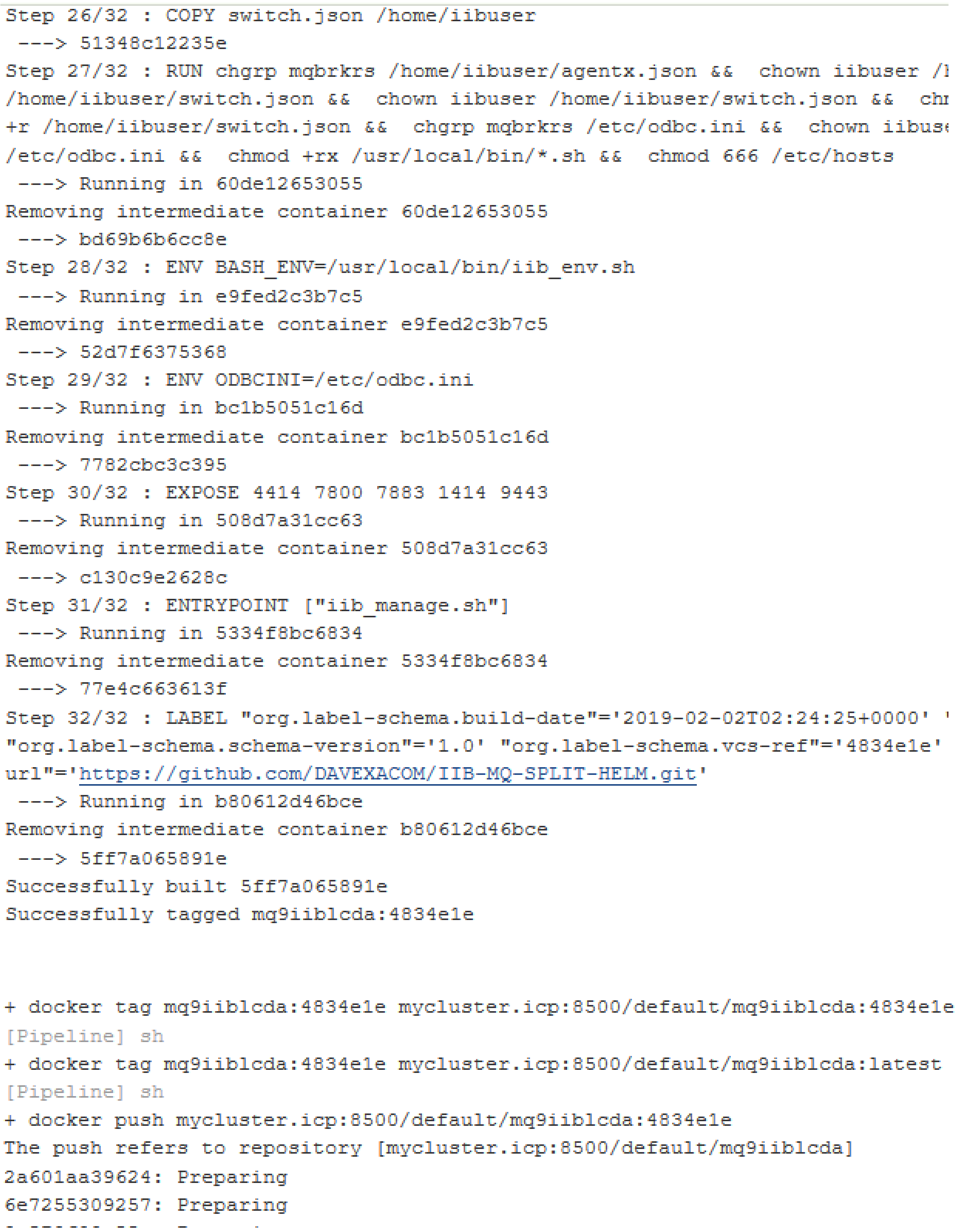


Console Output

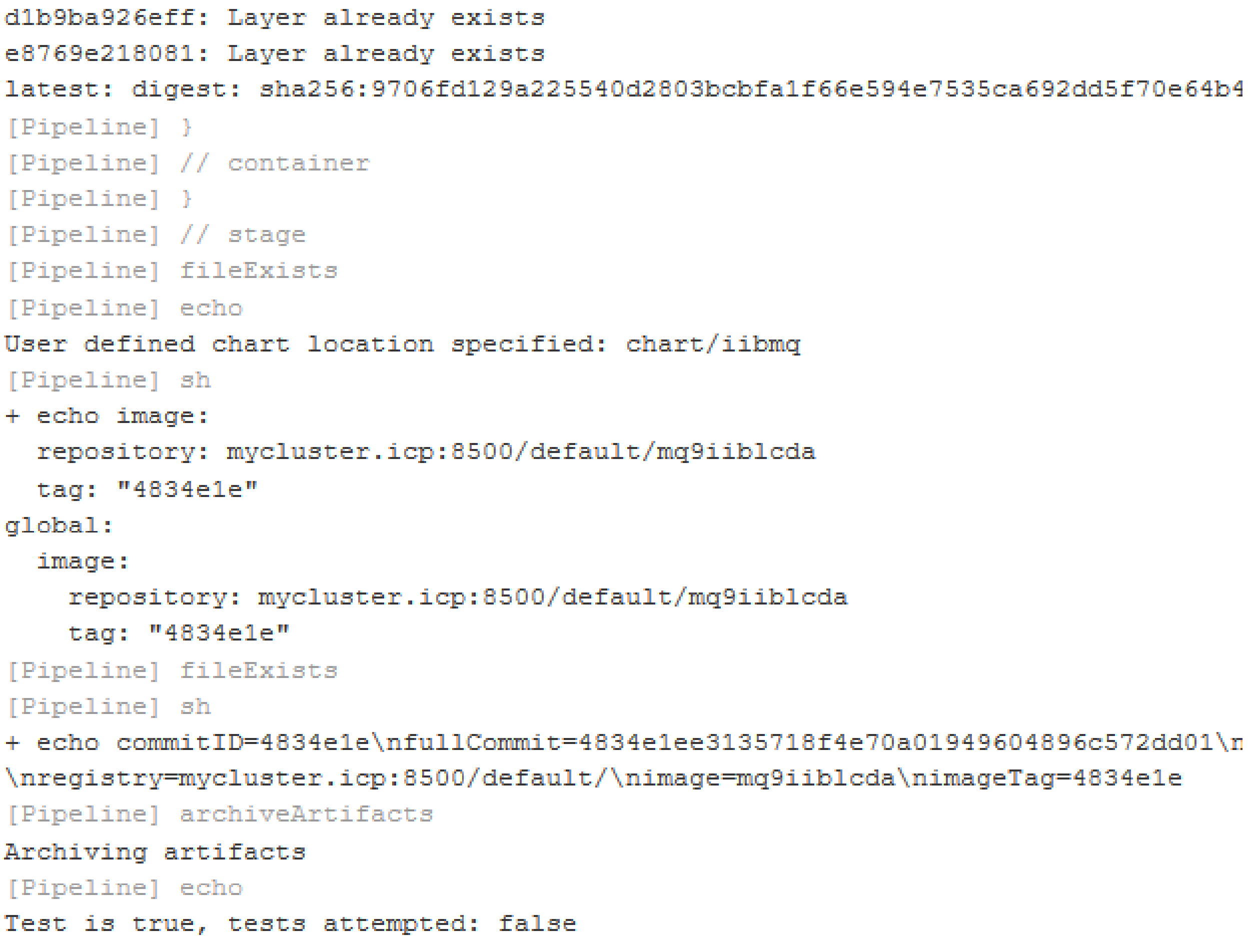
```
Branch indexing
 > git rev-parse --is-inside-work-tree # timeout=10
Setting origin to https://github.com/DAVEXACOM/IIB-MQ-SPLIT-HELM.git
 > git config remote.origin.url https://github.com/DAVEXACOM/IIB-MQ-SPLIT-HELM.git #
Fetching origin...
Fetching upstream changes from origin
 > git --version # timeout=10
 > git config --get remote.origin.url # timeout=10
using GIT_ASKPASS to set credentials davexagitunpw
 > git fetch --tags --progress origin +refs/heads/*:refs/remotes/origin/*
Seen branch in repository origin/master
Seen 1 remote branch
Obtained Jenkinsfile from 4834e1ee3135718f4e70a01949604896c572dd01
Running in Durability level: MAX_SURVIVABILITY
Loading library MicroserviceBuilder@master
Attempting to resolve master from remote references...
 > git --version # timeout=10
 > git ls-remote -h https://github.com/cloudnativedemo/jenkins-library.git # timeout
Found match: refs/heads/master revision 5d299f730cdc1eca6947363ec650a438d9d509a6
Cloning the remote Git repository
```

```
Sending build context to Docker daemon  521.2kB

Step 1/32 : FROM ubuntu:14.04
14.04: Pulling from library/ubuntu
e53f134edff2: Pulling fs layer
efbbd466a715: Pulling fs layer
e11368b8e0c7: Pulling fs layer
7dab2de7692b: Pulling fs layer
7dab2de7692b: Waiting
e11368b8e0c7: Download complete
efbbd466a715: Download complete
7dab2de7692b: Download complete
e53f134edff2: Verifying Checksum
e53f134edff2: Download complete
e53f134edff2: Pull complete
efbbd466a715: Pull complete
e11368b8e0c7: Pull complete
7dab2de7692b: Pull complete
Digest: sha256:cac55e5d97fad634d954d00a5c2a56d80576a08dcc01036011f26b88263f1578
Status: Downloaded newer image for ubuntu:14.04

Step 7/32 : ARG MQ_PACKAGES="MQSeriesRuntime-*.rpm MQSeriesServer-*.rpm MQSeriesMsg*.r
MQSeriesJRE*.rpm MQSeriesGSKit*.rpm MQSeriesWeb*.rpm"
 ---> Running in 41293cb6b08f
Removing intermediate container 41293cb6b08f
 ---> f6cb920cd982
Step 8/32 : RUN mkdir -p /tmp/mq      && cd /tmp/mq         && curl -LO $MQ_URL
./*.tar.gz     && groupadd --gid 1000 mqm     && useradd --create-home --home-dir /h
mqm     && usermod -G mqm root  && cd /tmp/mq/MQServer         && ./mqlicense.sh -tex
--force-debian $MQ_PACKAGES     && /opt/mqm/bin/setmqinst -p /opt/mqm -i       && rm
/var/lib/apt/lists/*    && rm -rf /var/mqm     && sed -i 's/PASS_MAX_DAYS\t99999/PASS
/etc/login.defs         && sed -i 's/PASS_MIN_DAYS\t0/PASS_MIN_DAYS\t1/' /etc/login.de
\[success=1 default=ignore\]\tpam_unix\.so obscure sha512/password\t[success=1 default
sha512 minlen=8/' /etc/pam.d/common-password
 ---> Running in b149557e1487
[91m  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                               Dload  Upload   Total   Spent    Left  Speed
 16  771M   16  126M    0     0   250M      0  0:00:03 --:--:--  0:00:03  250M[0m[91m
 55  771M   55  425M    0     0   282M      0  0:00:02  0:00:01  0:00:01  282M[0m[91m
100  771M  100  771M    0     0   320M      0  0:00:02  0:00:02 --:--:--  320M
[0mMQServer/
MQServer/MQSeriesMsg_pt-9.0.5-0.x86_64.rpm
MQServer/READMES/
MQServer/READMES/readme_es_ES
MQServer/READMES/readme_it_IT
```

```
MQSeriesGSKit-9.0.5-0                 ########################################
[91m143 of 143 tasks have been completed successfully.
[0m[91m'Installation1' (/opt/mqm) set as the primary installation.
[0mRemoving intermediate container b149557e1487
 ---> f52ff112560a
Step 9/32 : COPY mq-dev-config.sh mq-license-check.sh mq.sh setup-mqm-web.sh setup-var-mqm.sh /usr/local/bin/
 ---> 5ccf05ca5fdb
Step 10/32 : COPY *.mqsc /etc/mqm/
 ---> 0b5bf89a42ec
Step 11/32 : COPY *.bar  /etc/mqm/
 ---> 6f310abfe269
Step 12/32 : COPY admin.json /etc/mqm/
 ---> edec88855c57
Step 13/32 : COPY mq-dev-config /etc/mqm/mq-dev-config
 ---> 451186ec3cd9
Step 14/32 : RUN chmod +x /usr/local/bin/*.sh
 ---> Running in 50c288054f38
Removing intermediate container 50c288054f38
 ---> f33884b87e99
Step 15/32 : RUN mkdir /opt/ibm &&     curl http://172.23.50.125/iib10/10.0.0.10-IIB-LINUX64-DEVELOPER.tar.gz     |
zx --exclude iib-10.0.0.10/tools --directory /opt/ibm &&     /opt/ibm/iib-10.0.0.10/iib make registry global accept
license silently
 ---> Running in 653f72de6204
[91m  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                    [0m[91m             Dload  Upload   Total   Spent    Left  Speed
 86 1211M   86 1051M    0     0  69.4M      0  0:00:17  0:00:15  0:00:02 81.2M[0m[91m
 94 1211M   94 1151M    0     0  71.2M      0  0:00:17  0:00:16  0:00:01 84.7M[0m[91m
```

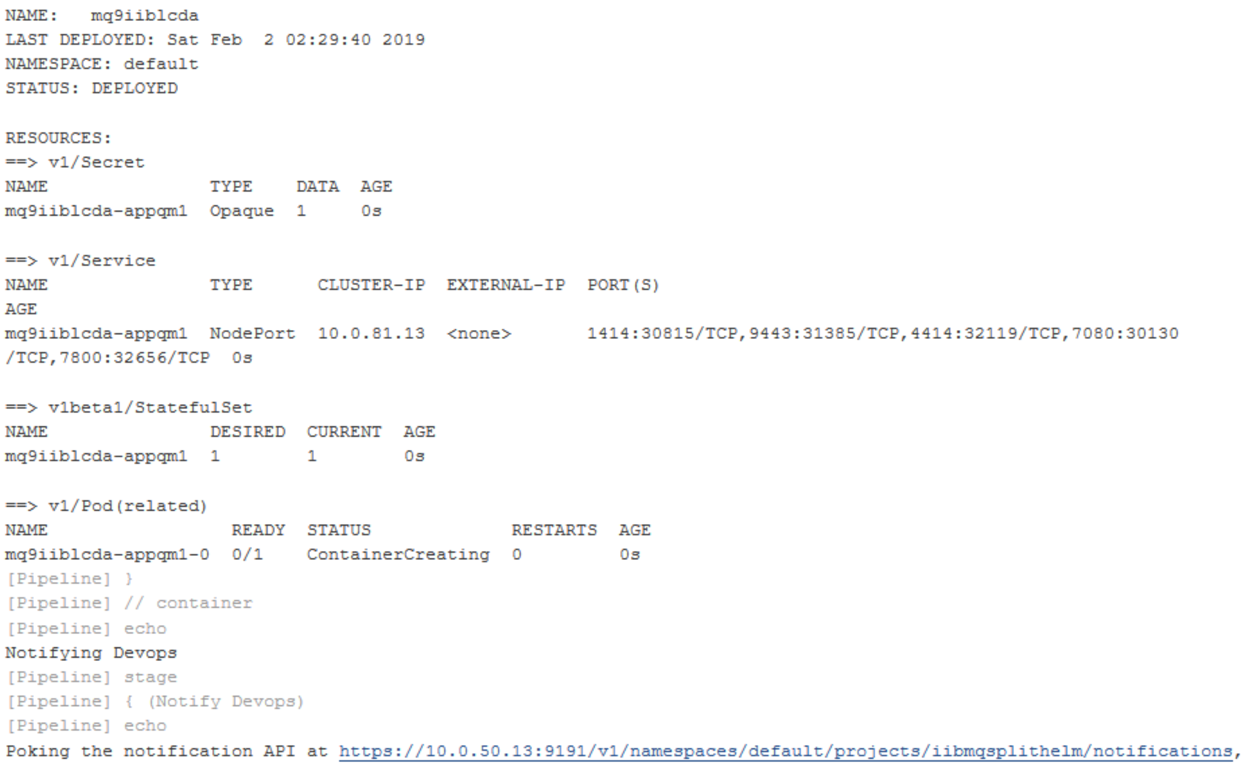


```
Step 26/32 : COPY switch.json /home/iibuser
 ---> 51348c12235e
Step 27/32 : RUN chgrp mqbrkrs /home/iibuser/agentx.json &&  chown iibuser /h
/home/iibuser/switch.json &&  chown iibuser /home/iibuser/switch.json &&  chm
+r /home/iibuser/switch.json &&  chgrp mqbrkrs /etc/odbc.ini &&  chown iibuse
/etc/odbc.ini &&  chmod +rx /usr/local/bin/*.sh &&  chmod 666 /etc/hosts
 ---> Running in 60de12653055
Removing intermediate container 60de12653055
 ---> bd69b6b6cc8e
Step 28/32 : ENV BASH_ENV=/usr/local/bin/iib_env.sh
 ---> Running in e9fed2c3b7c5
Removing intermediate container e9fed2c3b7c5
 ---> 52d7f6375368
Step 29/32 : ENV ODBCINI=/etc/odbc.ini
 ---> Running in bc1b5051c16d
Removing intermediate container bc1b5051c16d
 ---> 7782cbc3c395
Step 30/32 : EXPOSE 4414 7800 7883 1414 9443
 ---> Running in 508d7a31cc63
Removing intermediate container 508d7a31cc63
 ---> c130c9e2628c
Step 31/32 : ENTRYPOINT ["iib_manage.sh"]
 ---> Running in 5334f8bc6834
Removing intermediate container 5334f8bc6834
 ---> 77e4c663613f
Step 32/32 : LABEL "org.label-schema.build-date"='2019-02-02T02:24:25+0000' '
"org.label-schema.schema-version"='1.0' "org.label-schema.vcs-ref"='4834e1e'
url"='https://github.com/DAVEXACOM/IIB-MQ-SPLIT-HELM.git'
 ---> Running in b80612d46bce
Removing intermediate container b80612d46bce
 ---> 5ff7a065891e
Successfully built 5ff7a065891e
Successfully tagged mq9iiblcda:4834e1e



+ docker tag mq9iiblcda:4834e1e mycluster.icp:8500/default/mq9iiblcda:4834e1e
[Pipeline] sh
+ docker tag mq9iiblcda:4834e1e mycluster.icp:8500/default/mq9iiblcda:latest
[Pipeline] sh
+ docker push mycluster.icp:8500/default/mq9iiblcda:4834e1e
The push refers to repository [mycluster.icp:8500/default/mq9iiblcda]
2a601aa39624: Preparing
6e7255309257: Preparing
```

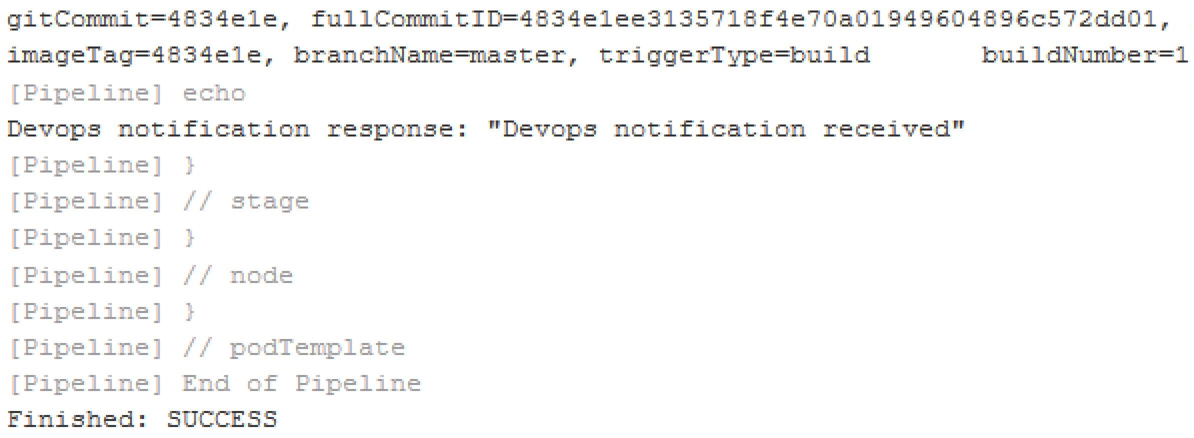


```
d1b9ba926eff: Layer already exists
e8769e218081: Layer already exists
latest: digest: sha256:9706fd129a225540d2803bcbfa1f66e594e7535ca692dd5f70e64b4
[Pipeline] }
[Pipeline] // container
[Pipeline] }
[Pipeline] // stage
[Pipeline] fileExists
[Pipeline] echo
User defined chart location specified: chart/iibmq
[Pipeline] sh
+ echo image:
  repository: mycluster.icp:8500/default/mq9iiblcda
  tag: "4834e1e"
global:
  image:
    repository: mycluster.icp:8500/default/mq9iiblcda
    tag: "4834e1e"
[Pipeline] fileExists
[Pipeline] sh
+ echo commitID=4834e1e\nfullCommit=4834e1ee3135718f4e70a01949604896c572dd01\n
\nregistry=mycluster.icp:8500/default/\nimage=mq9iiblcda\nimageTag=4834e1e
[Pipeline] archiveArtifacts
Archiving artifacts
[Pipeline] echo
Test is true, tests attempted: false
```

```
+ helm init --skip-refresh --client-only
Creating /home/jenkins/.helm
Creating /home/jenkins/.helm/repository
Creating /home/jenkins/.helm/repository/cache
Creating /home/jenkins/.helm/repository/local
Creating /home/jenkins/.helm/plugins
Creating /home/jenkins/.helm/starters
Creating /home/jenkins/.helm/cache/archive
Creating /home/jenkins/.helm/repository/repositories.yaml
Adding stable repo with URL: https://kubernetes-charts.storage.googleapis.com
Adding local repo with URL: http://127.0.0.1:8879/charts
$HELM_HOME has been configured at /home/jenkins/.helm.
Not installing Tiller due to 'client-only' flag having been set
Happy Helming!
```



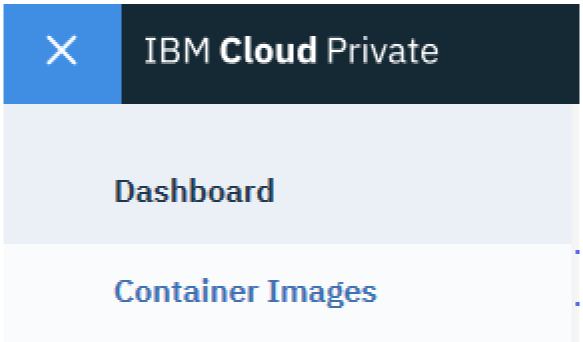
```
NAME:   mq9iiblcda
LAST DEPLOYED: Sat Feb  2 02:29:40 2019
NAMESPACE: default
STATUS: DEPLOYED

RESOURCES:
==> v1/Secret
NAME                TYPE    DATA  AGE
mq9iiblcda-appqm1  Opaque  1     0s

==> v1/Service
NAME                TYPE      CLUSTER-IP  EXTERNAL-IP  PORT(S)
AGE
mq9iiblcda-appqm1  NodePort  10.0.81.13  <none>       1414:30815/TCP,9443:31385/TCP,4414:32119/TCP,7080:30130
/TCP,7800:32656/TCP  0s

==> v1beta1/StatefulSet
NAME                DESIRED  CURRENT  AGE
mq9iiblcda-appqm1  1        1        0s

==> v1/Pod(related)
NAME                  READY  STATUS             RESTARTS  AGE
mq9iiblcda-appqm1-0  0/1    ContainerCreating  0         0s
[Pipeline] }
[Pipeline] // container
[Pipeline] echo
Notifying Devops
[Pipeline] stage
[Pipeline] { (Notify Devops)
[Pipeline] echo
Poking the notification API at https://10.0.50.13:9191/v1/namespaces/default/projects/iibmqsplithelm/notifications,
```



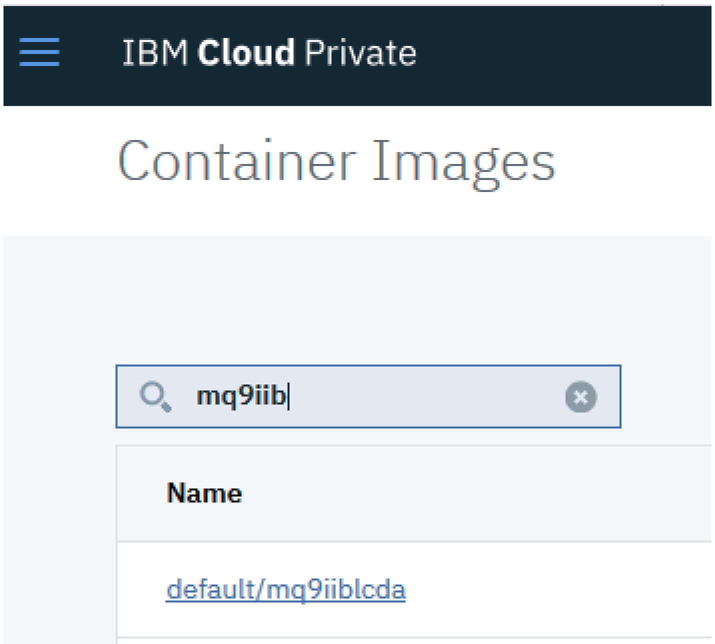
```
gitCommit=4834e1e, fullCommitID=4834e1ee3135718f4e70a01949604896c572dd01,
imageTag=4834e1e, branchName=master, triggerType=build       buildNumber=1
[Pipeline] echo
Devops notification response: "Devops notification received"
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] }
[Pipeline] // podTemplate
[Pipeline] End of Pipeline
Finished: SUCCESS
```

Checking the results of IIB-MQ-SPLIT-HELM build on IBM Cloud Private

From the IBM Cloud Private console select the container images



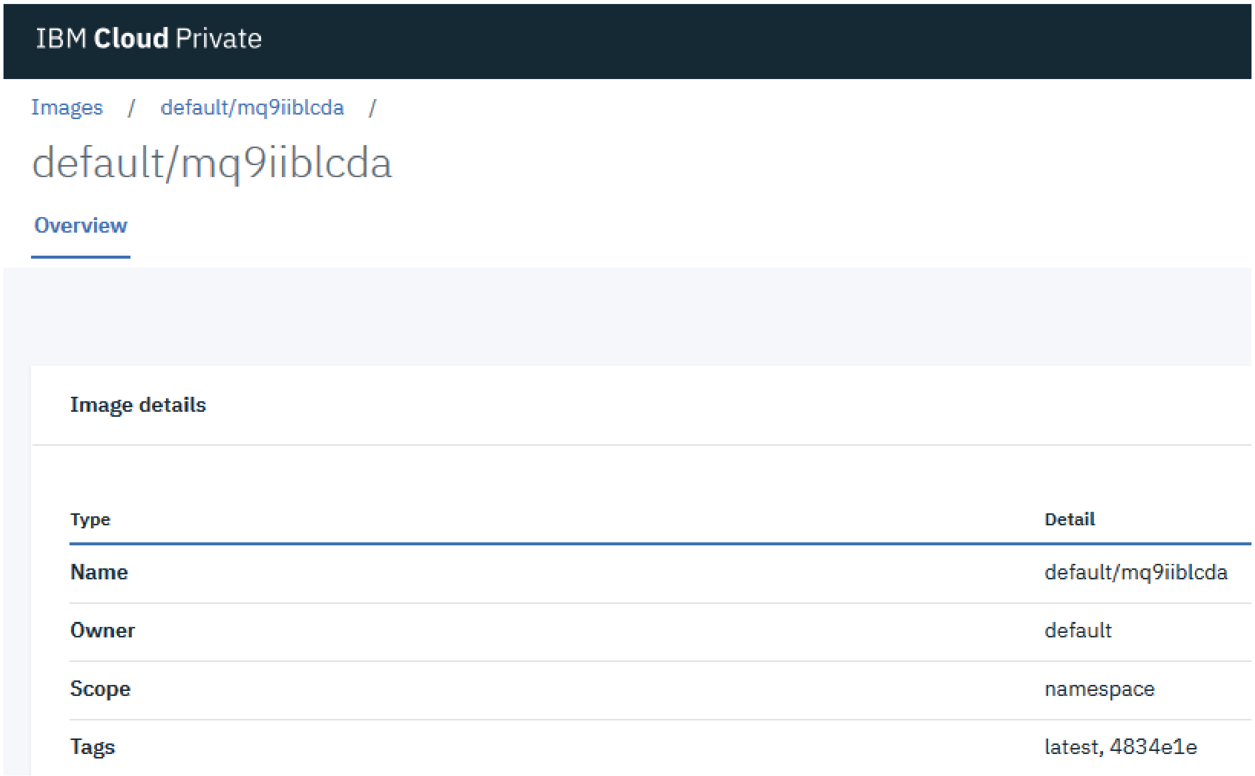
Filter on mq9iib



mq9iib

Name
default/mq9iiblcda

Click on the link to check the image details



default/mq9iiblcda

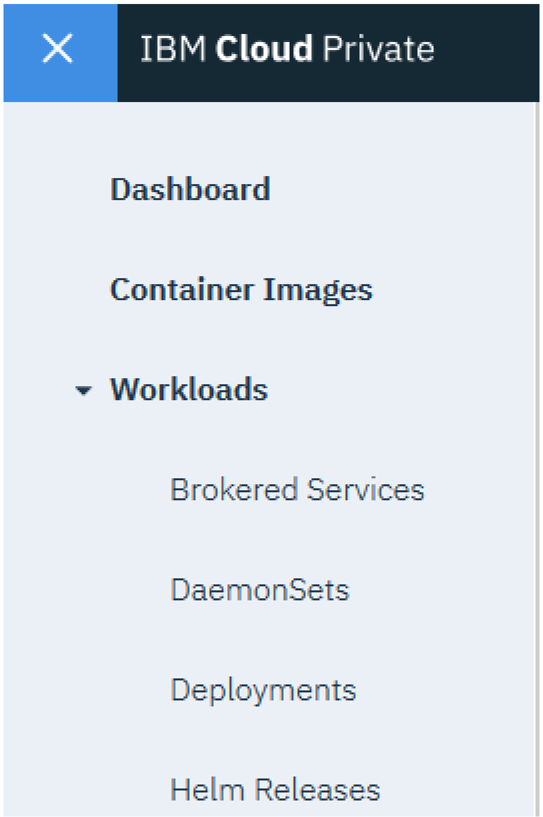
Image details

Type	Detail
Name	default/mq9iiblcda
Owner	default
Scope	namespace
Tags	latest, 4834e1e

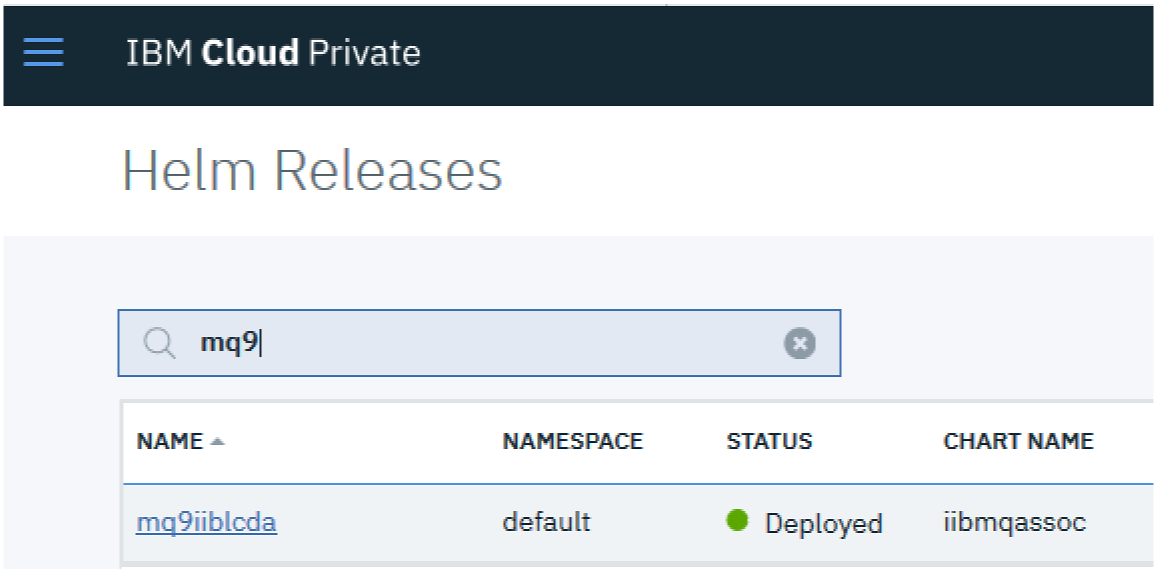
Make of note of the tag 4834e1e (in this example). We might need this in the helm charts for other repository build as “latest” does not always seem to work but we’ll try it 😊

Next let’s review the Helm releases.

Note: In the scope of this full demo we don’t really need a release for the IIB/MQ Image, we just need the image on the ICP. That said, there is value in checking that the Image delivers a runtime container that operates as expected.



Select Helm Releases and use the Search filter to find mq9iiblcda



mq9

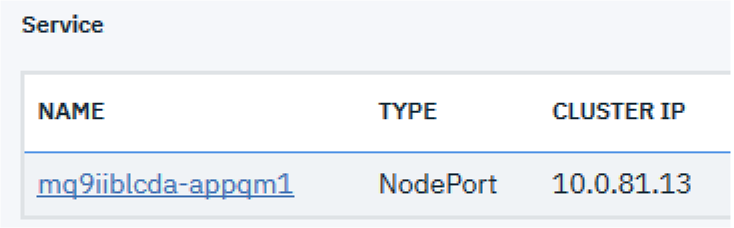
NAME	NAMESPACE	STATUS	CHART NAME
mq9iiblcda	default	● Deployed	iibmqassoc

Click on mp9iiblcda link and explore the release.

Note this is a single container running a combined IIBv10 runtime with a local MQ Queue Manager. Exploring and testing this release allows to prove we have a working container build to be leveraged by

MQ4WAYCLUSIIBLC (the MQ 4-Way cluster with IIB local binding) Helm release in the next part of the demo set up.

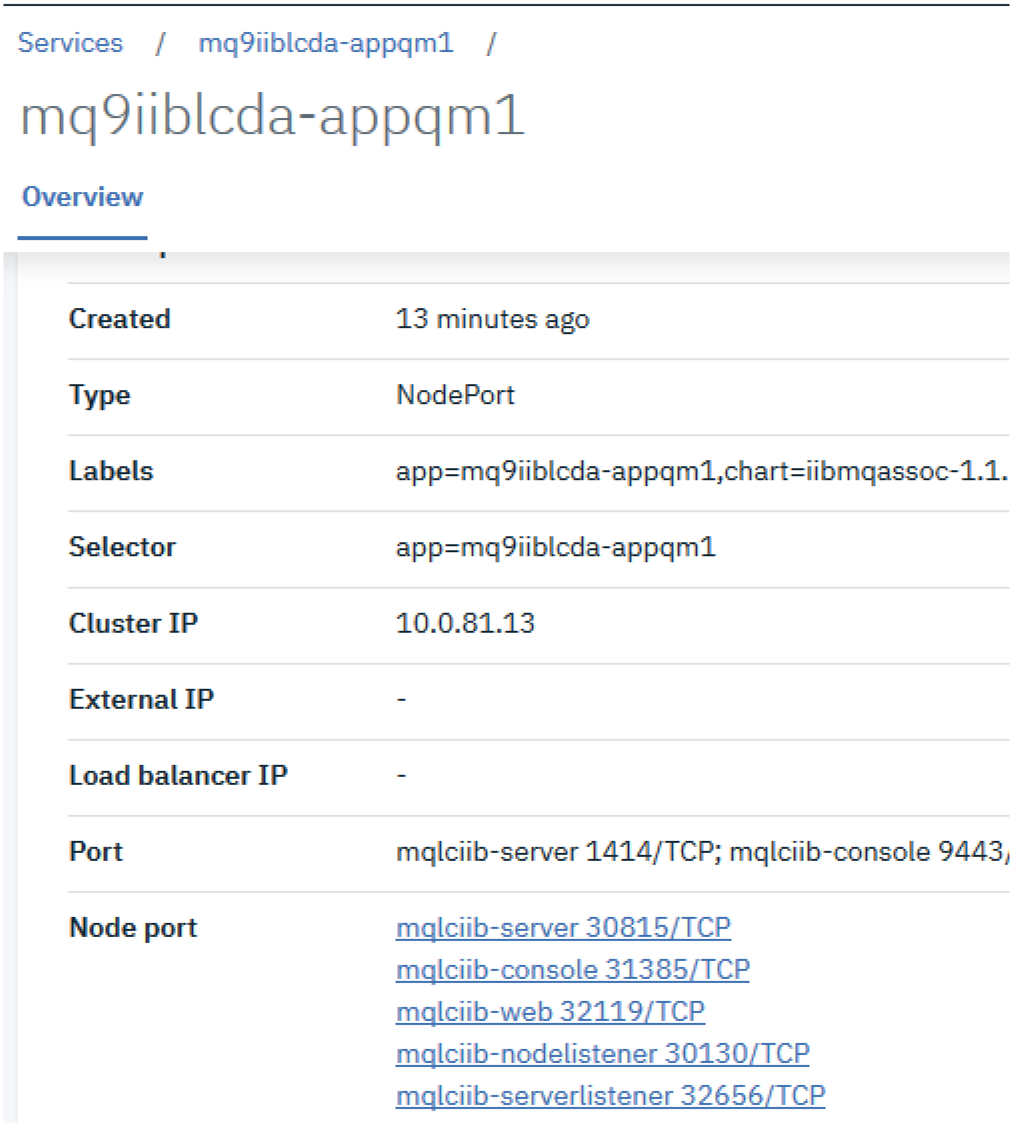
Navigate to the Service and click on the link



Service

NAME	TYPE	CLUSTER IP
mq9iiblcda-appqm1	NodePort	10.0.81.13

Here you'll find the links to for the port nodes (connection) links for the MQ and IIB services



Services / mq9iiblcda-appqm1 /

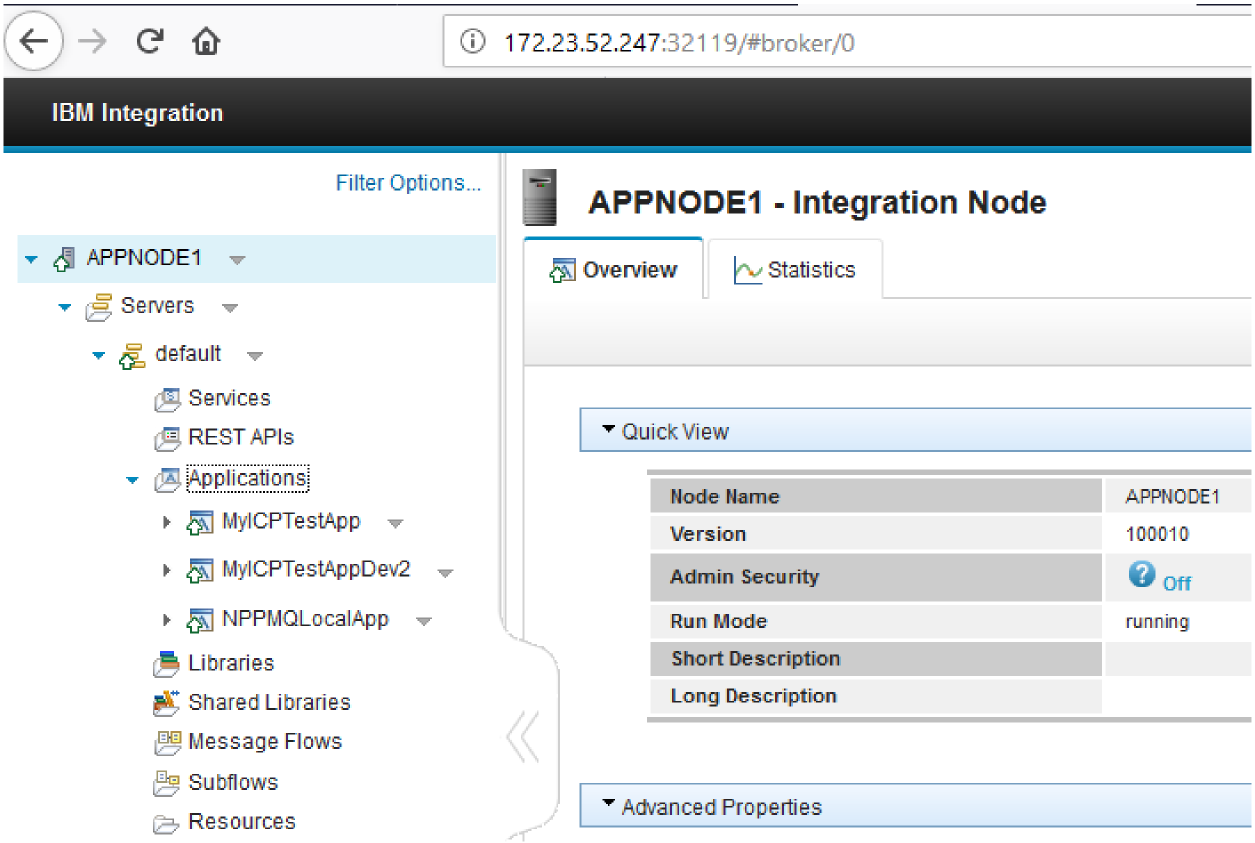
mq9iiblcda-appqm1

Created	13 minutes ago
Type	NodePort
Labels	app=mq9iiblcda-appqm1,chart=iibmqassoc-1.1.
Selector	app=mq9iiblcda-appqm1
Cluster IP	10.0.81.13
External IP	-
Load balancer IP	-
Port	mqlciib-server 1414/TCP; mqlciib-console 9443,
Node port	mqlciib-server 30815/TCP mqlciib-console 31385/TCP mqlciib-web 32119/TCP mqlciib-nodelistener 30130/TCP mqlciib-serverlistener 32656/TCP

Validating our IIB/MQ build and helm release

Let's check the IIB node first

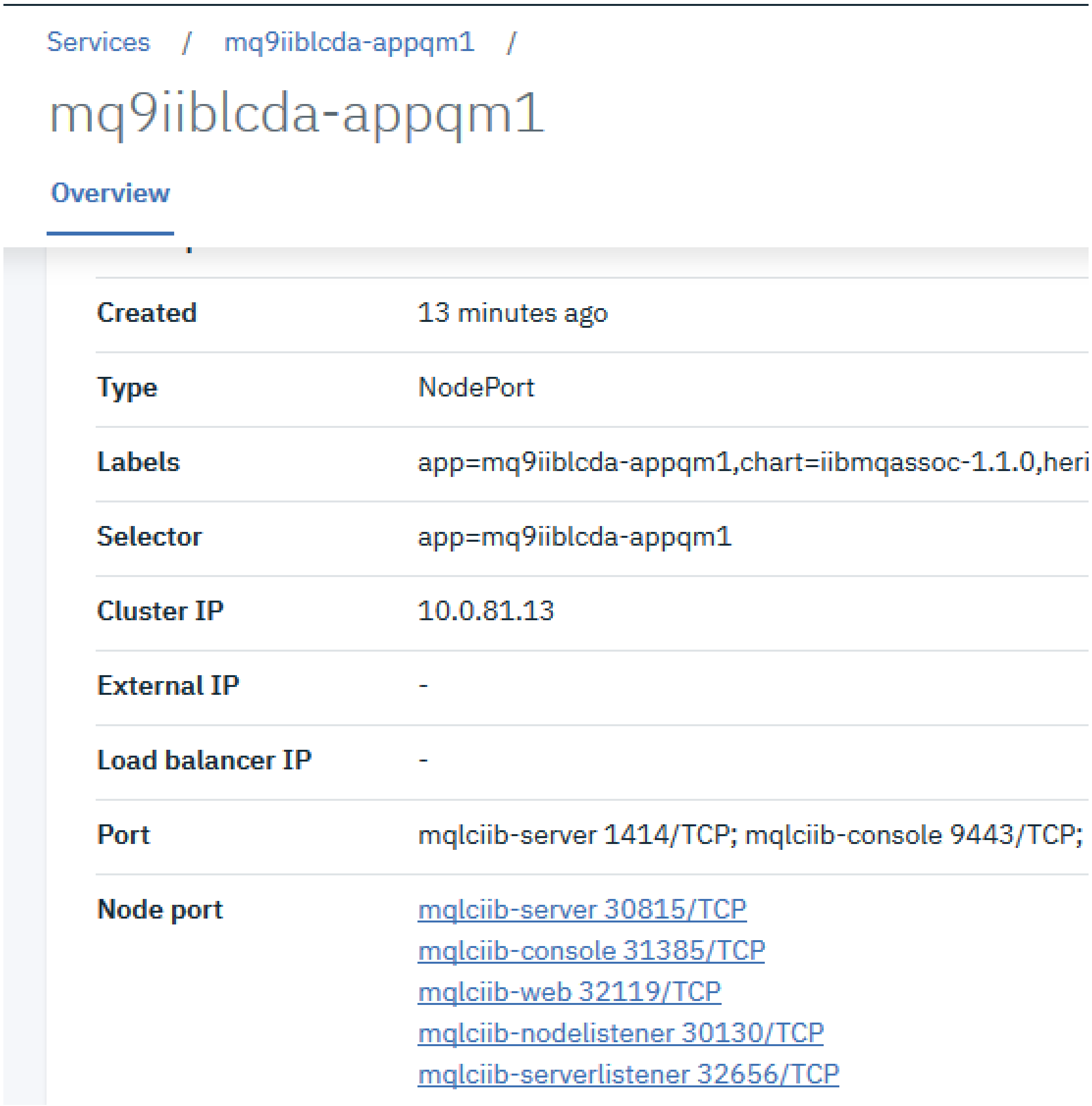
Click on the mqlciib-web nnnnn/TCP link



Node Name	APPNODE1
Version	100010
Admin Security	Off
Run Mode	running
Short Description	
Long Description	

Expand the APPNODE1 and integration server and you see a set of deployed applications.

Next Let's validate MQ. Back on the Helm Release services ICP Console window



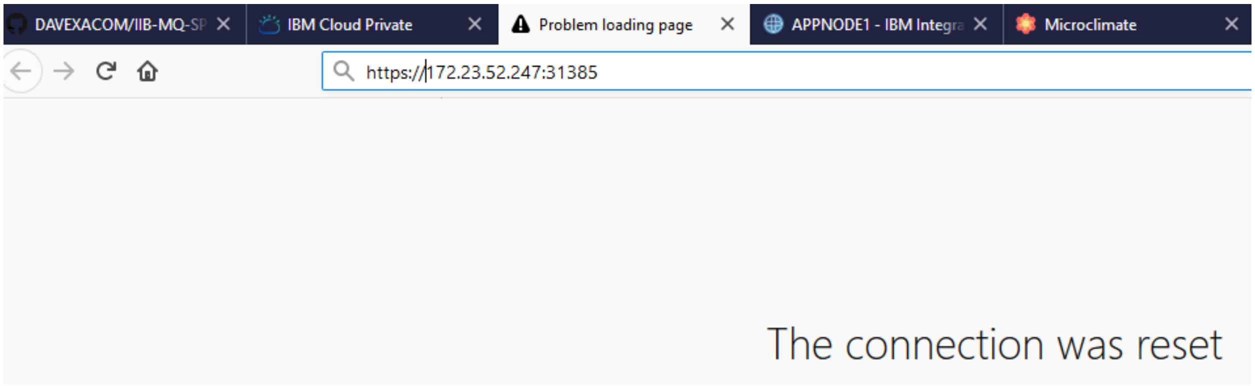
Services / mq9iiblcda-appqm1 /

mq9iiblcda-appqm1

Created	13 minutes ago
Type	NodePort
Labels	app=mq9iiblcda-appqm1,chart=iibmqassoc-1.1.0,heri
Selector	app=mq9iiblcda-appqm1
Cluster IP	10.0.81.13
External IP	-
Load balancer IP	-
Port	mqlciib-server 1414/TCP; mqlciib-console 9443/TCP;
Node port	mqlciib-server 30815/TCP mqlciib-console 31385/TCP mqlciib-web 32119/TCP mqlciib-nodelistener 30130/TCP mqlciib-serverlistener 32656/TCP

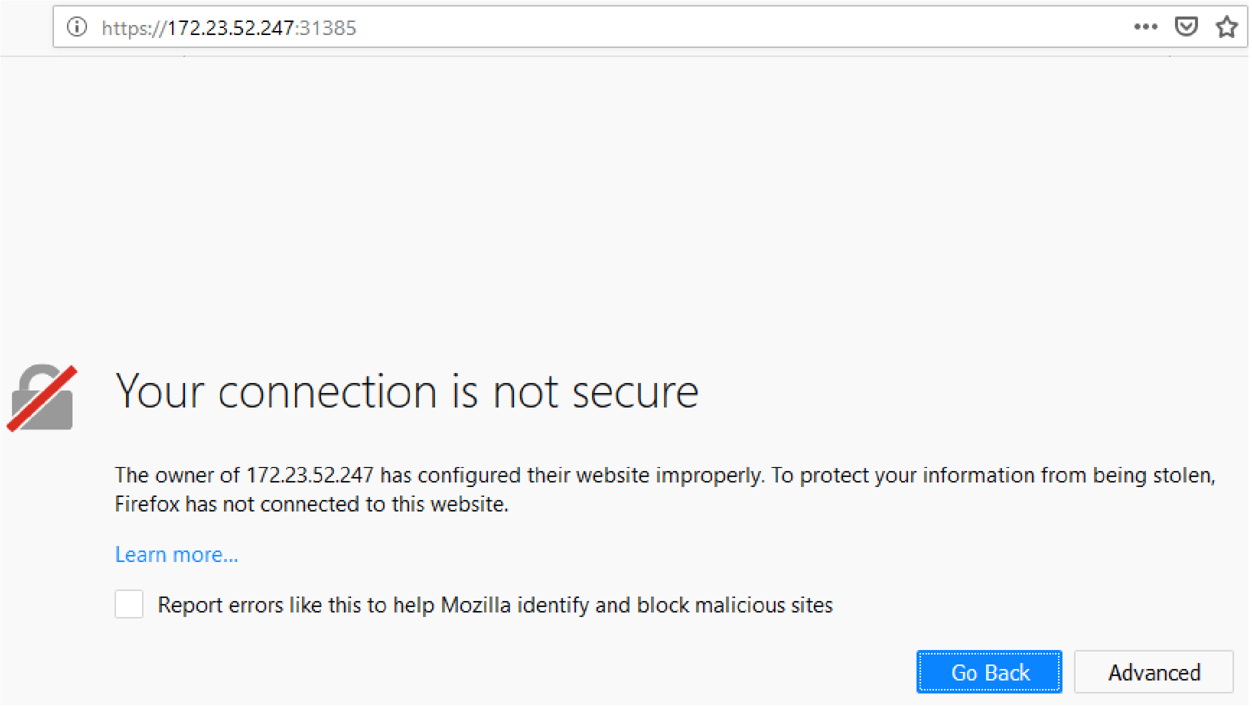
Click on the mqlciib-console nnnnn/TCP link

This will fail. You need to add https:// to the URL and hit enter again



https://172.23.52.247:31385

The connection was reset

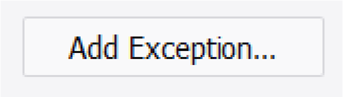

https://172.23.52.247:31385

Your connection is not secure

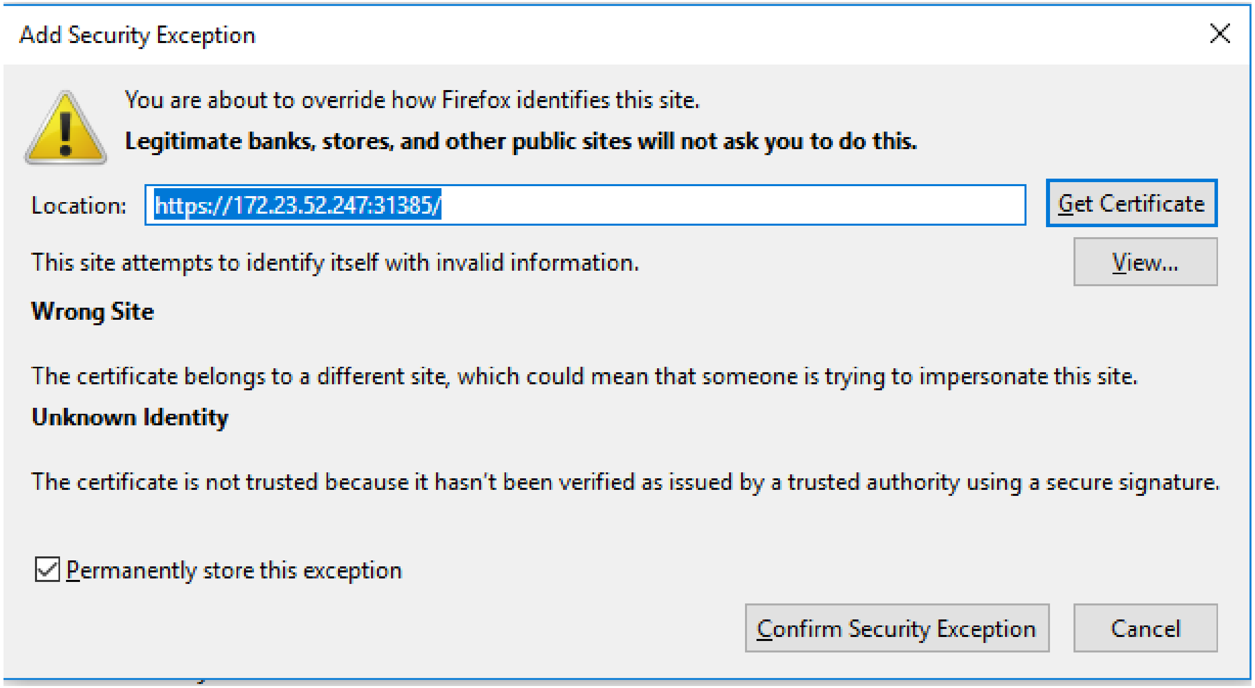
The owner of 172.23.52.247 has configured their website improperly. To protect your information from being stolen, Firefox has not connected to this website.

Learn more…
□ Report errors like this to help Mozilla identify and block malicious sites
Go Back
Advanced

Then hit Advanced button and then Add Exception button



At the next dialog hit Confirm Security Exception


Add Security Exception

You are about to override how Firefox identifies this site.

Legitimate banks, stores, and other public sites will not ask you to do this.

Location: https://172.23.52.247:31385/

This site attempts to identify itself with invalid information.

Wrong Site

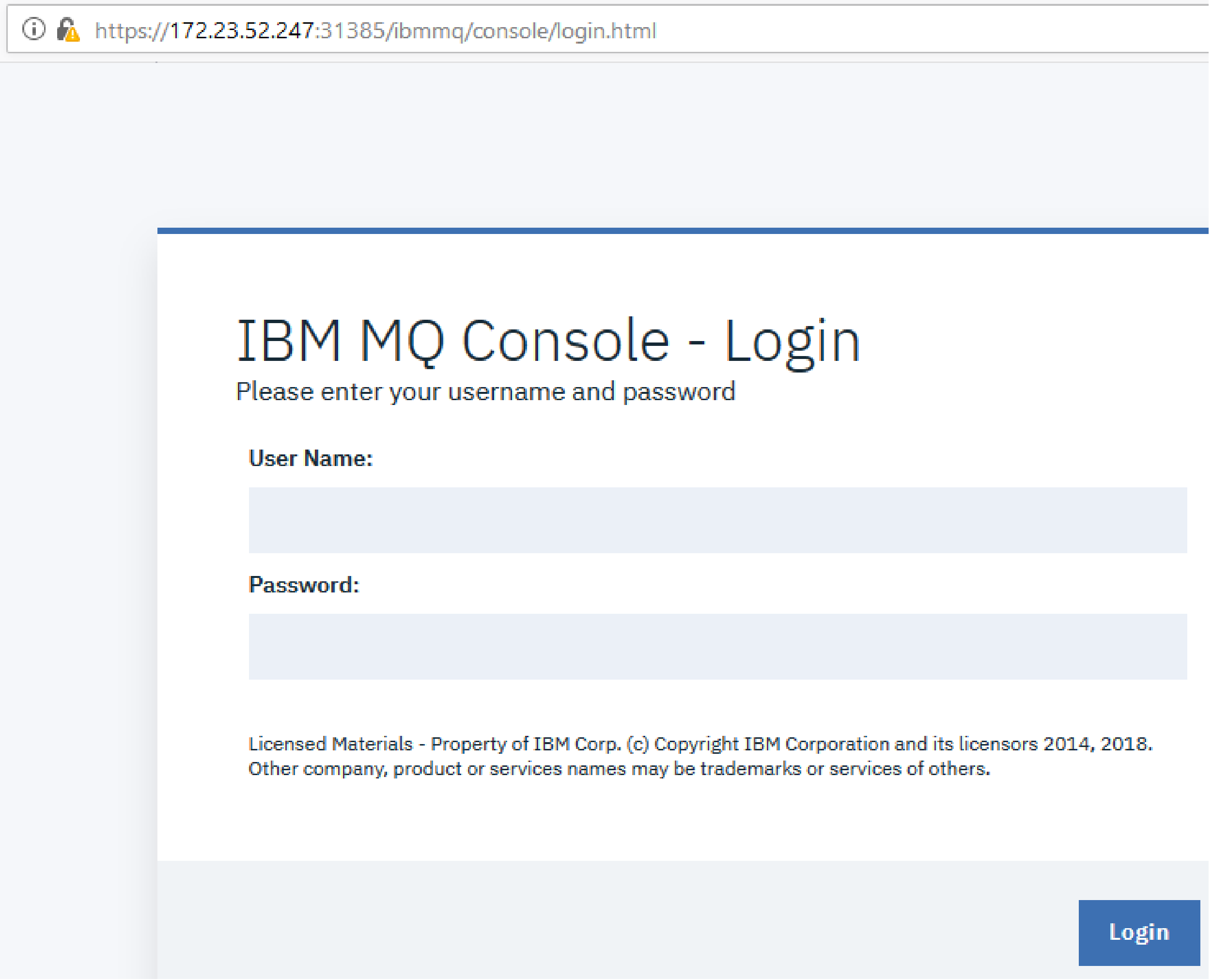
The certificate belongs to a different site, which could mean that someone is trying to impersonate this site.

Unknown Identity

The certificate is not trusted because it hasn't been verified as issued by a trusted authority using a secure signature.

☑ Permanently store this exception

Log into the MQ Console



https://172.23.52.247:31385/ibmmq/console/login.html

IBM MQ Console - Login

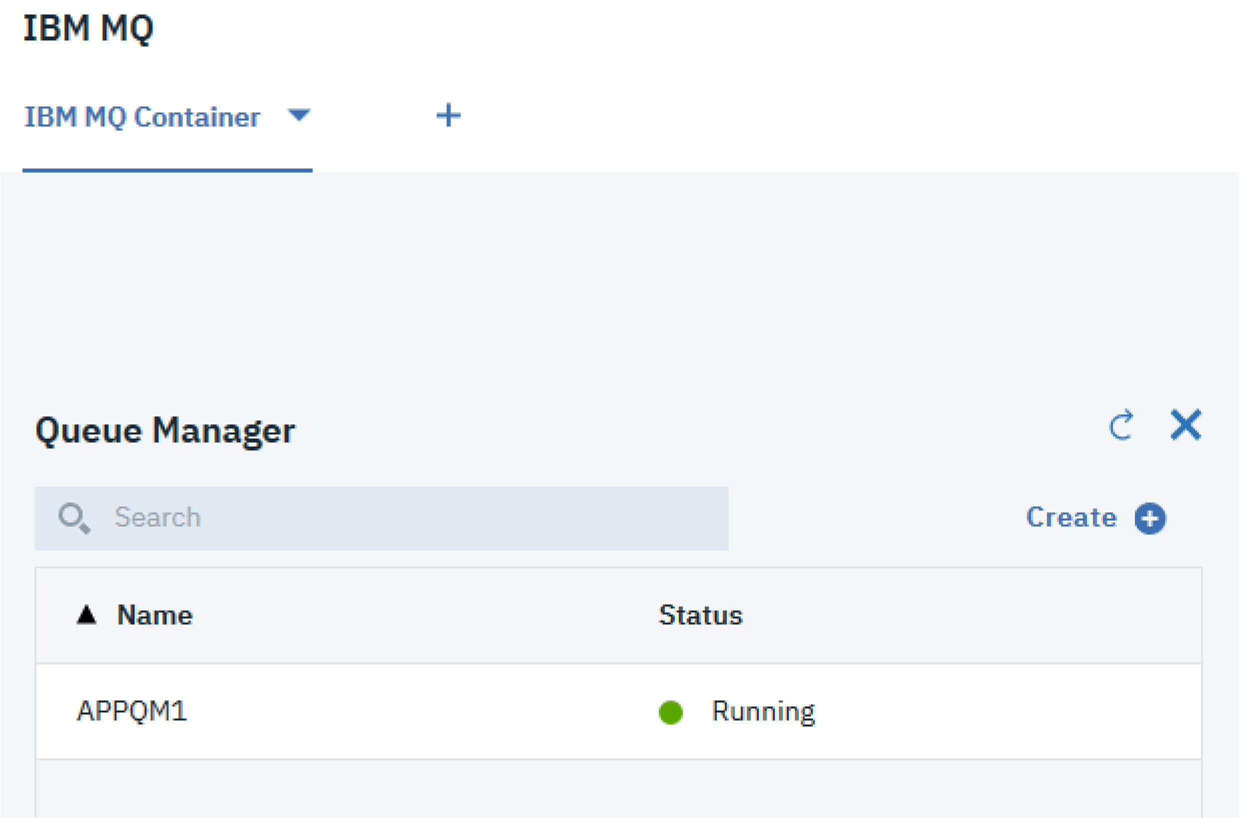
Please enter your username and password

Licensed Materials - Property of IBM Corp. (c) Copyright IBM Corporation and its licensors 2014, 2018. Other company, product or services names may be trademarks or services of others.

User Name: admin

Password: passw0rd (note that’s passw0rd with a zero)

The MQ Queue Manager APPQM1 is displayed



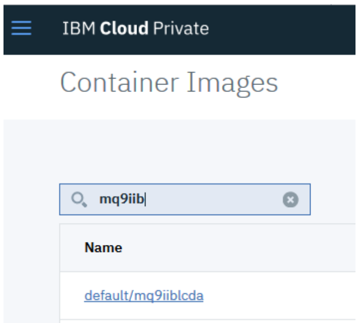
Name	Status
APPQM1	● Running

Looks like our IIB node and our MQ Queue manager are in good shape.

I'd say that proves the IIB-MQ-SPLIT-HELM image build, load to ICP repository and resulting runtime container with IIB and MQ works fine.

Next we'll remove the Helm Release and clean up. Remember we only need the image

Deleting and cleaning up our IIB-MQ-SPLIT-HELM release



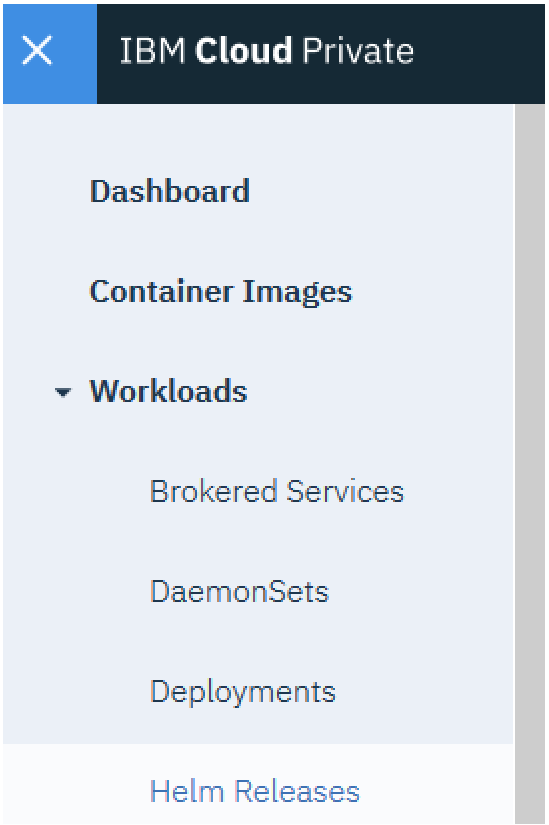
mq9iib

Name
default/mq9iiblcda

We don't need this release as part of our ESB. The MQ4WAYCLUS-LCIIB Helm Release will deliver a full 4 container release leveraging this image for the two "back office nodes" in the ESB pattern.

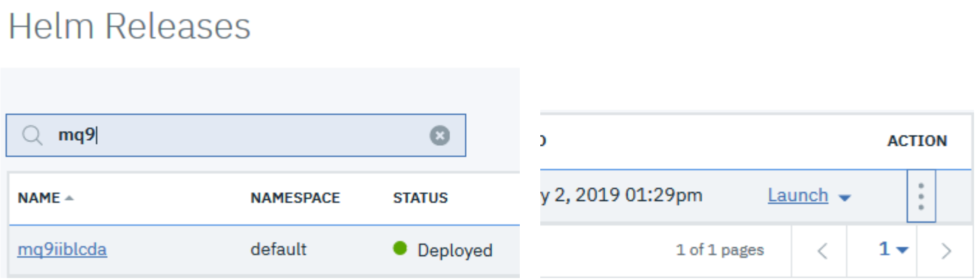
First close the browser tabs for the IIB WebUI and the MQ Console.

Then select Workloads->Helm Releases



In the Helm releases filter for mq9

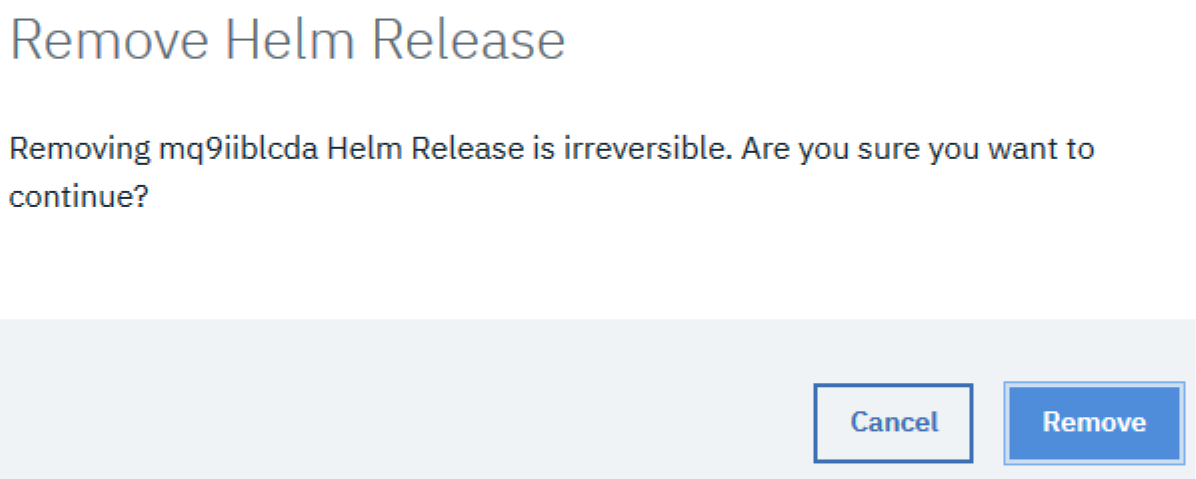
For the mq9iiblcda, on the right hand side click the action button (3 vertical dots)



NAME	NAMESPACE	STATUS	)	ACTION
mq9iiblcda	default	● Deployed	y 2, 2019 01:29pm	Launch ⋮

And select DELETE

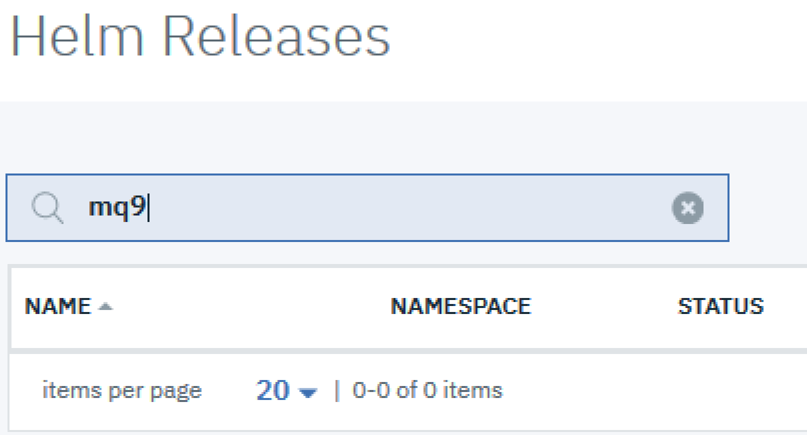
Then at the pop up window select REMOVE



Remove Helm Release

Removing mq9iiblcda Helm Release is irreversible. Are you sure you want to continue?

If you get an error, try again and refresh the screen. Sometimes there is a timeout waiting for the response for the remove. A refresh should show it has actually worked



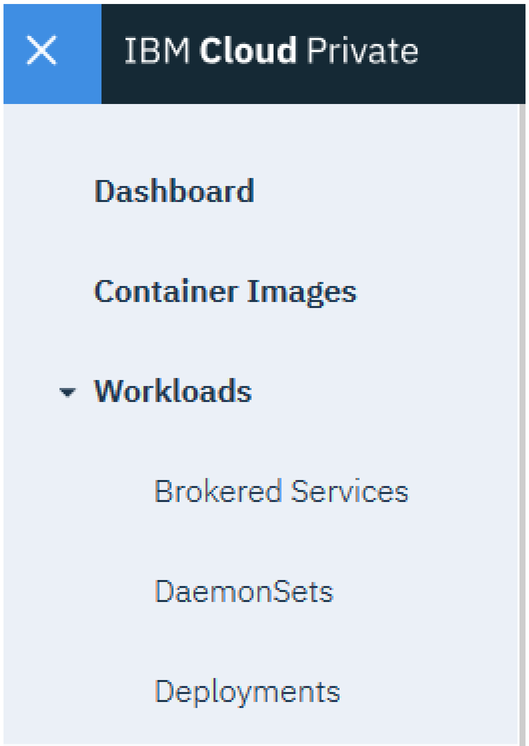
Helm Releases

NAME ▲	NAMESPACE	STATUS

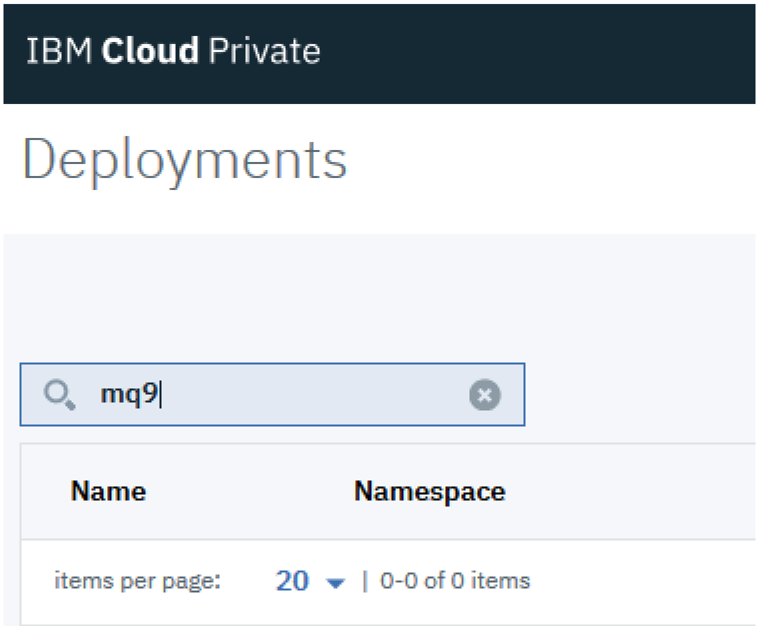
items per page 20 ▾ | 0-0 of 0 items

Double check the removal was successful by checking in the Deployments.

Select Workloads->Deployments



Filter on mq9



mq9

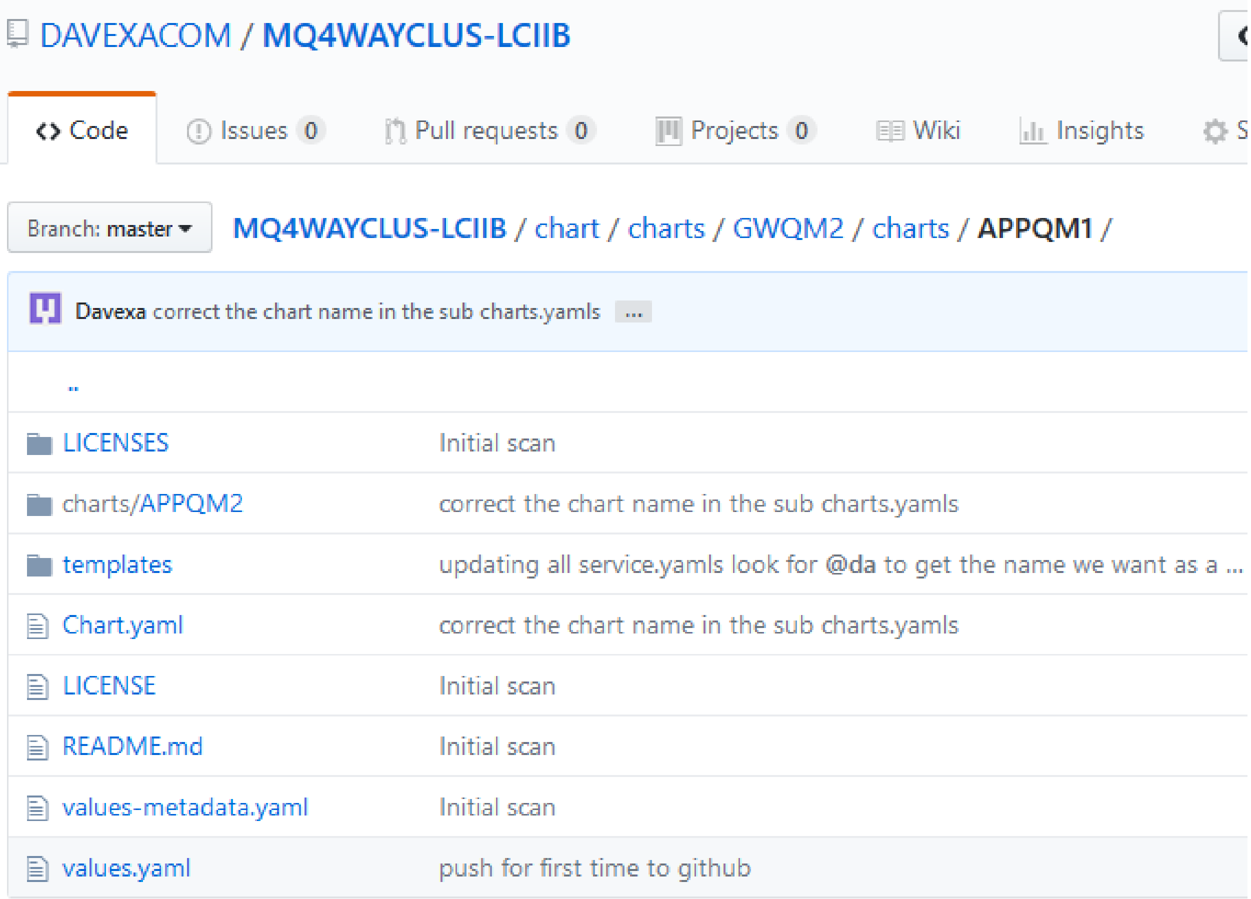
Name	Namespace

items per page: 20 | 0-0 of 0 items

Refresh the browser window until the no entries are shown.

Setting up the ESB on ICP Demo – Part 2 the MQ Image and ESB Helm Release

Review the MQ4WAYCLUS-LCIIB values.yaml for APPQM1 and APPQM2 in Github



MQ4WAYCLUS-LCIIB / chart / charts / GWQM2 / charts / APPQM1 /

Davexa correct the chart name in the sub charts.yamls ...

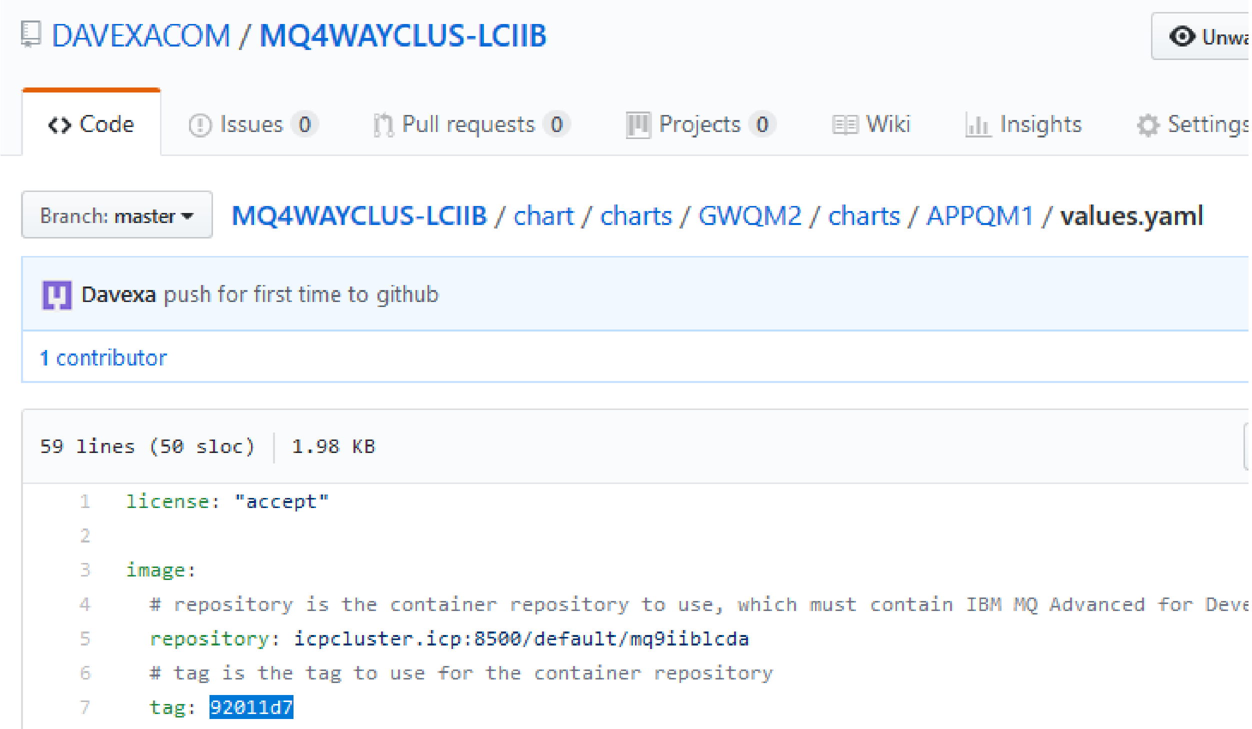
..

LICENSES	Initial scan
charts/APPQM2	correct the chart name in the sub charts.yamls
templates	updating all service.yamls look for @da to get the name we want as a ...
Chart.yaml	correct the chart name in the sub charts.yamls
LICENSE	Initial scan
README.md	Initial scan
values-metadata.yaml	Initial scan
values.yaml	push for first time to github

Select values.yaml.

See below the **TAG name** for the image **mqiiblcda** in the previous part.

In part 1 we built a new image and loaded it onto ICP therefore we have new tags.



MQ4WAYCLUS-LCIIB / chart / charts / GWQM2 / charts / APPQM1 / values.yaml

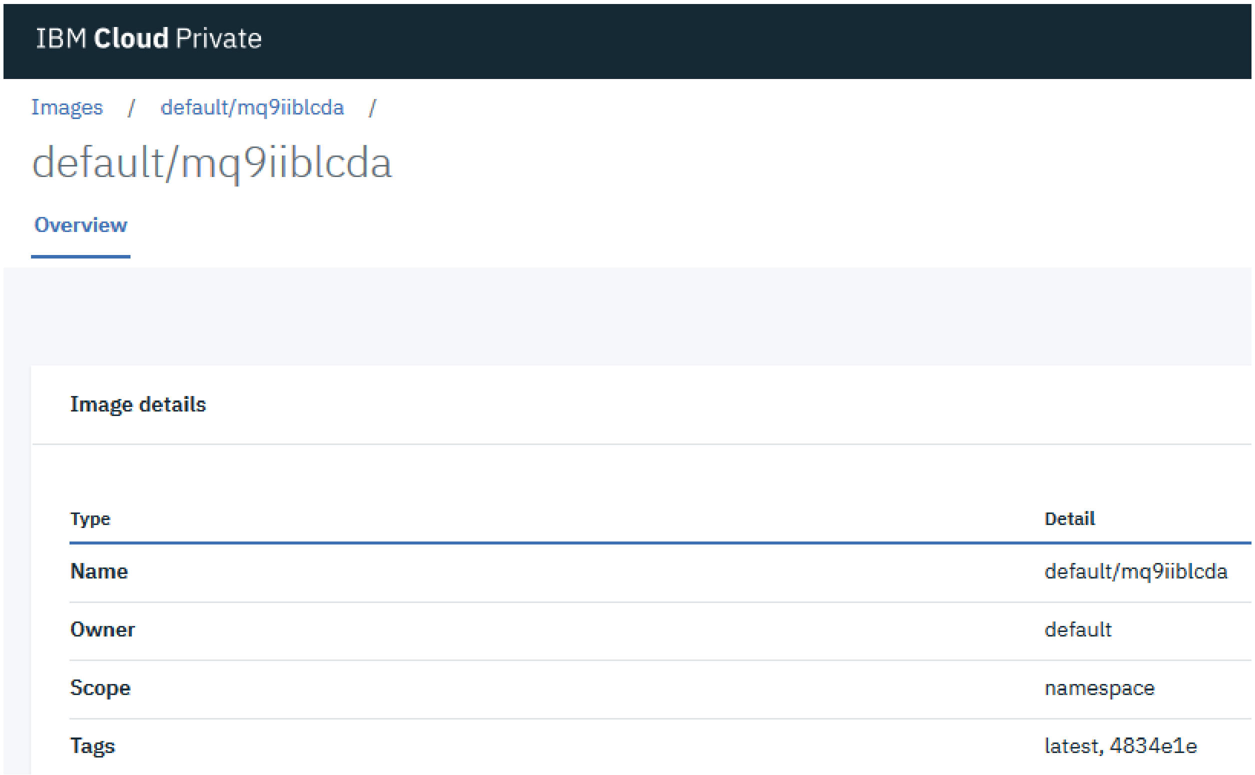
Davexa push for first time to github

1 contributor

59 lines (50 sloc) | 1.98 KB

```
license: "accept"

image:
  # repository is the container repository to use, which must contain IBM MQ Advanced for Deve
  repository: icpcluster.icp:8500/default/mq9iiblcda
  # tag is the tag to use for the container repository
  tag: 92011d7
```

As you can see we have a mis-match compared to the tags for our image on ICP.



default/mq9iiblcda

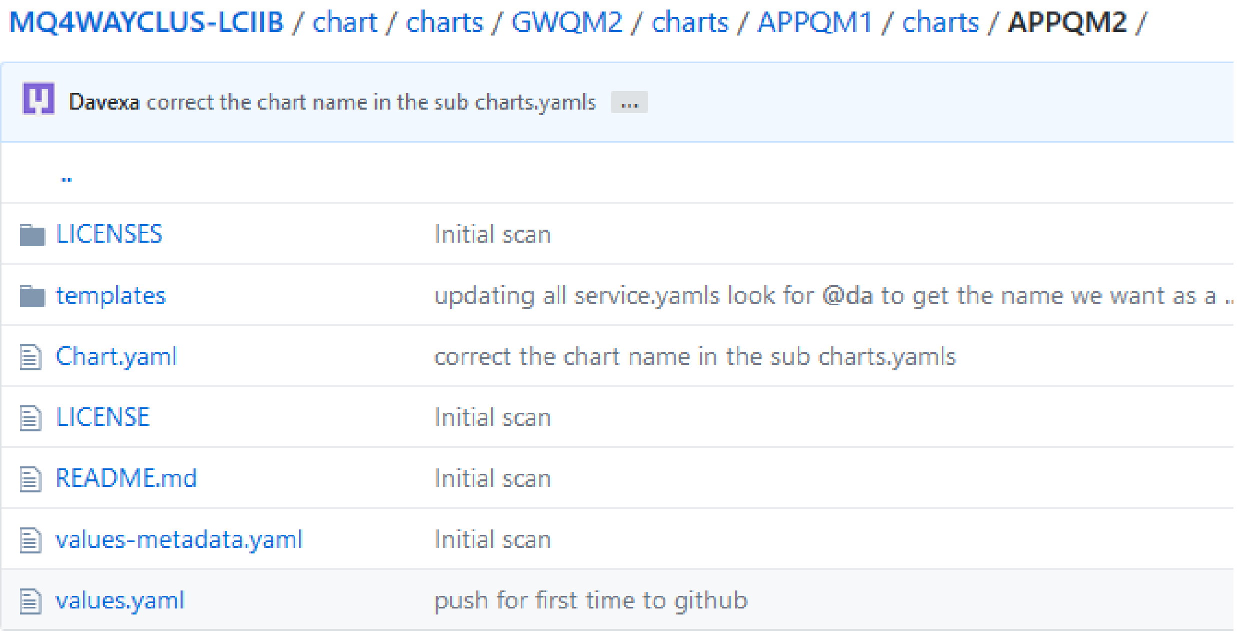
Image details

Type	Detail
Name	default/mq9iiblcda
Owner	default
Scope	namespace
Tags	latest, 4834e1e

Lets edit the values.yaml and correct the tag name. Use the Edit button on githib

Make the change to the tag name and hit the commit button

Now you must repeat the process for the values.yaml for APPQM2



MQ4WAYCLUS-LCIIB / chart / charts / GWQM2 / charts / APPQM1 / charts / APPQM2 /

Davexa correct the chart name in the sub charts.yamls …

..

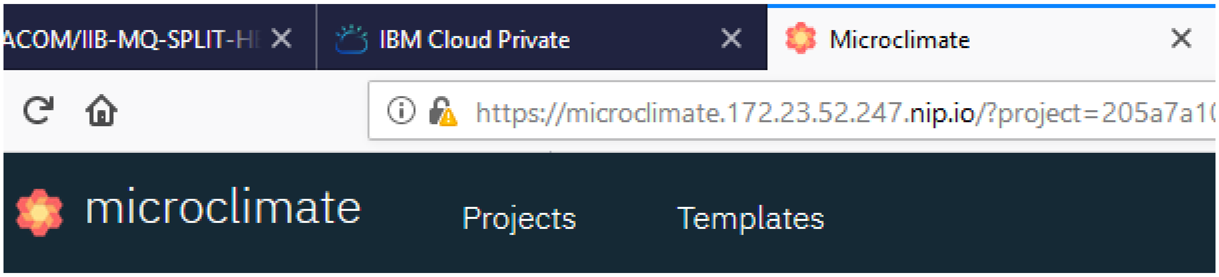
LICENSES	Initial scan
templates	updating all service.yamls look for @da to get the name we want as a ..
Chart.yaml	correct the chart name in the sub charts.yamls
LICENSE	Initial scan
README.md	Initial scan
values-metadata.yaml	Initial scan
values.yaml	push for first time to github

The github repository DAVEXACOM/MQ4WAYCLUS-LCIIC is now set up to

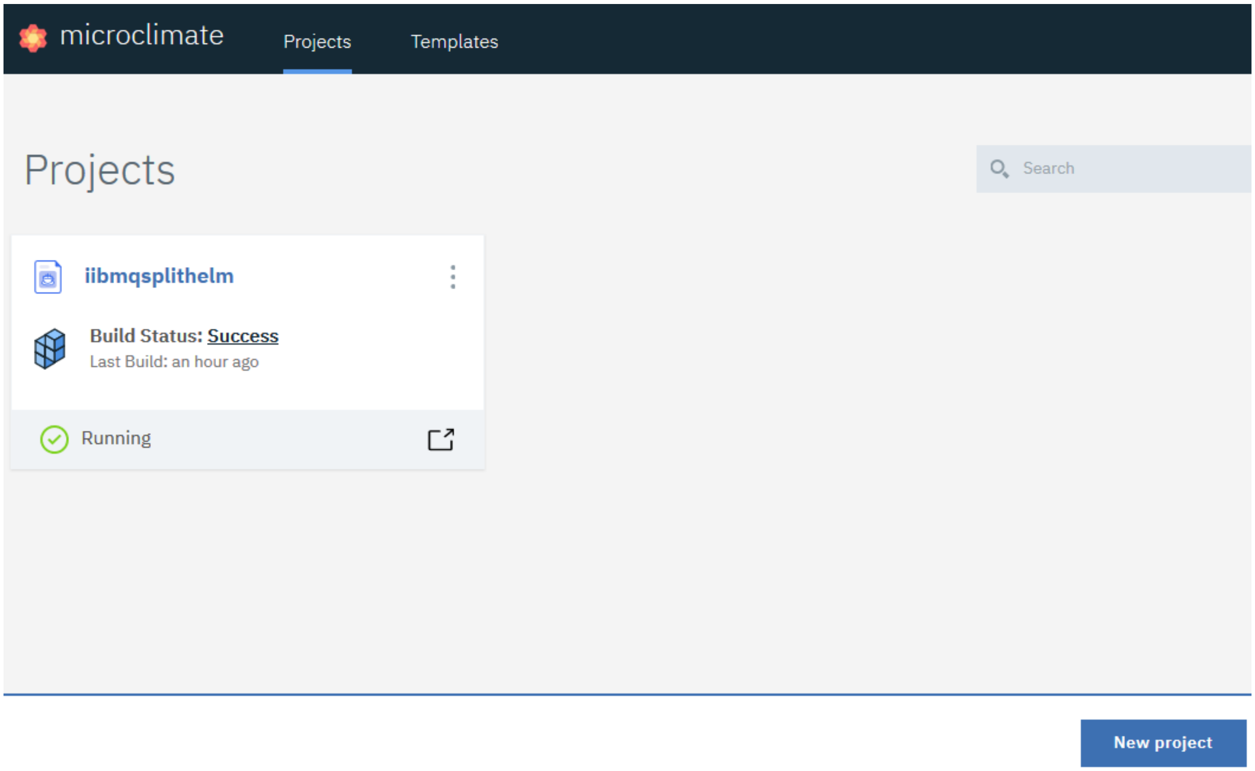
- Build and MQ Only image and load it onto the ICP image repository
- Perform a helm release delivering 4 containers
  - Two containers using the MQ Only image (built from this repos)
  - Two containers using the MQ with IIB image on ICP that we built in part 1

Create Microclimate project for the MQ image build and MQ-4Way Cluster Helm Release

Got to the Microclimate browser tab

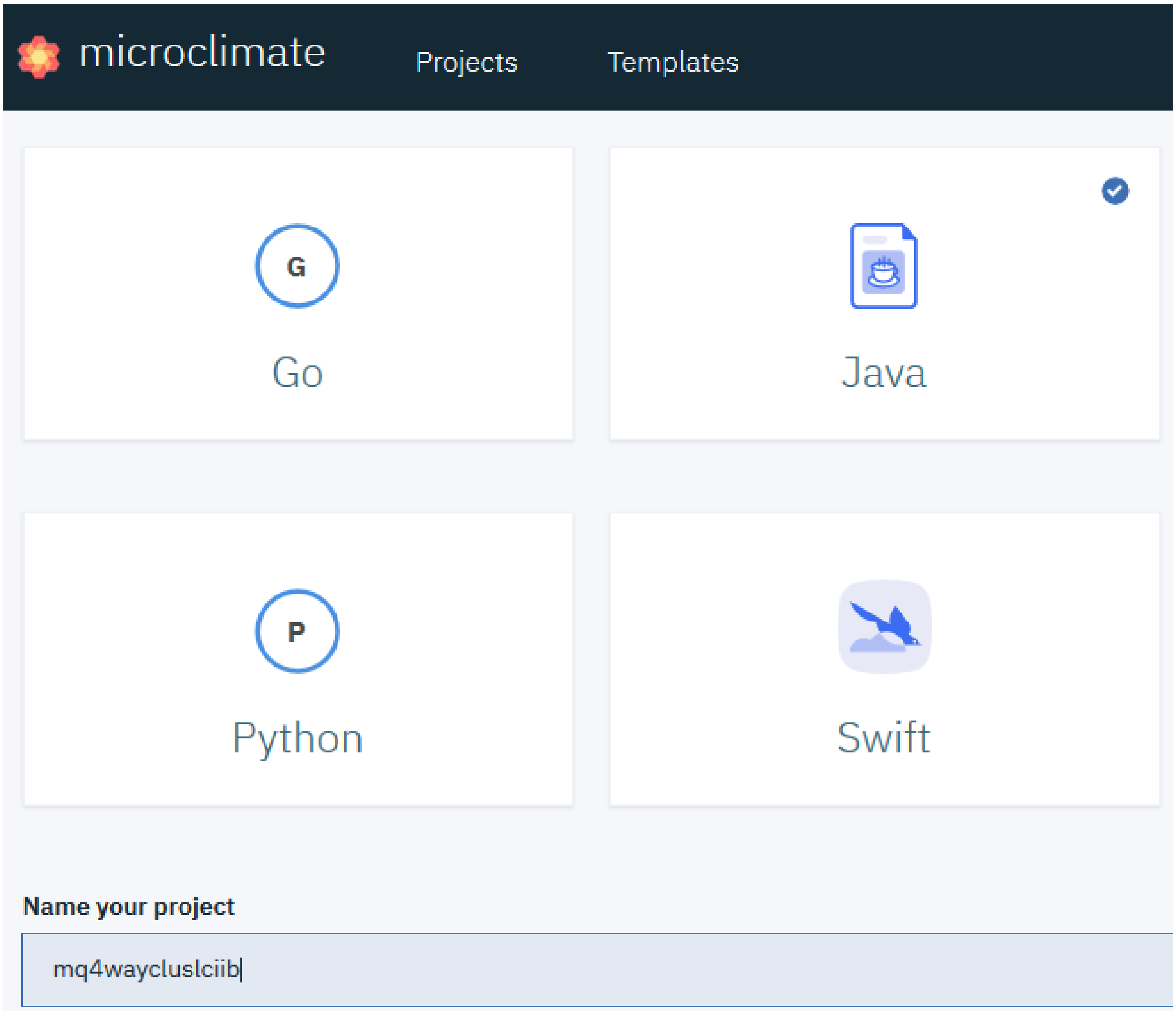


Click on Projects



Select New Project

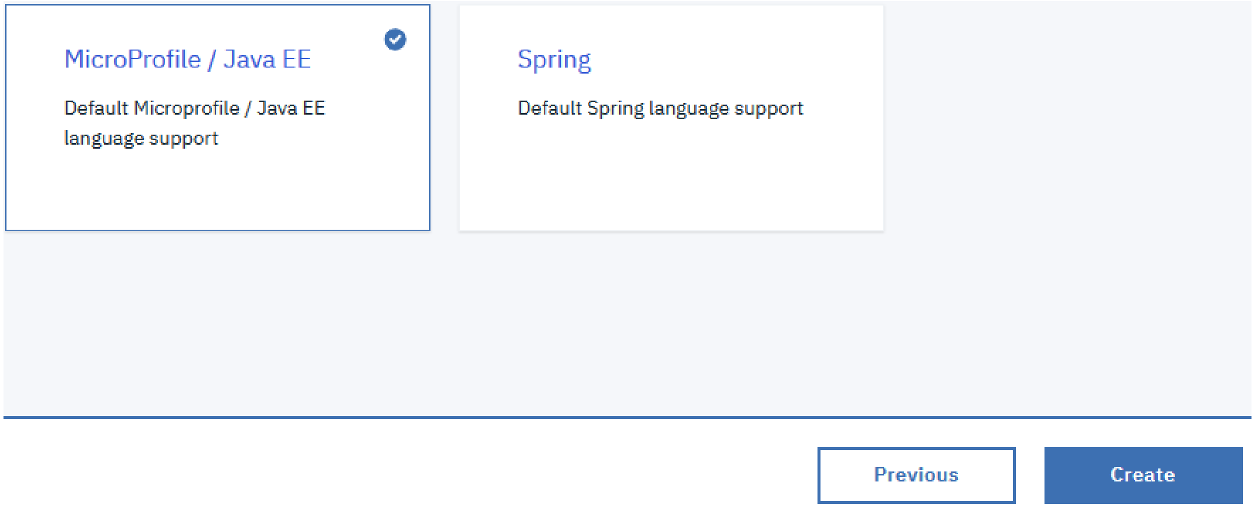
Select Java and enter the name for the project using a name that identifies the mq4wayclus-lciib repository



Name your project

mq4waycluslciib

Click next and select MicroProfile / Java EE



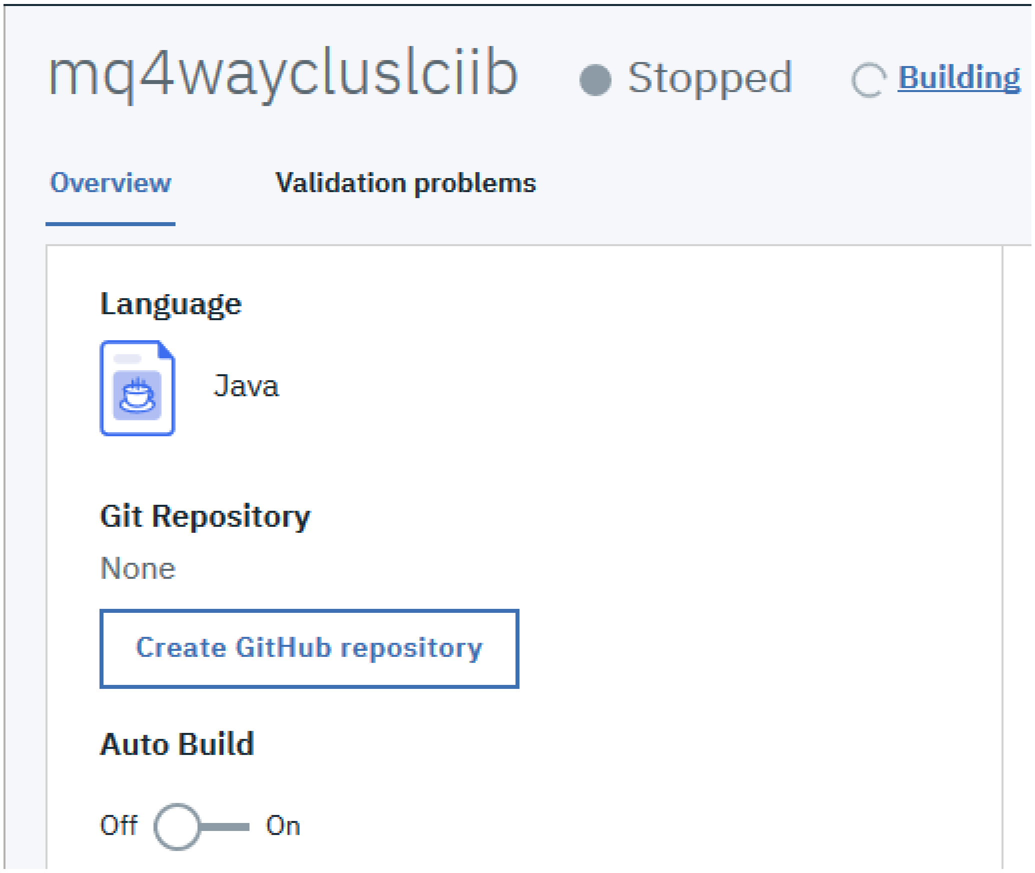
MicroProfile / Java EE ✓

Default Microprofile / Java EE language support

Spring

Default Spring language support

Click Create



mq4waycluslciib ● Stopped ◌ Building

Language

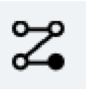
Java

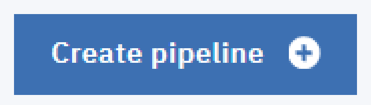
Git Repository

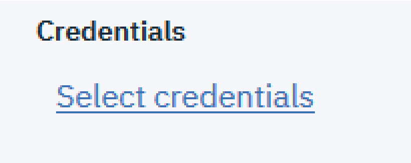
None

Auto Build

Off ○ On

click on the pipeline icon on the left hand side 

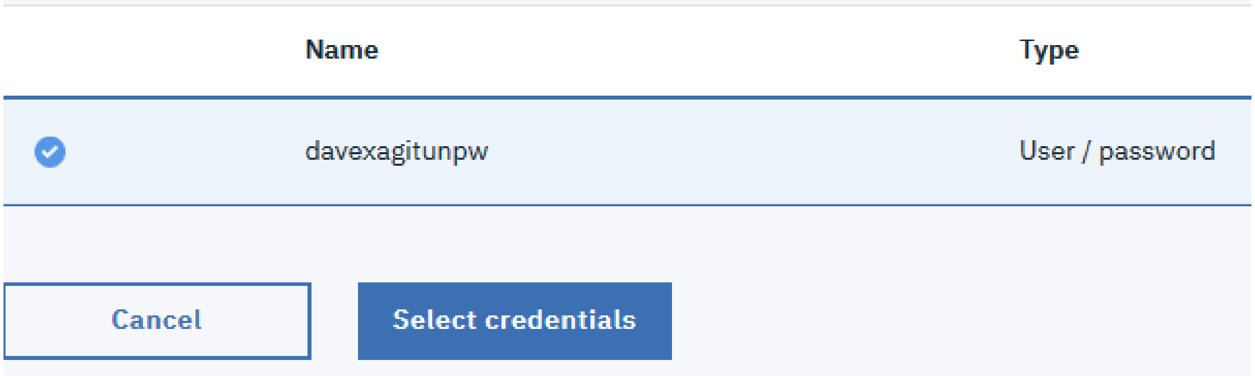
Click on create pipeline button 

click on select credentials 

Credentials

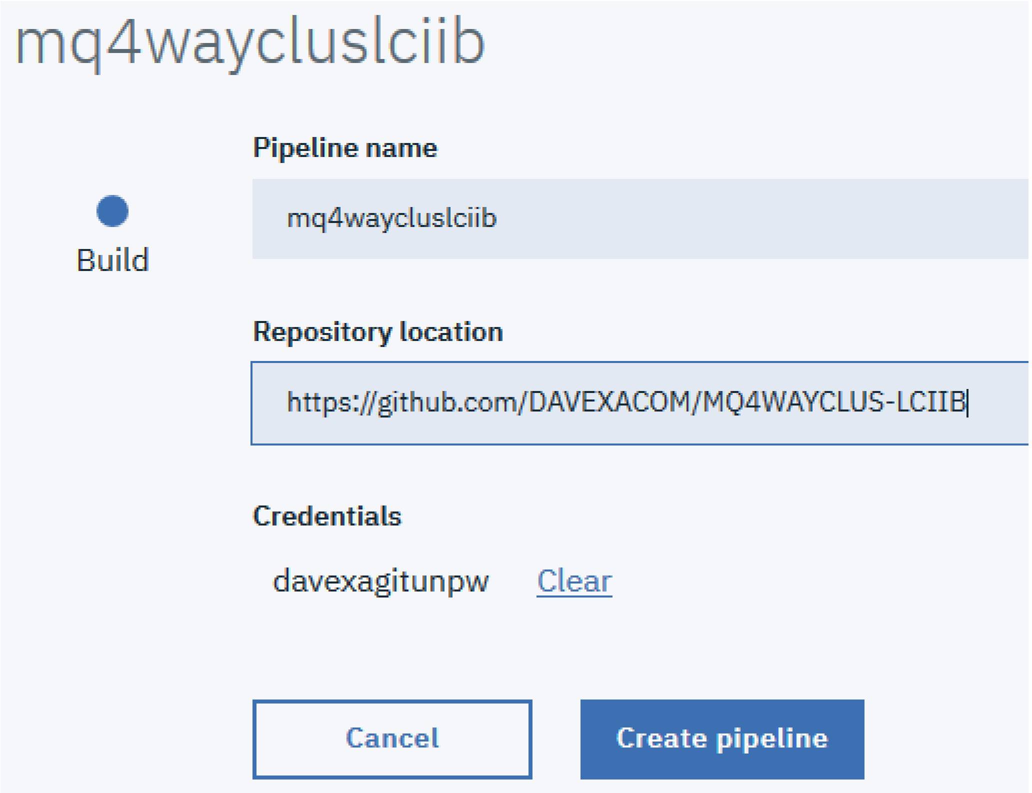
select the credentials you set up in Part 1. In this exampe davexagitunpw user/password

Hit Select credentials



	Name	Type
✓	davexagitunpw	User / password

Fill in the pipeline name and repos location and select create the pipeline



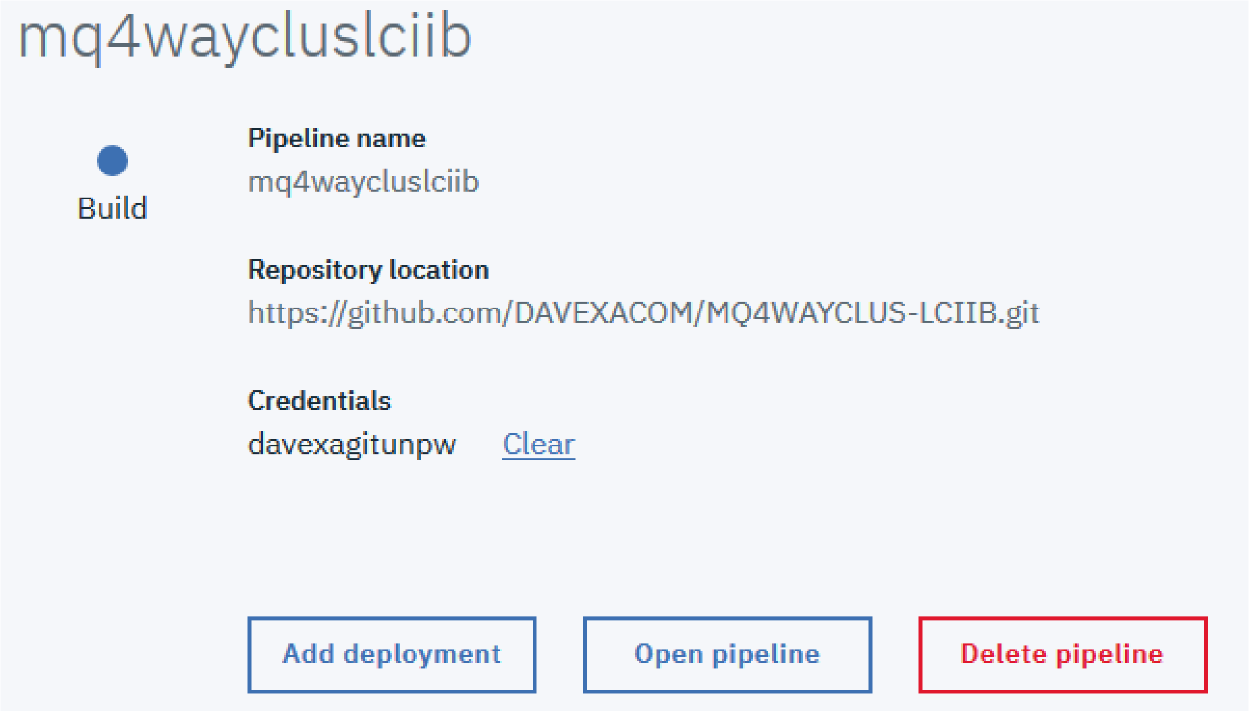
mq4wayclusIciib

Pipeline name: mq4wayclusIciib

Repository location: https://github.com/DAVEXACOM/MQ4WAYCLUS-LCIIB

Credentials: davexagitunpw

Open the pipeline by clicking on open pipeline



mq4wayclusIciib

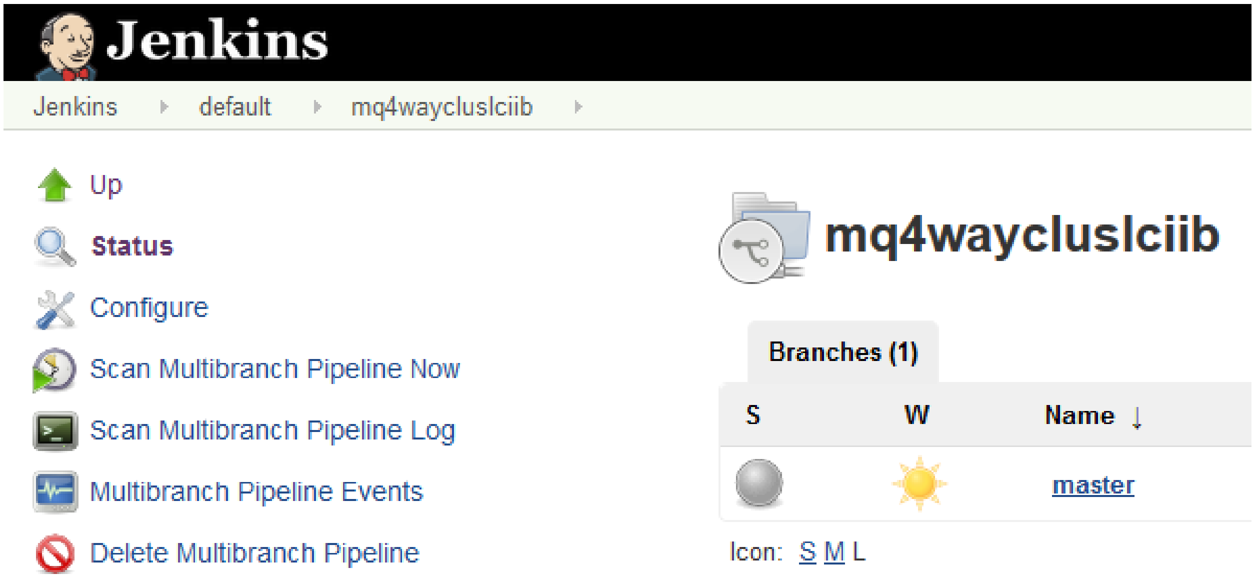
● Build

Pipeline name
mq4wayclusIciib

Repository location
https://github.com/DAVEXACOM/MQ4WAYCLUS-LCIIB.git

Credentials
davexagitunpw Clear

This will launch a new Jenkins client browser window



Jenkins
Jenkins ▸ default ▸ mq4wayclusIciib ▸
Up
Status
Configure
Scan Multibranch Pipeline Now
Scan Multibranch Pipeline Log
Multibranch Pipeline Events
Delete Multibranch Pipeline

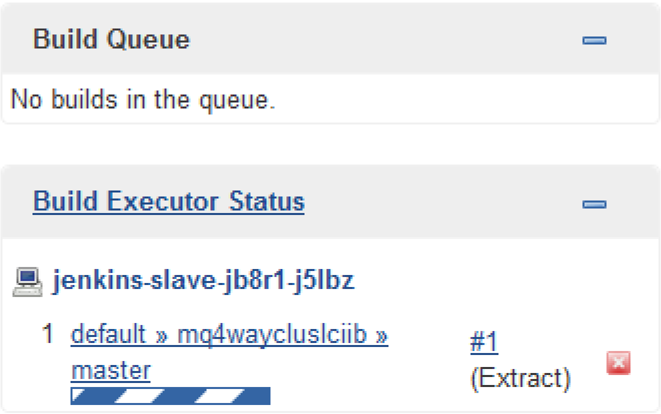
mq4wayclusIciib

Branches (1)

S	W	Name ↓
●	☀	master

Icon: S M L

Page down and you will see a build is initiated (occasionally you need to wait for an available slave)



Build Queue

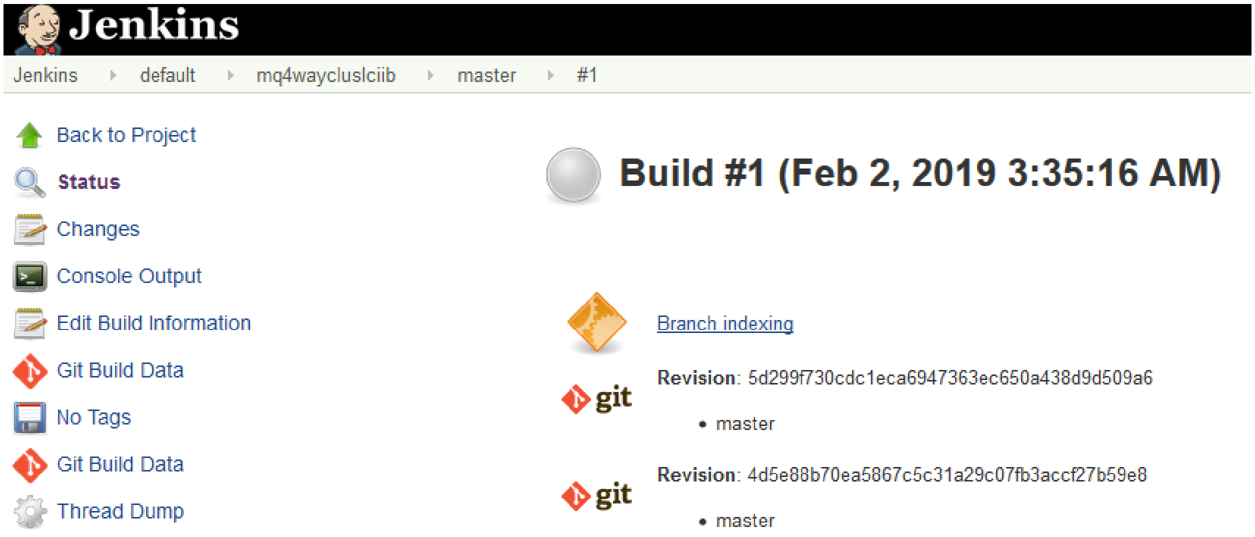
No builds in the queue.

Build Executor Status

jenkins-slave-jb8r1-j5lbz

1 default » mq4waycluslciib » master #1 (Extract)

Click on the #1 to follow the initial build and deploy



Build #1 (Feb 2, 2019 3:35:16 AM)

Branch indexing

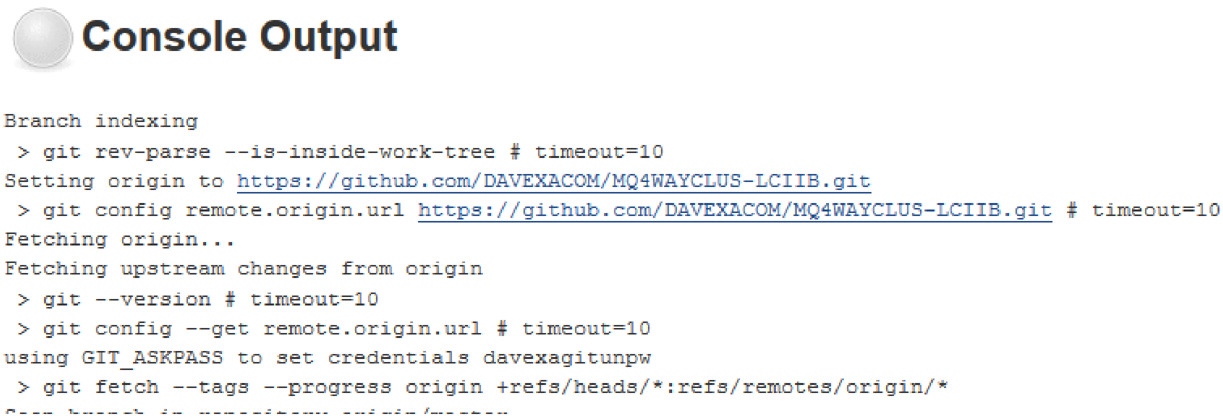
Revision: 5d299f730cdc1eca6947363ec650a438d9d509a6

- master

Revision: 4d5e88b70ea5867c5c31a29c07fb3accf27b59e8

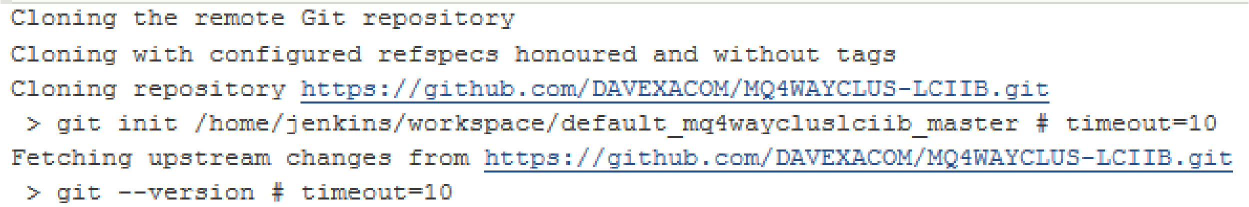
- master

Click on console Output and review the build. I have captured portions of a successful build and release log for your reference.

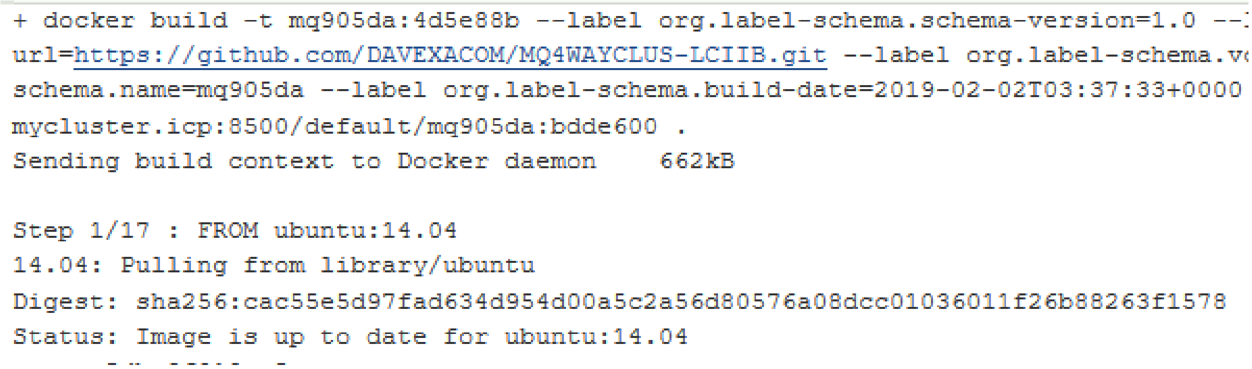


Console Output

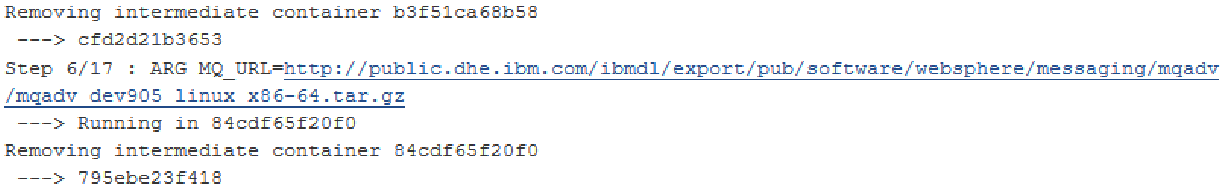
```
Branch indexing
 > git rev-parse --is-inside-work-tree # timeout=10
Setting origin to https://github.com/DAVEXACOM/MQ4WAYCLUS-LCIIB.git
 > git config remote.origin.url https://github.com/DAVEXACOM/MQ4WAYCLUS-LCIIB.git # timeout=10
Fetching origin...
Fetching upstream changes from origin
 > git --version # timeout=10
 > git config --get remote.origin.url # timeout=10
using GIT_ASKPASS to set credentials davexagitunpw
 > git fetch --tags --progress origin +refs/heads/*:refs/remotes/origin/*
Seen branch in repository origin/master
```



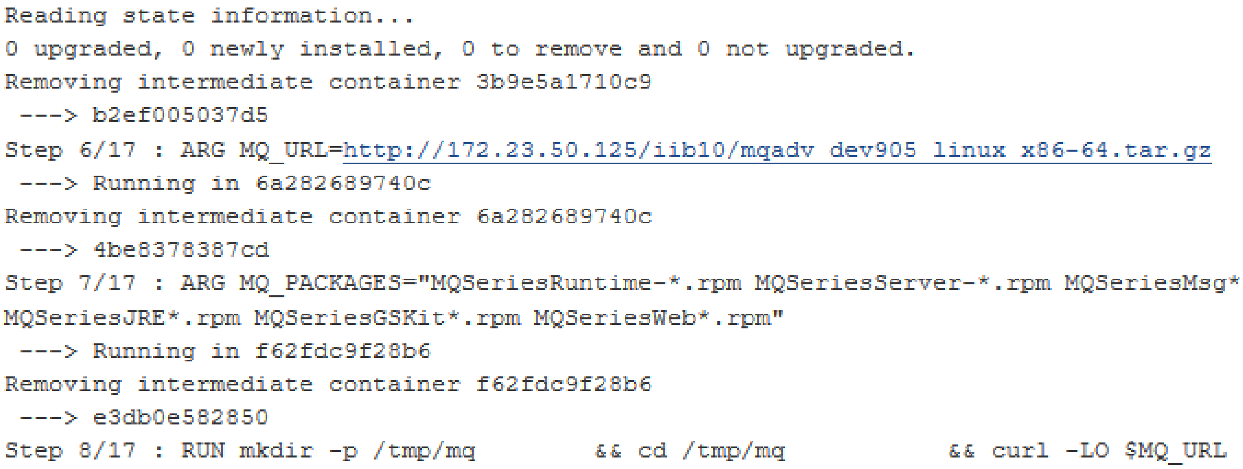
```
Cloning the remote Git repository
Cloning with configured refspecs honoured and without tags
Cloning repository https://github.com/DAVEXACOM/MQ4WAYCLUS-LCIIB.git
 > git init /home/jenkins/workspace/default_mq4waycluslciib_master # timeout=10
Fetching upstream changes from https://github.com/DAVEXACOM/MQ4WAYCLUS-LCIIB.git
 > git --version # timeout=10
```



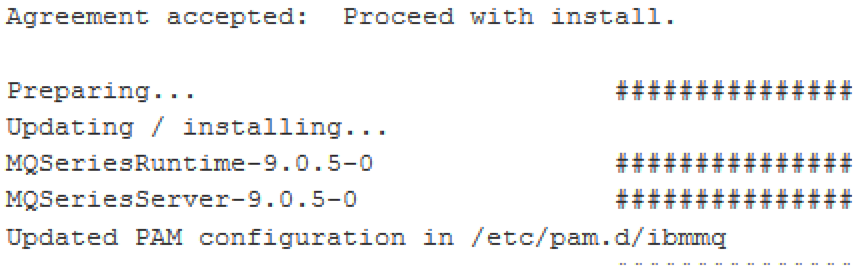
```
+ docker build -t mq905da:4d5e88b --label org.label-schema.schema-version=1.0 --
url=https://github.com/DAVEXACOM/MQ4WAYCLUS-LCIIB.git --label org.label-schema.v
schema.name=mq905da --label org.label-schema.build-date=2019-02-02T03:37:33+0000
mycluster.icp:8500/default/mq905da:bdde600 .
Sending build context to Docker daemon    662kB

Step 1/17 : FROM ubuntu:14.04
14.04: Pulling from library/ubuntu
Digest: sha256:cac55e5d97fad634d954d00a5c2a56d80576a08dcc01036011f26b88263f1578
Status: Image is up to date for ubuntu:14.04
```



```
Removing intermediate container b3f51ca68b58
 ---> cfd2d21b3653
Step 6/17 : ARG MQ_URL=http://public.dhe.ibm.com/ibmdl/export/pub/software/websphere/messaging/mqadv
/mqadv_dev905_linux_x86-64.tar.gz
 ---> Running in 84cdf65f20f0
Removing intermediate container 84cdf65f20f0
 ---> 795ebe23f418
```



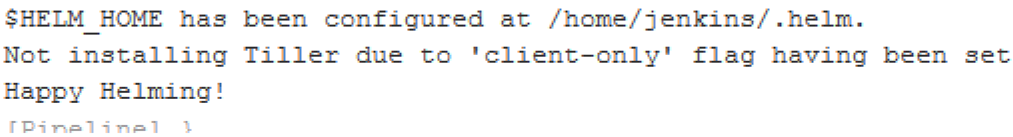
```
Reading state information...
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Removing intermediate container 3b9e5a1710c9
 ---> b2ef005037d5
Step 6/17 : ARG MQ_URL=http://172.23.50.125/iib10/mqadv_dev905_linux_x86-64.tar.gz
 ---> Running in 6a282689740c
Removing intermediate container 6a282689740c
 ---> 4be8378387cd
Step 7/17 : ARG MQ_PACKAGES="MQSeriesRuntime-*.rpm MQSeriesServer-*.rpm MQSeriesMsg*
MQSeriesJRE*.rpm MQSeriesGSKit*.rpm MQSeriesWeb*.rpm"
 ---> Running in f62fdc9f28b6
Removing intermediate container f62fdc9f28b6
 ---> e3db0e582850
Step 8/17 : RUN mkdir -p /tmp/mq        && cd /tmp/mq           && curl -LO $MQ_URL
```



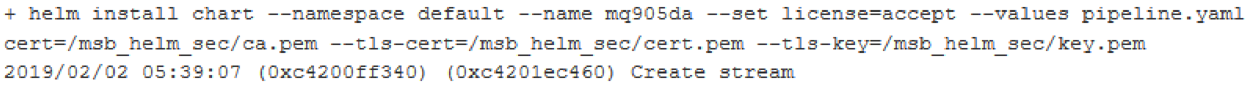
```
Agreement accepted:  Proceed with install.

Preparing...                          ###############
Updating / installing...
MQSeriesRuntime-9.0.5-0               ###############
MQSeriesServer-9.0.5-0                ###############
Updated PAM configuration in /etc/pam.d/ibmmq
```

```
Step 10/17 : COPY *.mqsc /etc/mqm/
 ---> 968b159d669d
Step 11/17 : COPY admin.json /etc/mqm/
 ---> 79106cea3bfd
Step 12/17 : COPY mq-dev-config /etc/mqm/mq-dev-config
 ---> f93b13f35125
Step 13/17 : RUN chmod +x /usr/local/bin/*.sh
 ---> Running in d0a65c0a4791
Removing intermediate container d0a65c0a4791
 ---> fa635049ec64
Step 14/17 : EXPOSE 1414 9443
 ---> Running in 7c5803bbd879
Removing intermediate container 7c5803bbd879
 ---> 883314c93d41
Step 15/17 : ENV LANG=en_US.UTF-8
 ---> Running in 1fb14fb31ca5
Removing intermediate container 1fb14fb31ca5
 ---> 4a043025c3ca
Step 16/17 : ENTRYPOINT ["mq.sh"]
 ---> Running in 2d565cc6e483
Removing intermediate container 2d565cc6e483
 ---> e06070646a95
Step 17/17 : LABEL "org.label-schema.build-date"='2019-02-02T05:35:15+0000'
"org.label-schema.schema-version"='1.0' "org.label-schema.vcs-ref"='e438635'
url"='https://github.com/DAVEXACOM/MQ4WAYCLUS-LCIIB.git'
 ---> Running in 6eed32ac88ac
```

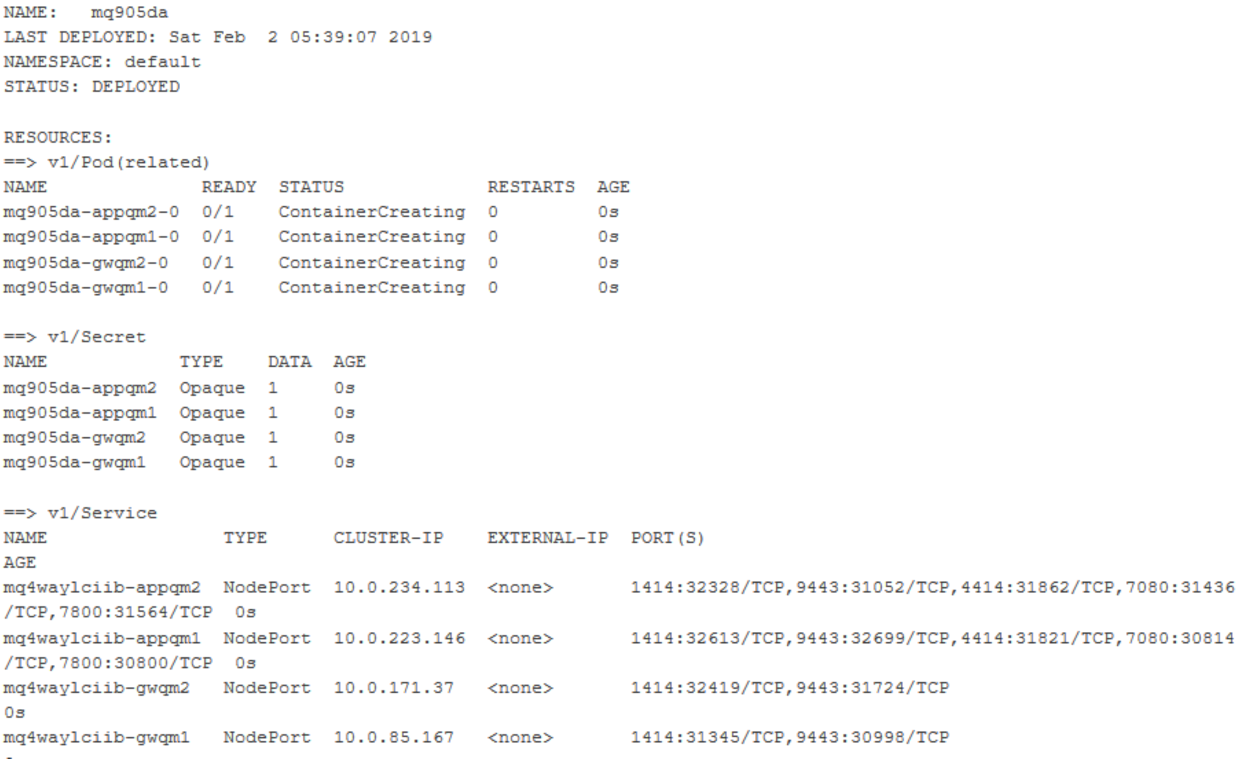
```
[Pipeline] sh
+ docker push mycluster.icp:8500/default/mq905da:e438635
The push refers to repository [mycluster.icp:8500/default/mq905da]
d7225b353de8: Preparing
d24fb6c93d29: Preparing

User defined chart location specified: chart
[Pipeline] sh
+ echo image:
  repository: mycluster.icp:8500/default/mq905da
  tag: "e438635"
global:
  image:
    repository: mycluster.icp:8500/default/mq905da
    tag: "e438635"
```



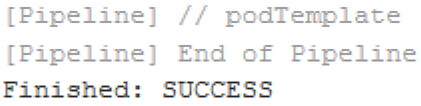
```
$HELM_HOME has been configured at /home/jenkins/.helm.
Not installing Tiller due to 'client-only' flag having been set
Happy Helming!
[Pipeline] }
```



```
+ helm install chart --namespace default --name mq905da --set license=accept --values pipeline.yaml
cert=/msb_helm_sec/ca.pem --tls-cert=/msb_helm_sec/cert.pem --tls-key=/msb_helm_sec/key.pem
2019/02/02 05:39:07 (0xc4200ff340) (0xc4201ec460) Create stream
```



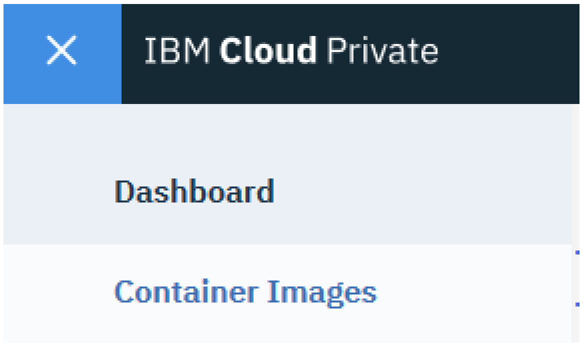
```
NAME:   mq905da
LAST DEPLOYED: Sat Feb  2 05:39:07 2019
NAMESPACE: default
STATUS: DEPLOYED

RESOURCES:
==> v1/Pod(related)
NAME              READY  STATUS             RESTARTS  AGE
mq905da-appqm2-0  0/1    ContainerCreating  0         0s
mq905da-appqm1-0  0/1    ContainerCreating  0         0s
mq905da-gwqm2-0   0/1    ContainerCreating  0         0s
mq905da-gwqm1-0   0/1    ContainerCreating  0         0s

==> v1/Secret
NAME            TYPE    DATA  AGE
mq905da-appqm2  Opaque  1     0s
mq905da-appqm1  Opaque  1     0s
mq905da-gwqm2   Opaque  1     0s
mq905da-gwqm1   Opaque  1     0s

==> v1/Service
NAME                  TYPE      CLUSTER-IP    EXTERNAL-IP  PORT(S)
AGE
mq4waylciib-appqm2  NodePort  10.0.234.113  <none>       1414:32328/TCP,9443:31052/TCP,4414:31862/TCP,7080:31436
/TCP,7800:31564/TCP  0s
mq4waylciib-appqm1  NodePort  10.0.223.146  <none>       1414:32613/TCP,9443:32699/TCP,4414:31821/TCP,7080:30814
/TCP,7800:30800/TCP  0s
mq4waylciib-gwqm2   NodePort  10.0.171.37   <none>       1414:32419/TCP,9443:31724/TCP
0s
mq4waylciib-gwqm1   NodePort  10.0.85.167   <none>       1414:31345/TCP,9443:30998/TCP
-
```



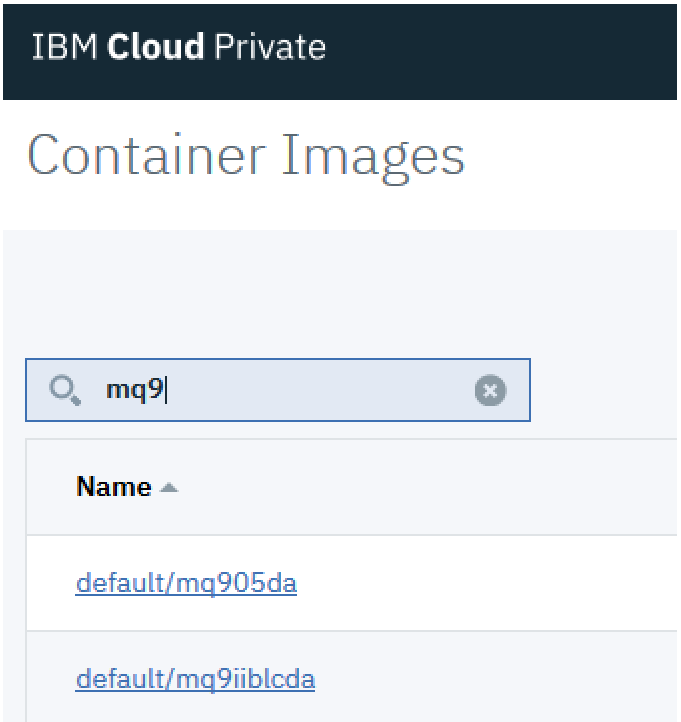
```
[Pipeline] // podTemplate
[Pipeline] End of Pipeline
Finished: SUCCESS
```

Checking the results of MQ4WAYCLUS-IIBLC build on ICP

From the ICP Console select Container Images



Filter on mq9



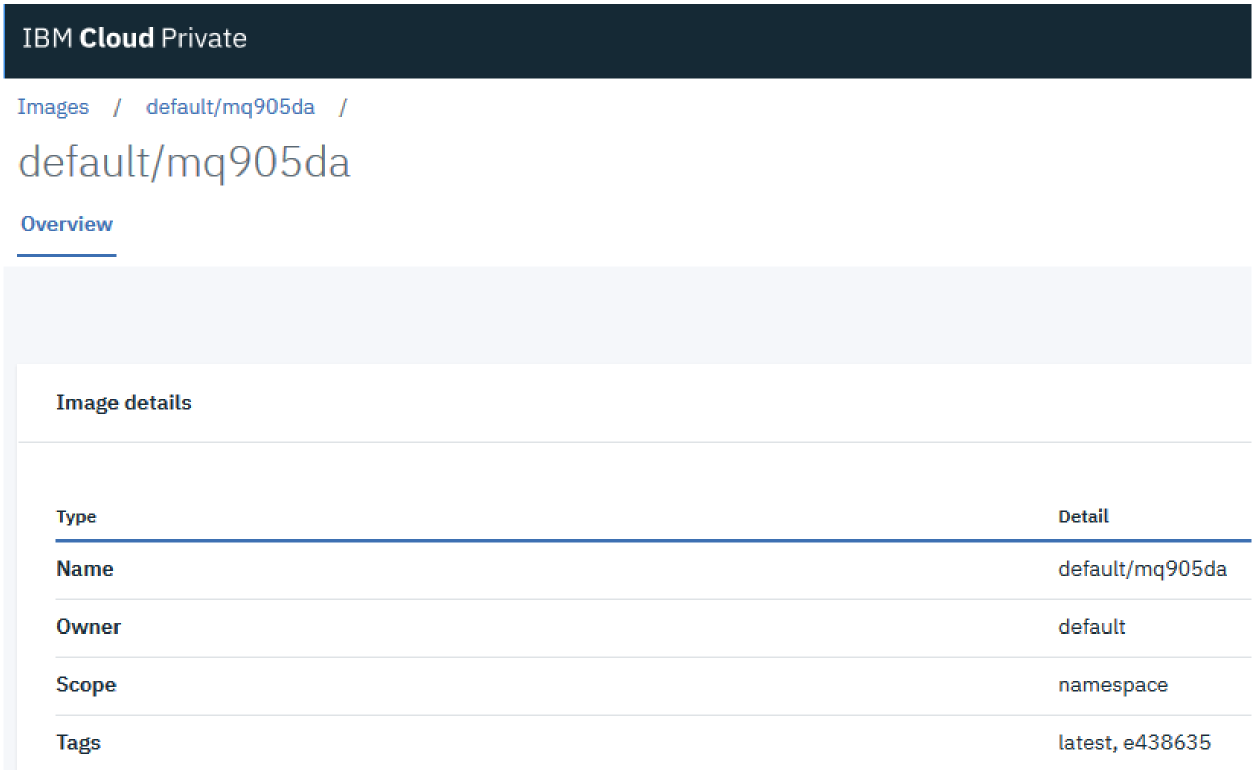
IBM **Cloud** Private

Container Images

mq9

Name ▲
default/mq905da
default/mq9iiblcda

Click on the link for default/mq905da and check out the image details



IBM **Cloud** Private

Images / default/mq905da /

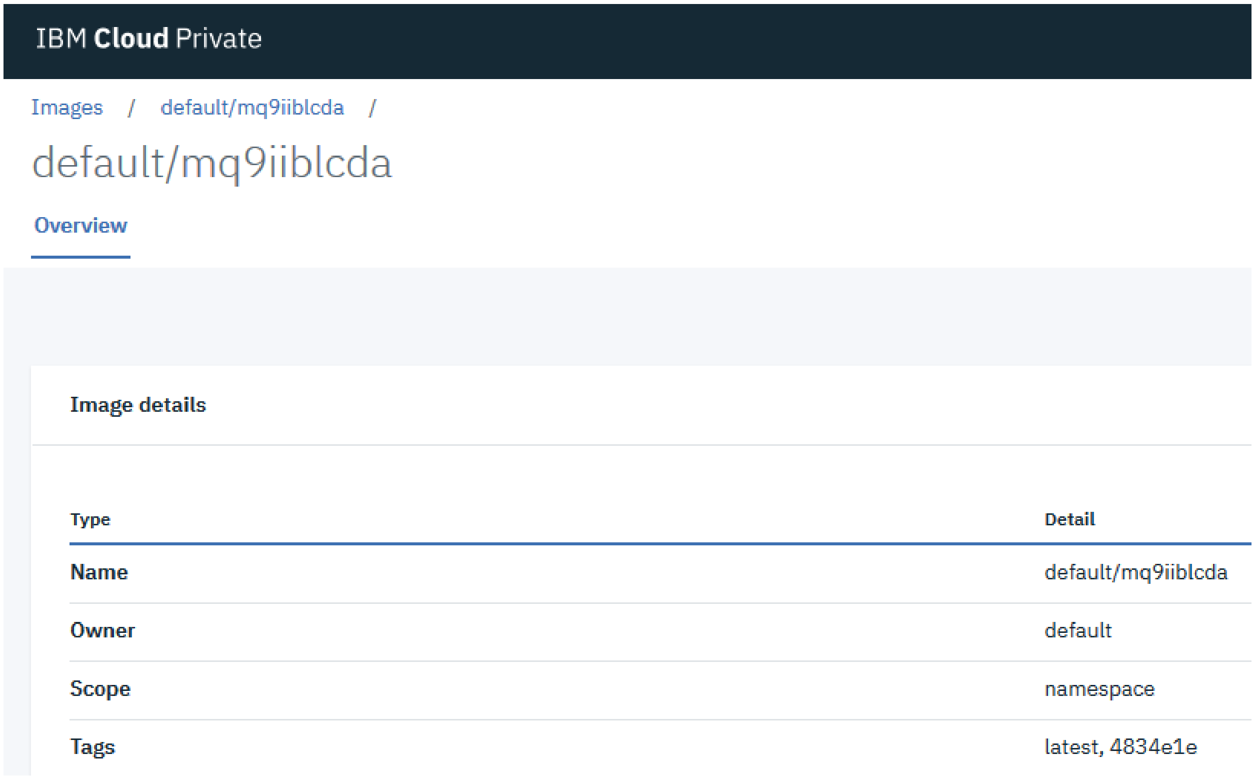
default/mq905da

Overview

Image details

Type	Detail
Name	default/mq905da
Owner	default
Scope	namespace
Tags	latest, e438635

Click on the link for mq9iiblcda (image build and loaded in the first part of the document)


IBM **Cloud** Private

Images / default/mq9iiblcda /

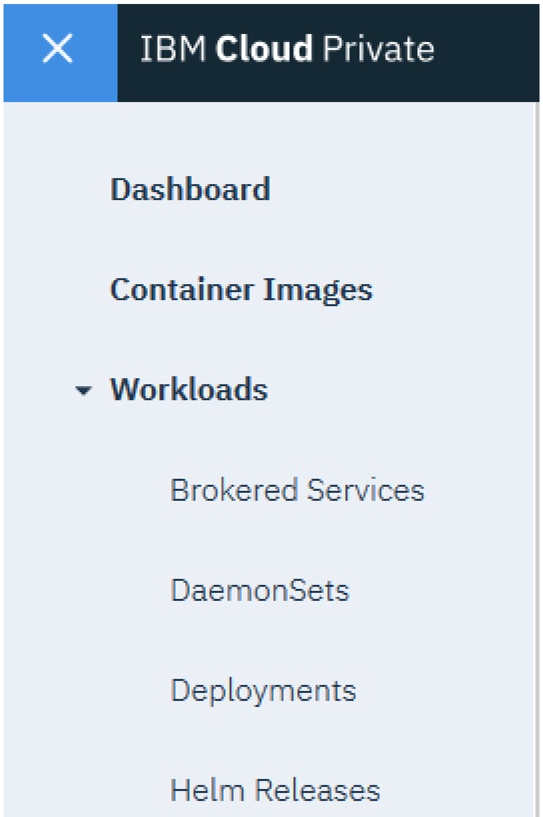
default/mq9iiblcda

Overview

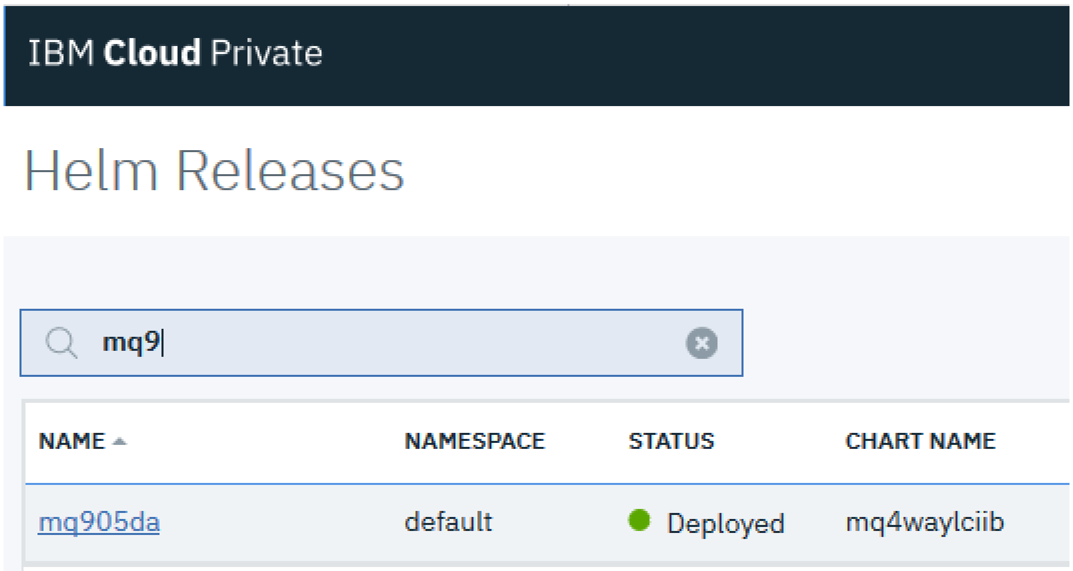
Image details

Type	Detail
Name	default/mq9iiblcda
Owner	default
Scope	namespace
Tags	latest, 4834e1e

Now lets take a look at the Helm Release. From IBM Cloud Private select Workloads-> Helm releases



Select Helm Releases and use the Search filter with mq9 to find mq905da



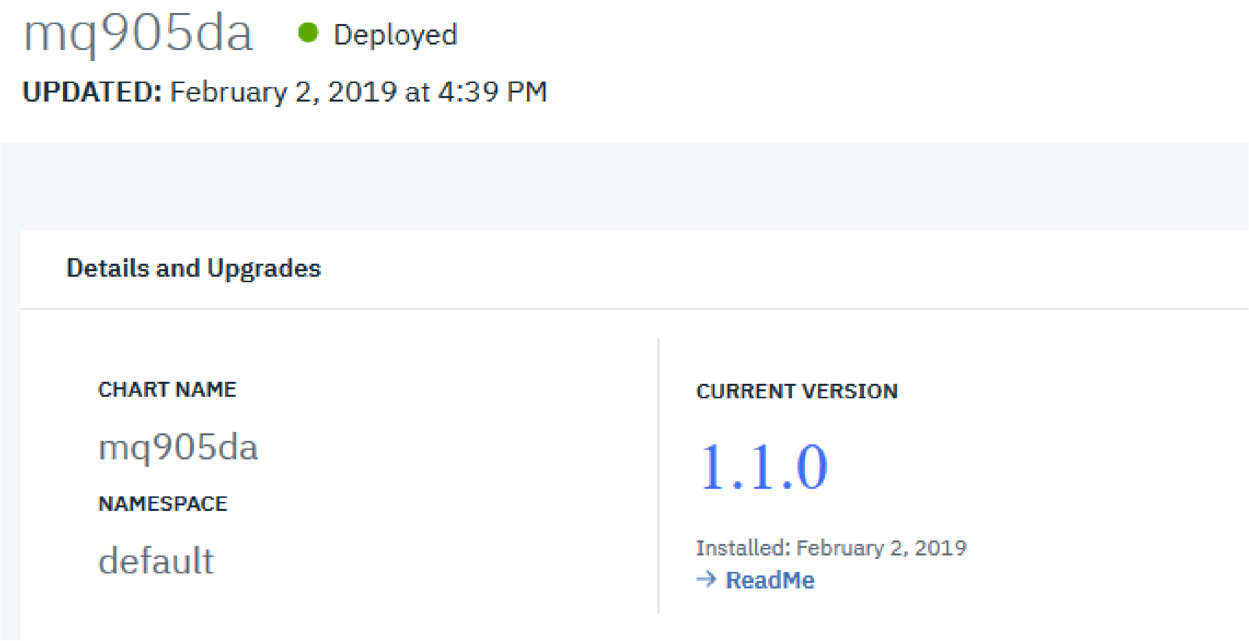
IBM **Cloud** Private

Helm Releases

mq9

NAME	NAMESPACE	STATUS	CHART NAME
mq905da	default	● Deployed	mq4waylciib

Click on mp905da link and explore the release.



mq905da ● Deployed

UPDATED: February 2, 2019 at 4:39 PM

Details and Upgrades

CHART NAME

mq905da

NAMESPACE

default

CURRENT VERSION

1.1.0

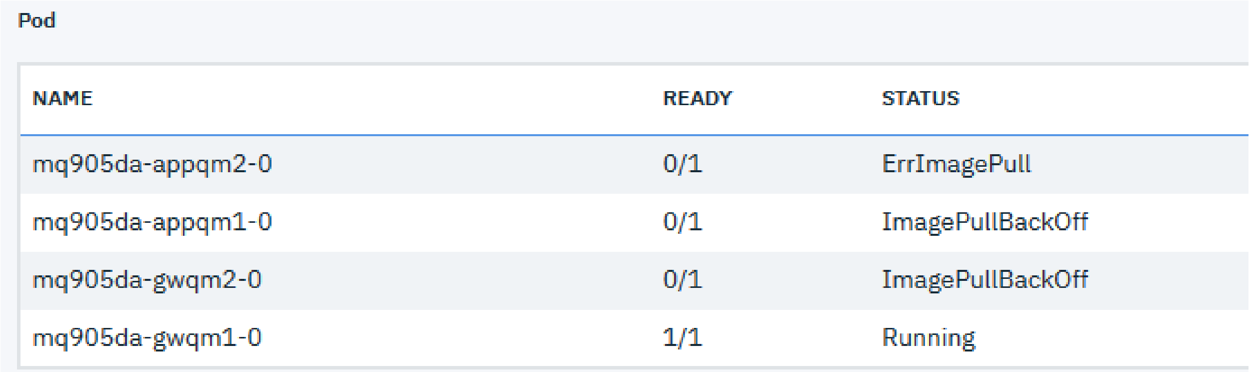
Installed: February 2, 2019

→ ReadMe

Scroll down and you'll see the 4 pods each housing a container running GWQM1,GWQM2, APPQM1 and APPQM2

Now for this first build and deploy of the Helm release we may see all but the GWQM1 pod fail. The reason for this is the tagging of the images.

You may see the following



Pod

NAME	READY	STATUS
mq905da-appqm2-0	0/1	ErrImagePull
mq905da-appqm1-0	0/1	ImagePullBackOff
mq905da-gwqm2-0	0/1	ImagePullBackOff
mq905da-gwqm1-0	1/1	Running

This could be for a number of reasons. Things to check:

The high level chart files in the repository are for GWQM1 this generally works because it has the container build associated with it and Jenkins gets involved and overrides the repository and the tag values rather than relying on those in the charts.

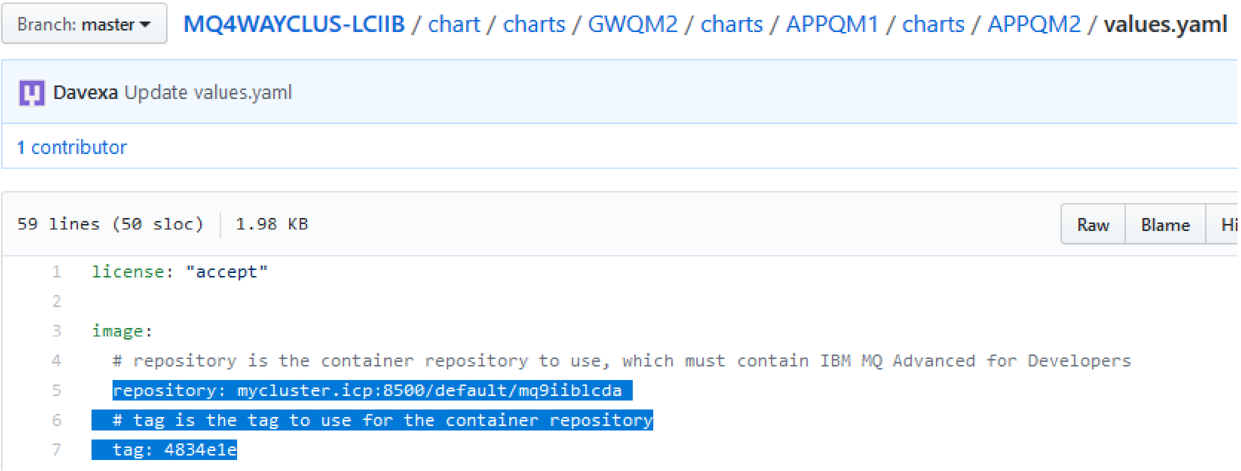
The others, GWQM2, APPQM1 and APPQM2 don't have this luxury so we need to ensure that the values.yaml for each really have the tag names and repository URLs that are correct.

If using the tag latest has not worked for APPQM1 and APPQM2 go back and put an actual tag name in their values.yaml file but most importantly check the repository URLs. (look out for icpcluster / mycluster). You can check your target cluster name on your instance of ICP using the client connection information available via the icon on the top right corner of the ICP Console.

For the GWQM2 MQ only image we did not know the tag name ahead of time as it was only build and tagged on the initial build, therefore this will need to be fixed up in it's values.yaml chart file.

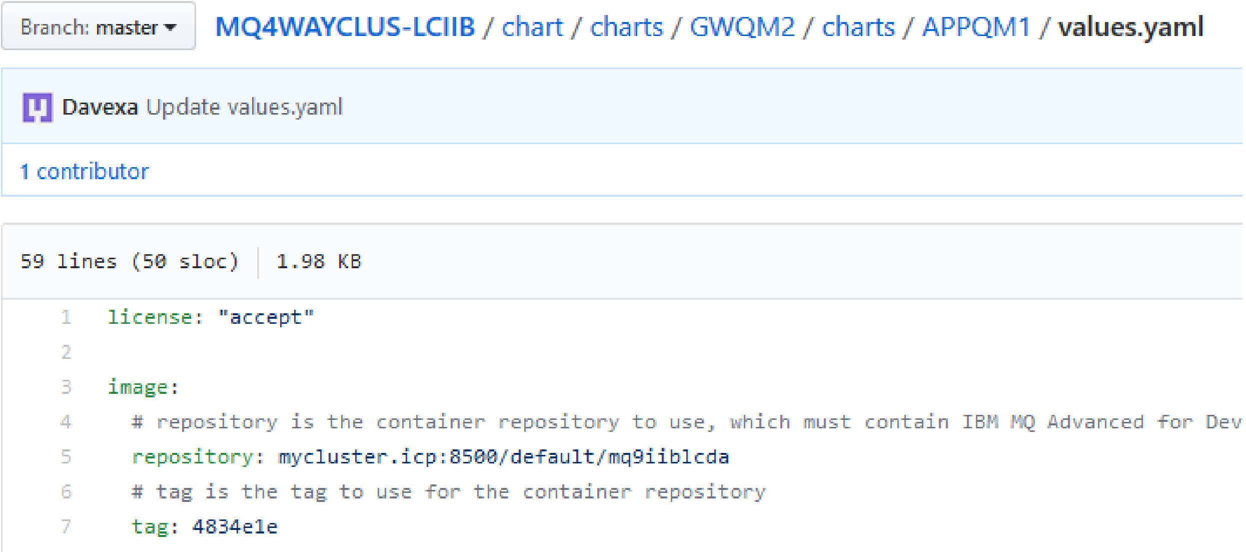
If your Jenkins pipeline script customization is correct in theory using latest in your values.yamls files should ensure these mismatches don't happen.

Go back and correct these and then run a new build from Jenkins if necessary



Branch: master
MQ4WAYCLUS-LCIIB / chart / charts / GWQM2 / charts / APPQM1 / charts / APPQM2 / values.yaml
Davexa Update values.yaml
1 contributor
59 lines (50 sloc) | 1.98 KB
Raw
Blame
Hi

```
license: "accept"

image:
  # repository is the container repository to use, which must contain IBM MQ Advanced for Developers
  repository: mycluster.icp:8500/default/mq9iiblcda
  # tag is the tag to use for the container repository
  tag: 4834e1e
```



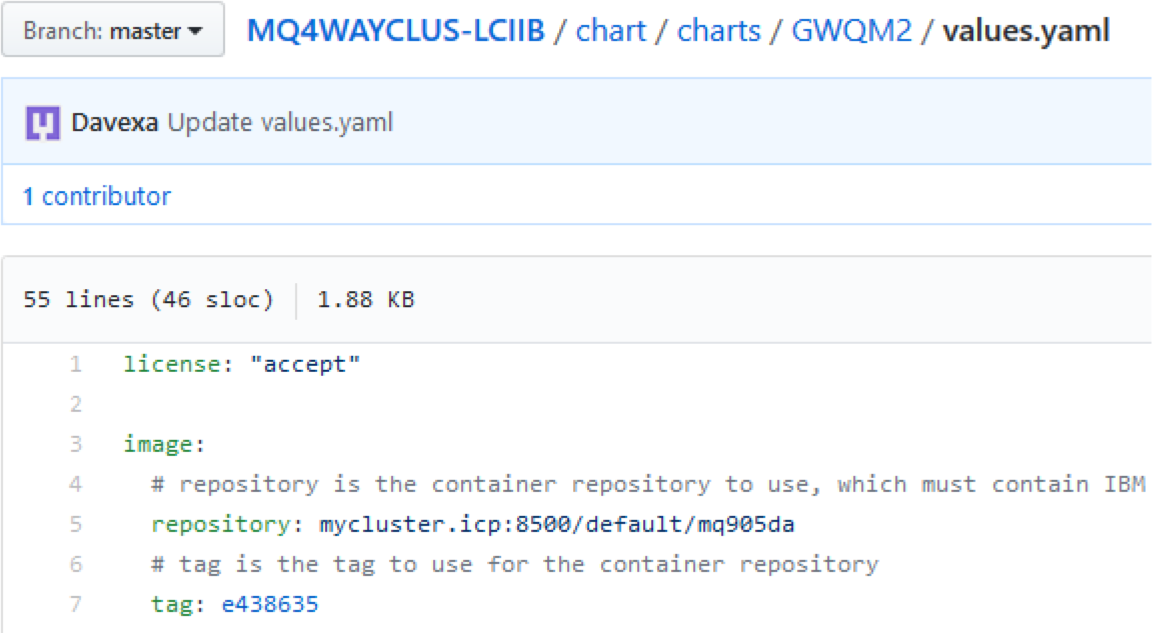
Branch: master ▾ MQ4WAYCLUS-LCIIB / chart / charts / GWQM2 / charts / APPQM1 / values.yaml

Davexa Update values.yaml

1 contributor

59 lines (50 sloc) | 1.98 KB

```
license: "accept"

image:
  # repository is the container repository to use, which must contain IBM MQ Advanced for Dev
  repository: mycluster.icp:8500/default/mq9iiblcda
  # tag is the tag to use for the container repository
  tag: 4834e1e
```



Branch: master ▾ MQ4WAYCLUS-LCIIB / chart / charts / GWQM2 / values.yaml

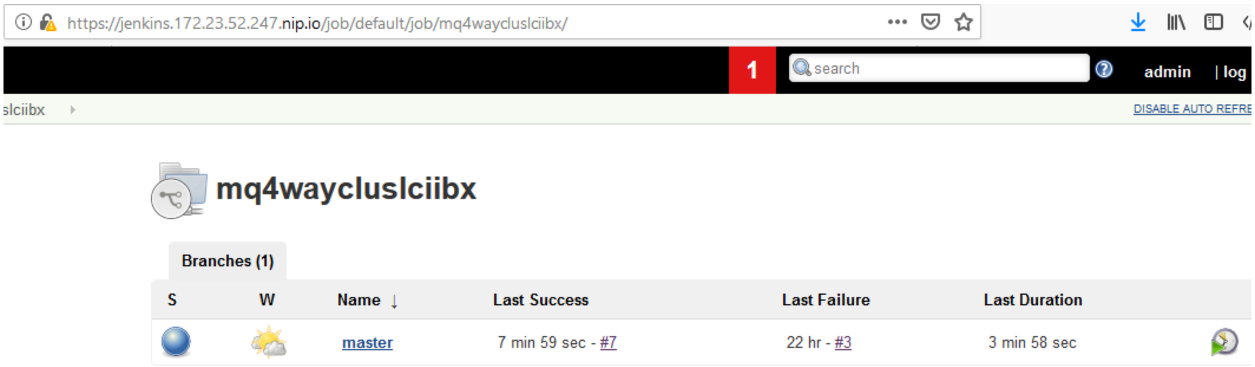
Davexa Update values.yaml

1 contributor

55 lines (46 sloc) | 1.88 KB

```
license: "accept"

image:
  # repository is the container repository to use, which must contain IBM
  repository: mycluster.icp:8500/default/mq905da
  # tag is the tag to use for the container repository
  tag: e438635
```

Re-run the Jenkins pipeline by clicking on the Circle with the green triangle.



https://jenkins.172.23.52.247.nip.io/job/default/job/mq4wayclusIciibx/
1
search
admin | log
slciibx
DISABLE AUTO REFRE

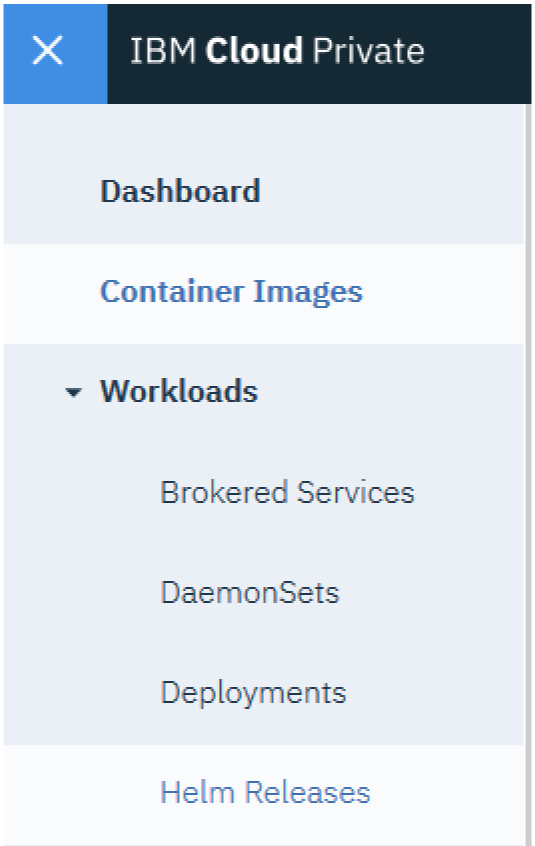
mq4wayclusIciibx

Branches (1)

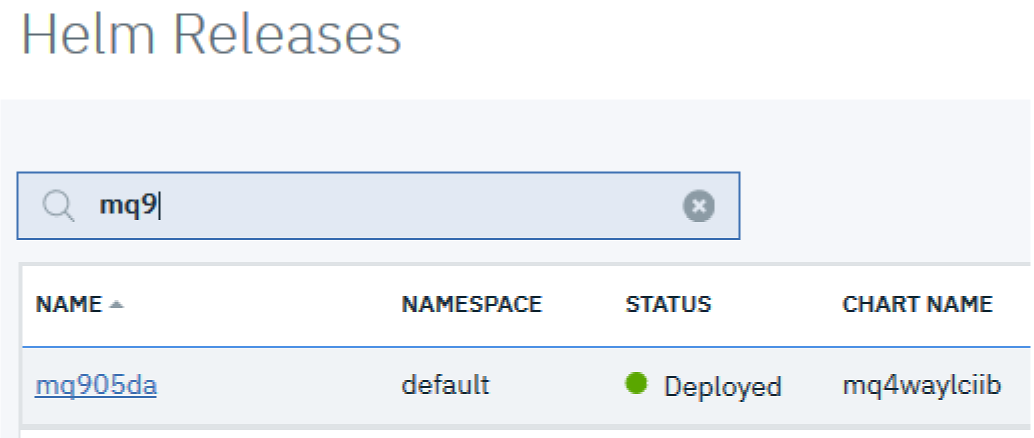
S	W	Name ↓	Last Success	Last Failure	Last Duration
●		master	7 min 59 sec - #7	22 hr - #3	3 min 58 sec

Validating the MQ4WAYCLUS-LCIIB release

On successful completion of the build and release. Check the Helm Release



Filter on mq9

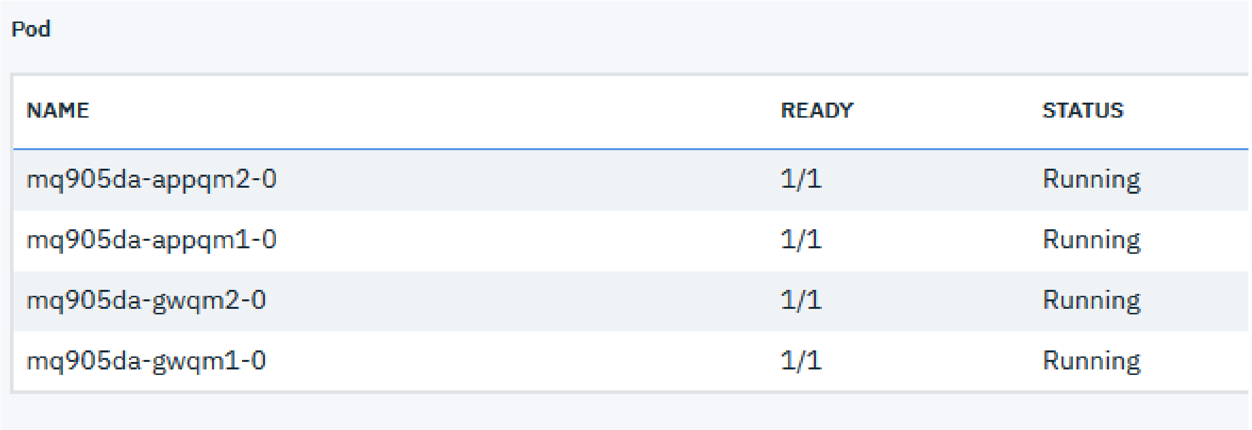

Helm Releases

mq9

NAME	NAMESPACE	STATUS	CHART NAME
mq905da	default	● Deployed	mq4waylciib

Click on mq905da

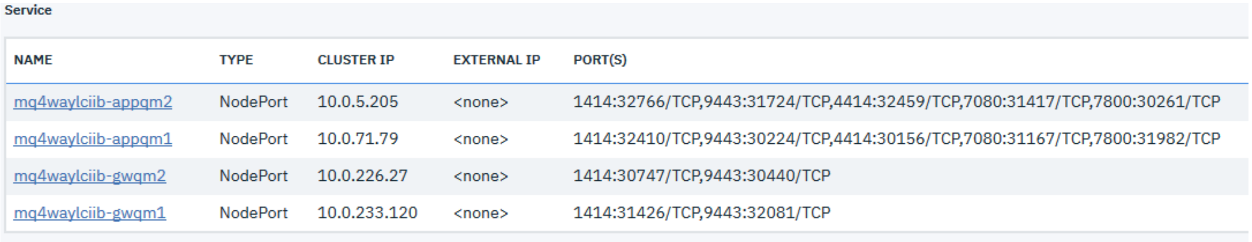
And if all is correct you should see all pods running this time.



Pod

NAME	READY	STATUS
mq905da-appqm2-0	1/1	Running
mq905da-appqm1-0	1/1	Running
mq905da-gwqm2-0	1/1	Running
mq905da-gwqm1-0	1/1	Running

Scroll down to services

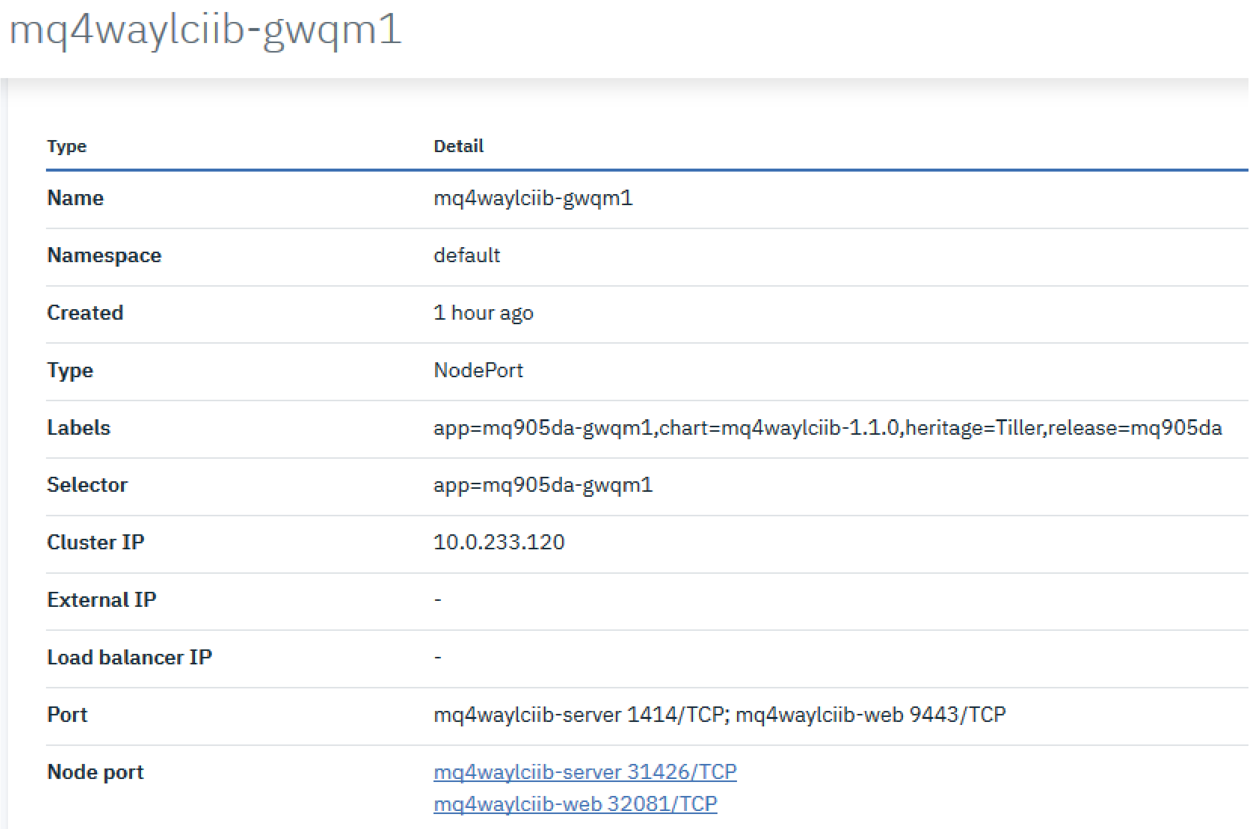


Service

NAME	TYPE	CLUSTER IP	EXTERNAL IP	PORT(S)
mq4waylciib-appqm2	NodePort	10.0.5.205	<none>	1414:32766/TCP,9443:31724/TCP,4414:32459/TCP,7080:31417/TCP,7800:30261/TCP
mq4waylciib-appqm1	NodePort	10.0.71.79	<none>	1414:32410/TCP,9443:30224/TCP,4414:30156/TCP,7080:31167/TCP,7800:31982/TCP
mq4waylciib-gwqm2	NodePort	10.0.226.27	<none>	1414:30747/TCP,9443:30440/TCP
mq4waylciib-gwqm1	NodePort	10.0.233.120	<none>	1414:31426/TCP,9443:32081/TCP

Testing GWQM1 MQ

Select GWQM1

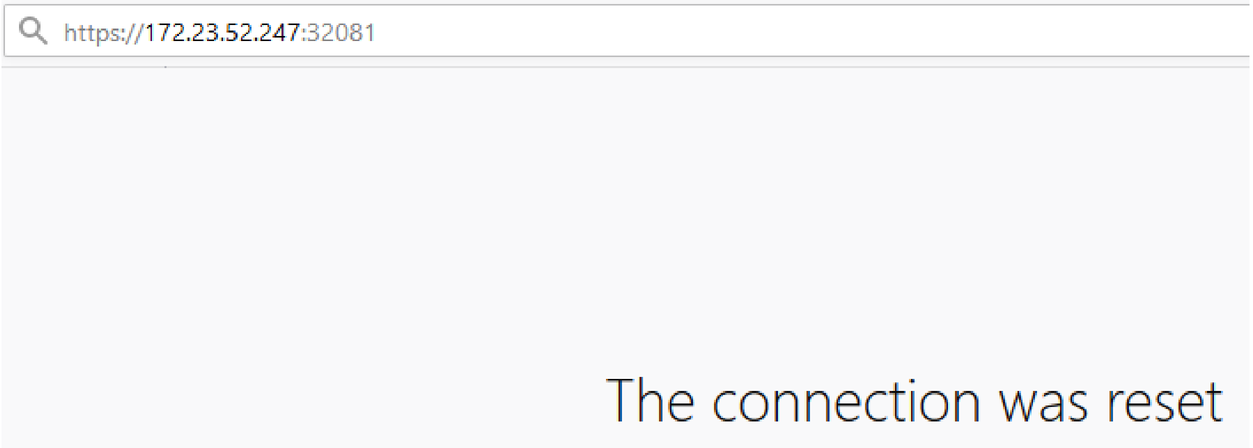


mq4waylciib-gwqm1

Type	Detail
Name	mq4waylciib-gwqm1
Namespace	default
Created	1 hour ago
Type	NodePort
Labels	app=mq905da-gwqm1,chart=mq4waylciib-1.1.0,heritage=Tiller,release=mq905da
Selector	app=mq905da-gwqm1
Cluster IP	10.0.233.120
External IP	-
Load balancer IP	-
Port	mq4waylciib-server 1414/TCP; mq4waylciib-web 9443/TCP
Node port	mq4waylciib-server 31426/TCP mq4waylciib-web 32081/TCP

Click on the web console link mq4waylciib-web nnnn/TCP

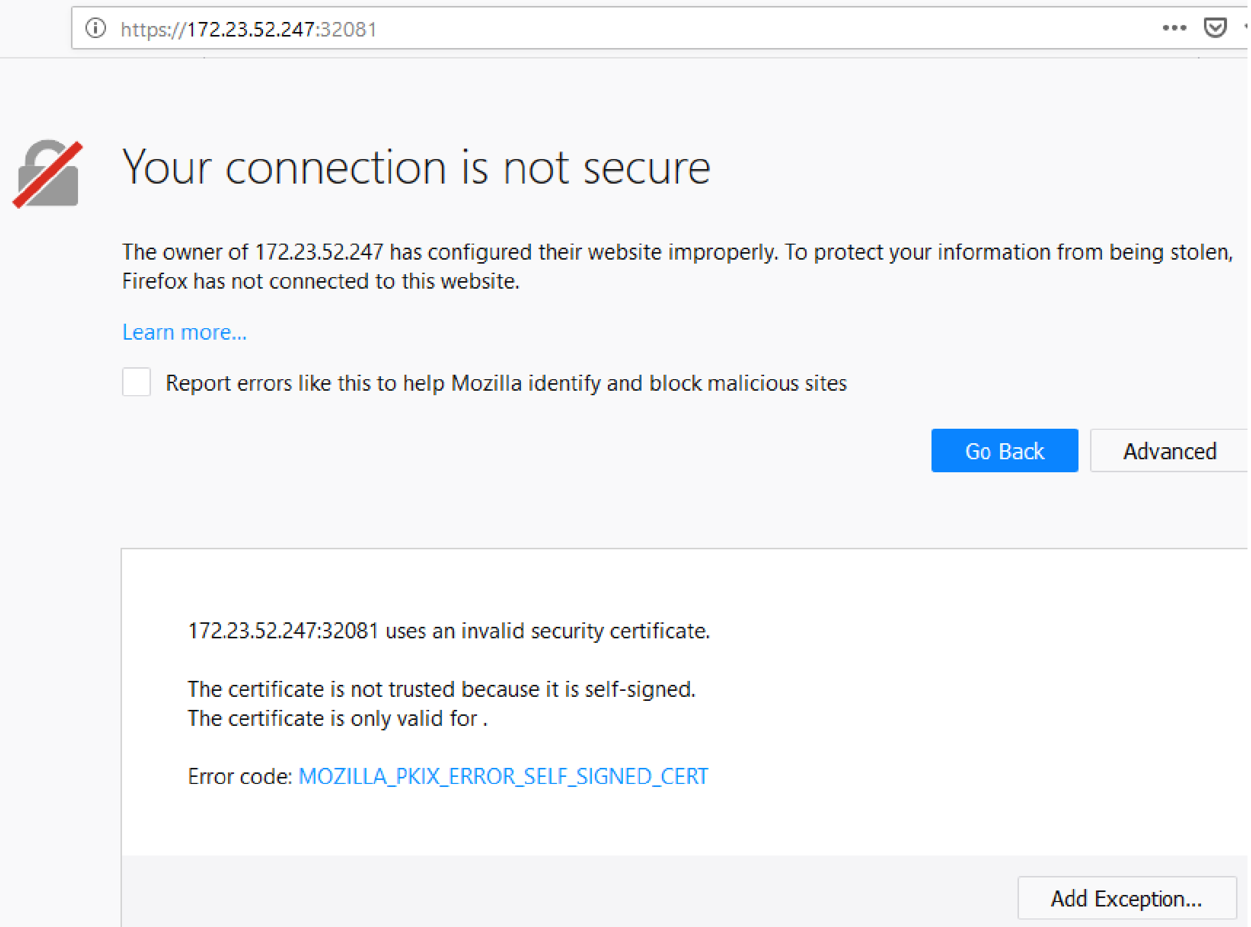
Add the HTTPS:// to the URL



https://172.23.52.247:32081

The connection was reset

Add the exception and confirm security



https://172.23.52.247:32081

Your connection is not secure

The owner of 172.23.52.247 has configured their website improperly. To protect your information from being stolen, Firefox has not connected to this website.

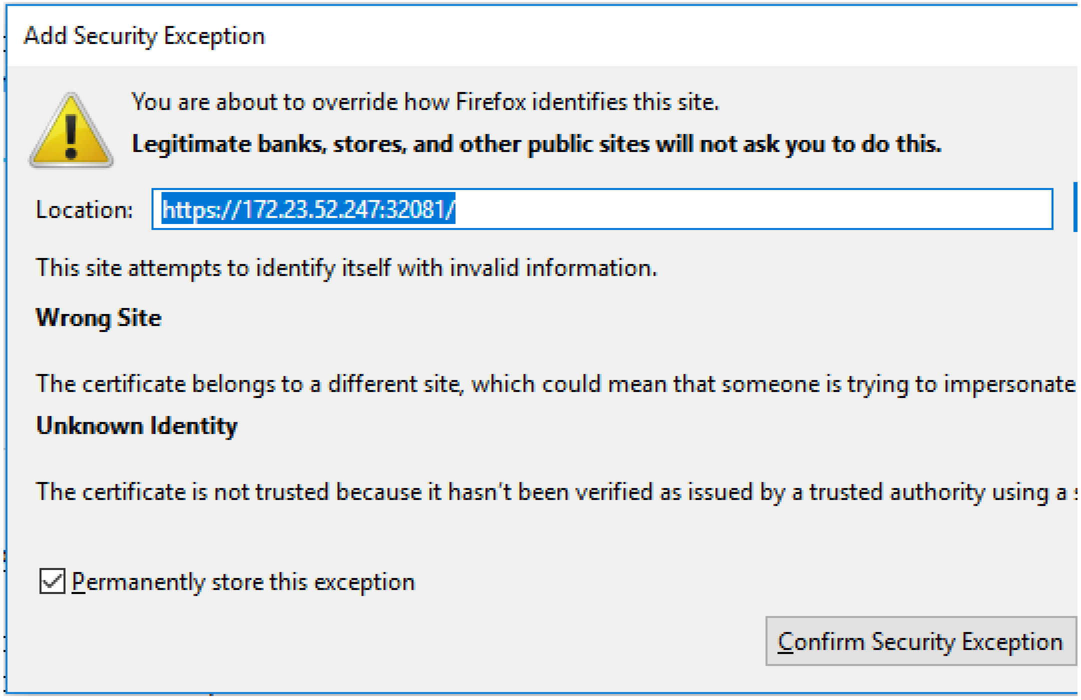
Learn more...

□ Report errors like this to help Mozilla identify and block malicious sites

172.23.52.247:32081 uses an invalid security certificate.

The certificate is not trusted because it is self-signed.
The certificate is only valid for .

Error code: MOZILLA_PKIX_ERROR_SELF_SIGNED_CERT



Add Security Exception

You are about to override how Firefox identifies this site.

Legitimate banks, stores, and other public sites will not ask you to do this.

Location: https://172.23.52.247:32081/

This site attempts to identify itself with invalid information.

Wrong Site

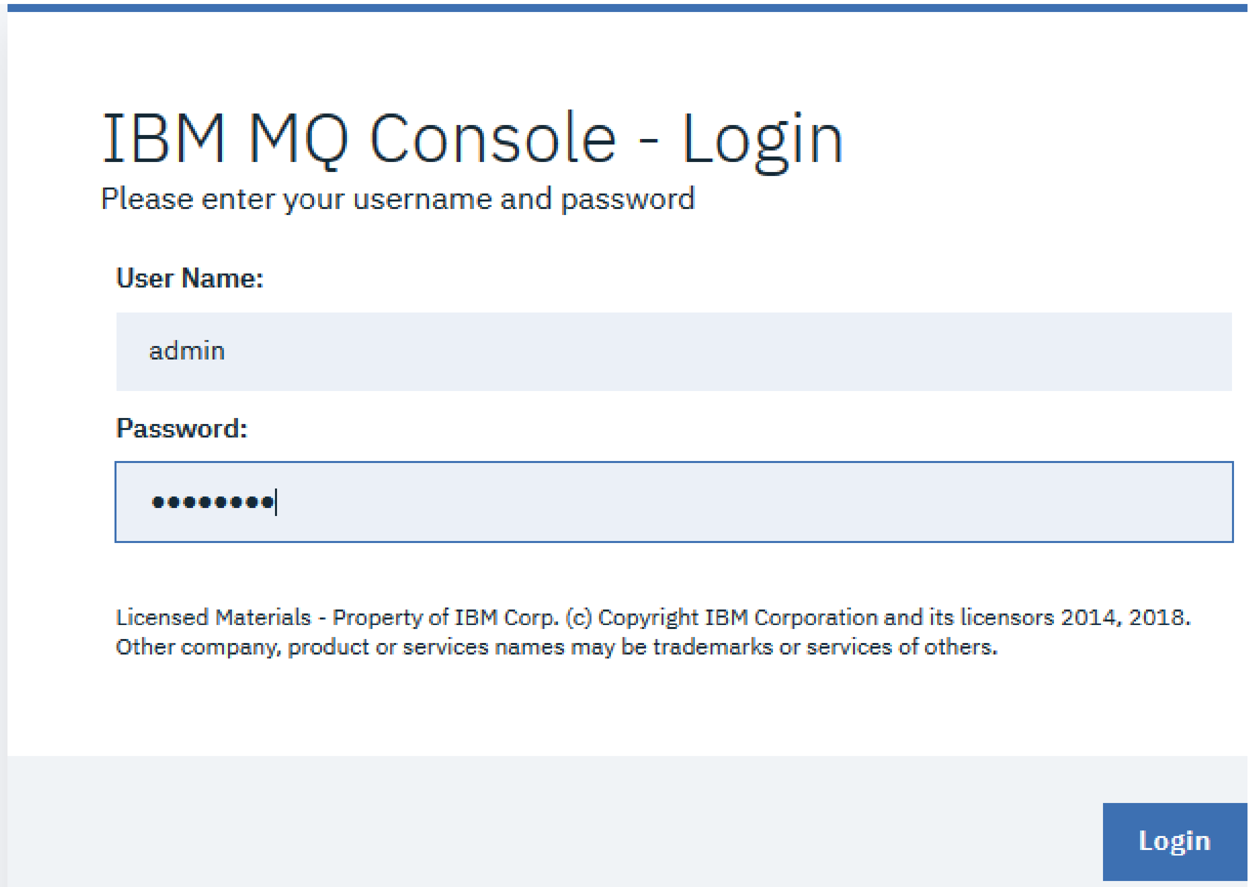
The certificate belongs to a different site, which could mean that someone is trying to impersonate

Unknown Identity

The certificate is not trusted because it hasn't been verified as issued by a trusted authority using a s

☑ Permanently store this exception

Login into the MQ console with admin/passw0rd



IBM MQ Console - Login

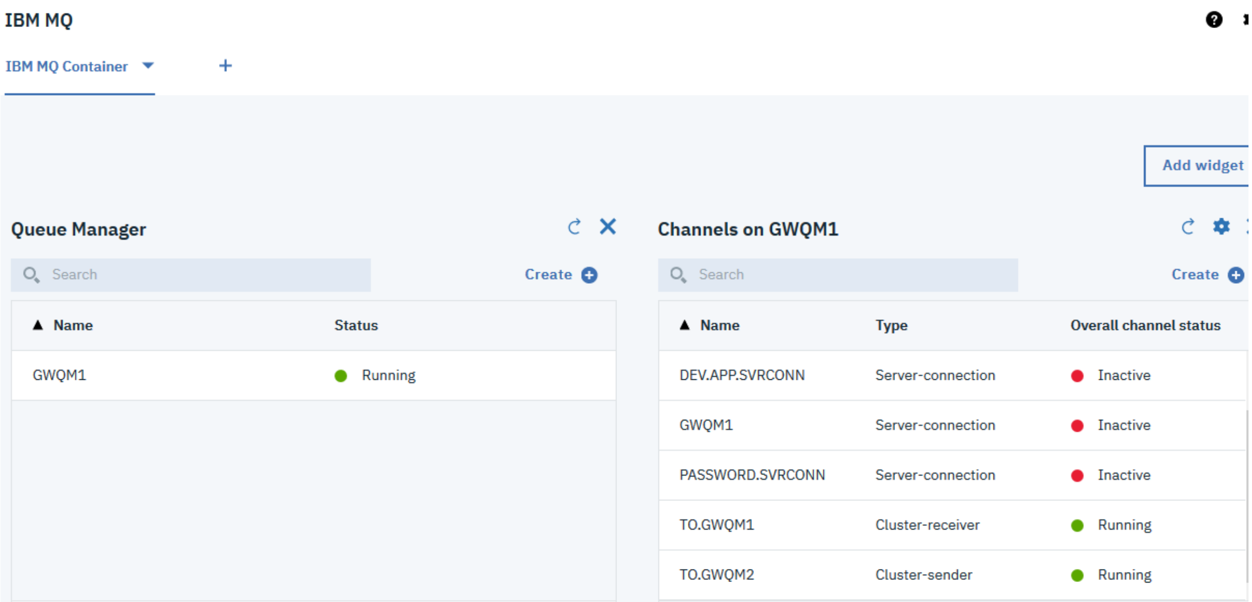
Please enter your username and password

User Name: admin

Password: ••••••••

Licensed Materials - Property of IBM Corp. (c) Copyright IBM Corporation and its licensors 2014, 2018. Other company, product or services names may be trademarks or services of others.

You'll see GWQM1 is running and there are connected Cluster Receiver and Sender channels



Queue Manager

▲ Name	Status
GWQM1	● Running

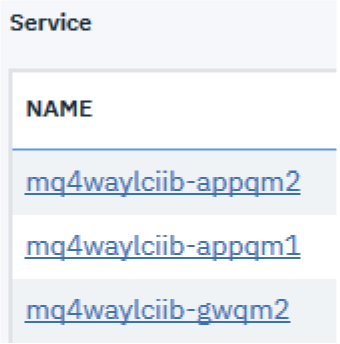
Channels on GWQM1

▲ Name	Type	Overall channel status
DEV.APP.SVRCONN	Server-connection	● Inactive
GWQM1	Server-connection	● Inactive
PASSWORD.SVRCONN	Server-connection	● Inactive
TO.GWQM1	Cluster-receiver	● Running
TO.GWQM2	Cluster-sender	● Running

Close the MQ console browser and tab and Repeat the process for GWQM2.

Testing GWQM2 MQ

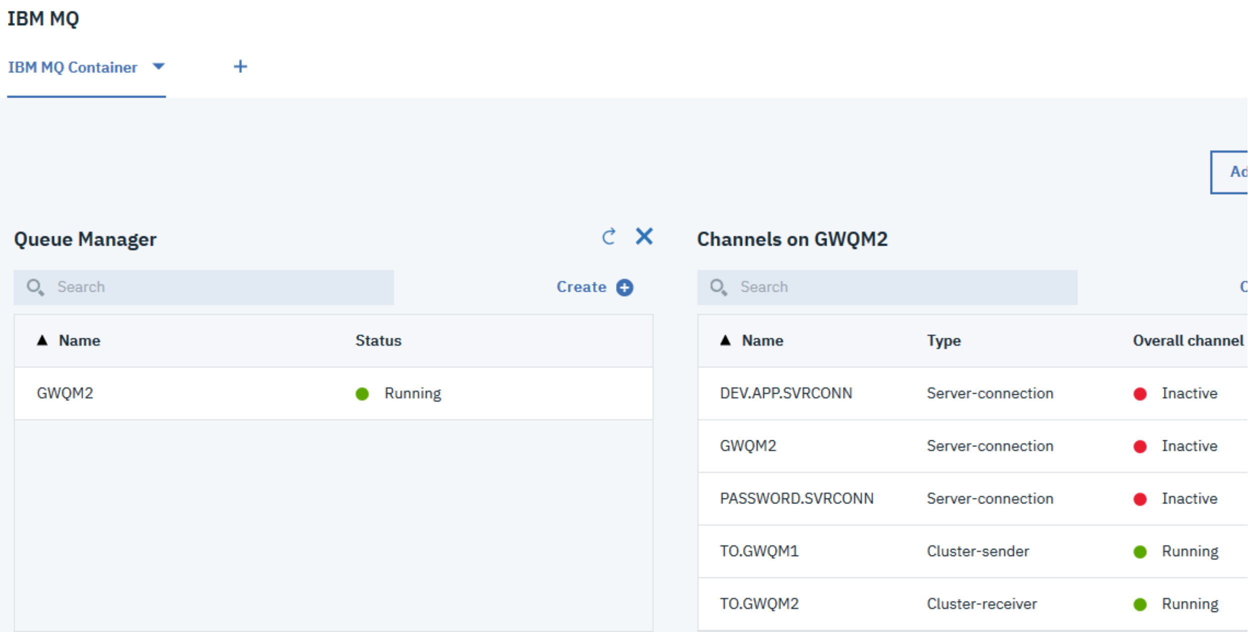
Select the GWQM2 service



Service

NAME
mq4waylciib-appqm2
mq4waylciib-appqm1
mq4waylciib-gwqm2

And repeat the process to bring up the MQ console



Queue Manager

Name	Status
GWQM2	● Running

Channels on GWQM2

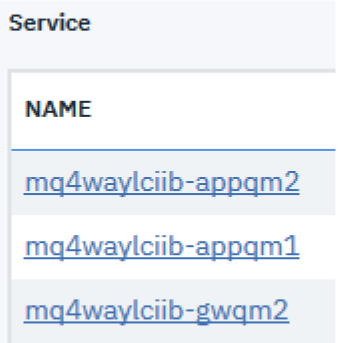
Name	Type	Overall channel
DEV.APP.SVRCONN	Server-connection	● Inactive
GWQM2	Server-connection	● Inactive
PASSWORD.SVRCONN	Server-connection	● Inactive
TO.GWQM1	Cluster-sender	● Running
TO.GWQM2	Cluster-receiver	● Running

Now lets check APPQM1 queue manager

Close the MQ console browser and tab and Repeat the process for APPQM1.

Testing APPQM1 MQ

Select the APPQM1 service

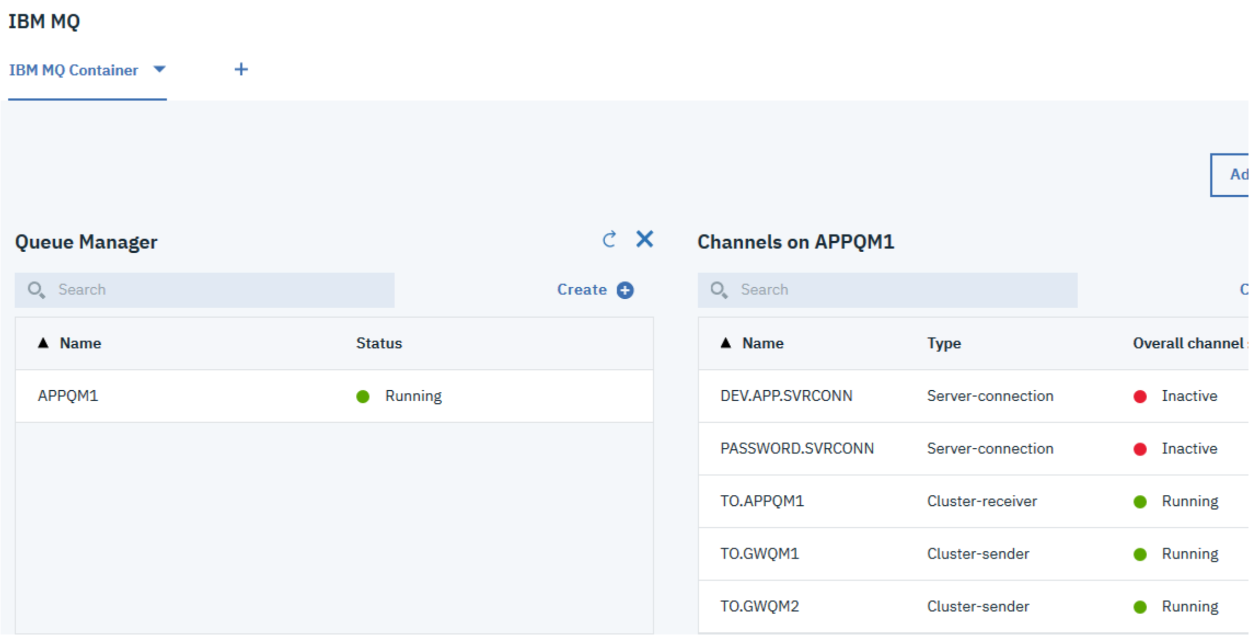


Service

NAME
mq4waylciib-appqm2
mq4waylciib-appqm1
mq4waylciib-gwqm2

And repeat the process to bring up the MQ console using the following link after selecting the mq4waylciib-appqm1 service.

mq4waylciib-console 30224/TCP



Queue Manager

Name	Status
APPQM1	● Running

Channels on APPQM1

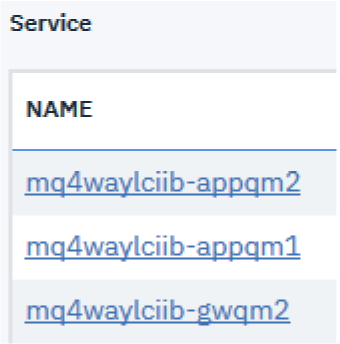
Name	Type	Overall channel
DEV.APP.SVRCONN	Server-connection	● Inactive
PASSWORD.SVRCONN	Server-connection	● Inactive
TO.APPQM1	Cluster-receiver	● Running
TO.GWQM1	Cluster-sender	● Running
TO.GWQM2	Cluster-sender	● Running

Now lets check APPQM2 queue manager

Close the MQ console browser and tab and Repeat the process for APPQM2.

Testing APPQM2 MQ

Select the APPQM2 service

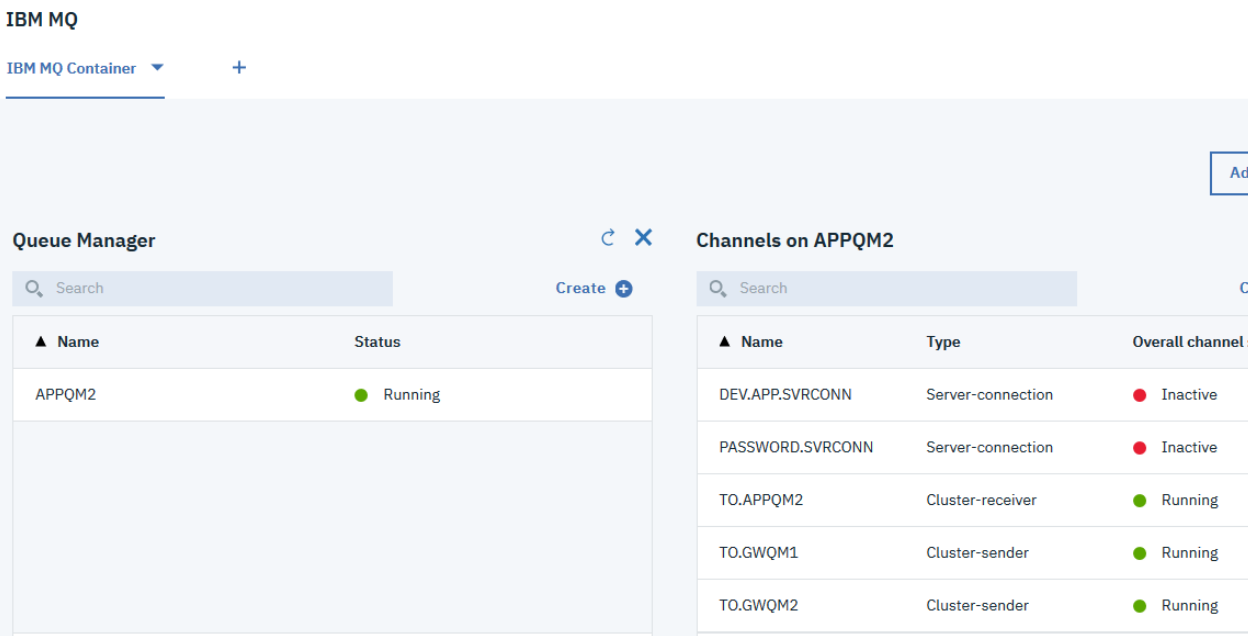


Service

NAME
mq4waylciib-appqm2
mq4waylciib-appqm1
mq4waylciib-gwqm2

And repeat the process to bring up the MQ console using the following link after selecting the mq4waylciib-appqm2 service.

mq4waylciib-console 31724/TCP



Queue Manager

Name	Status
APPQM2	● Running

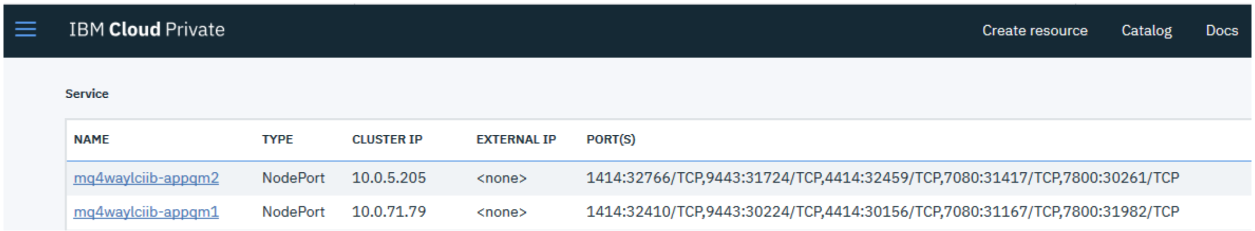
Channels on APPQM2

Name	Type	Overall channel
DEV.APP.SVRCONN	Server-connection	● Inactive
PASSWORD.SVRCONN	Server-connection	● Inactive
TO.APPQM2	Cluster-receiver	● Running
TO.GWQM1	Cluster-sender	● Running
TO.GWQM2	Cluster-sender	● Running

Now lets check the IIB nodes servicing APPQM1 and APPQM2

Testing APPNODE1 IIB

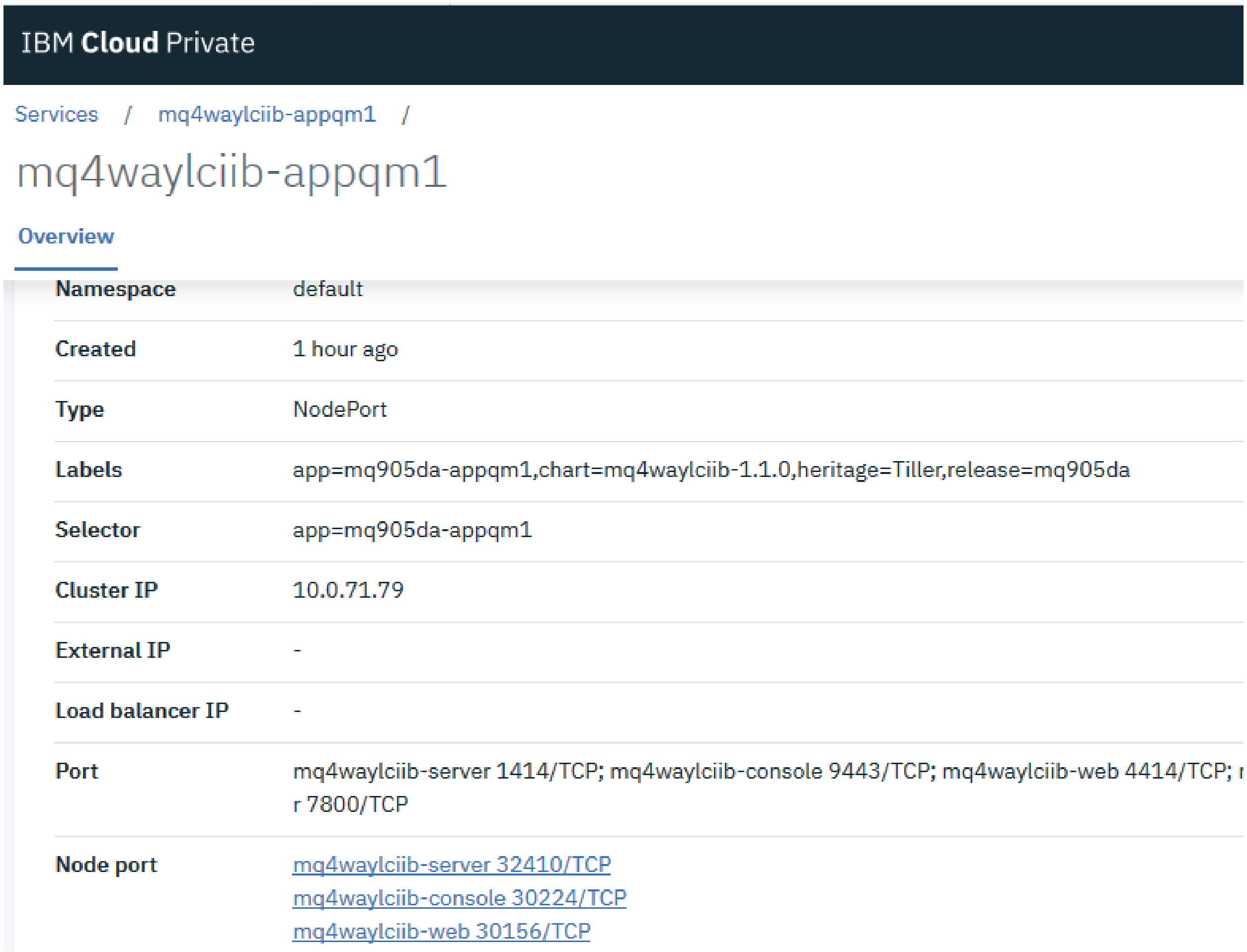
From the services in the MQ905da Helm Release



Service

NAME	TYPE	CLUSTER IP	EXTERNAL IP	PORT(S)
mq4waylciib-appqm2	NodePort	10.0.5.205	<none>	1414:32766/TCP,9443:31724/TCP,4414:32459/TCP,7080:31417/TCP,7800:30261/TCP
mq4waylciib-appqm1	NodePort	10.0.71.79	<none>	1414:32410/TCP,9443:30224/TCP,4414:30156/TCP,7080:31167/TCP,7800:31982/TCP

Select APPQM1 service



IBM **Cloud** Private

Services / mq4waylciib-appqm1 /

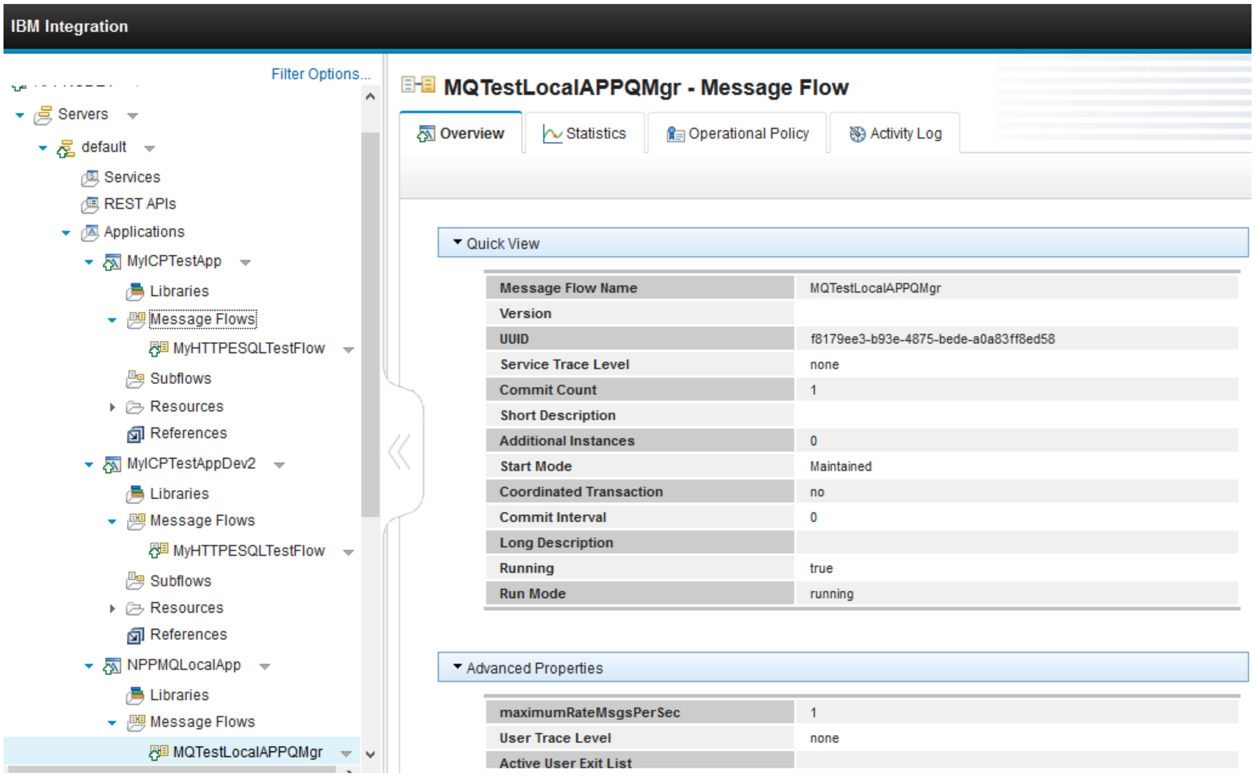
mq4waylciib-appqm1

Overview

Namespace	default
Created	1 hour ago
Type	NodePort
Labels	app=mq905da-appqm1,chart=mq4waylciib-1.1.0,heritage=Tiller,release=mq905da
Selector	app=mq905da-appqm1
Cluster IP	10.0.71.79
External IP	-
Load balancer IP	-
Port	mq4waylciib-server 1414/TCP; mq4waylciib-console 9443/TCP; mq4waylciib-web 4414/TCP; r r 7800/TCP
Node port	mq4waylciib-server 32410/TCP mq4waylciib-console 30224/TCP mq4waylciib-web 30156/TCP

mq4waylciib-web nnnnn/TCP

observe the IIB Node with applications and message flows deployed



MQTestLocalAPPQMgr - Message Flow

Quick View

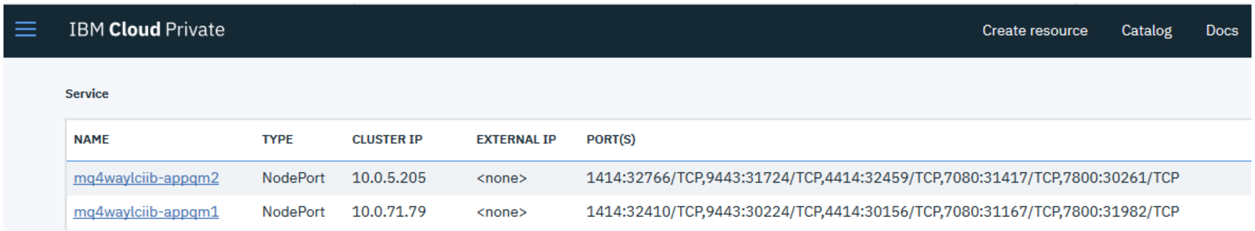
Message Flow Name	MQTestLocalAPPQMgr
Version	
UUID	f8179ee3-b93e-4875-bede-a0a83ff8ed58
Service Trace Level	none
Commit Count	1
Short Description	
Additional Instances	0
Start Mode	Maintained
Coordinated Transaction	no
Commit Interval	0
Long Description	
Running	true
Run Mode	running

Advanced Properties

maximumRateMsgsPerSec	1
User Trace Level	none
Active User Exit List	

Testing APPNODE2 IIB

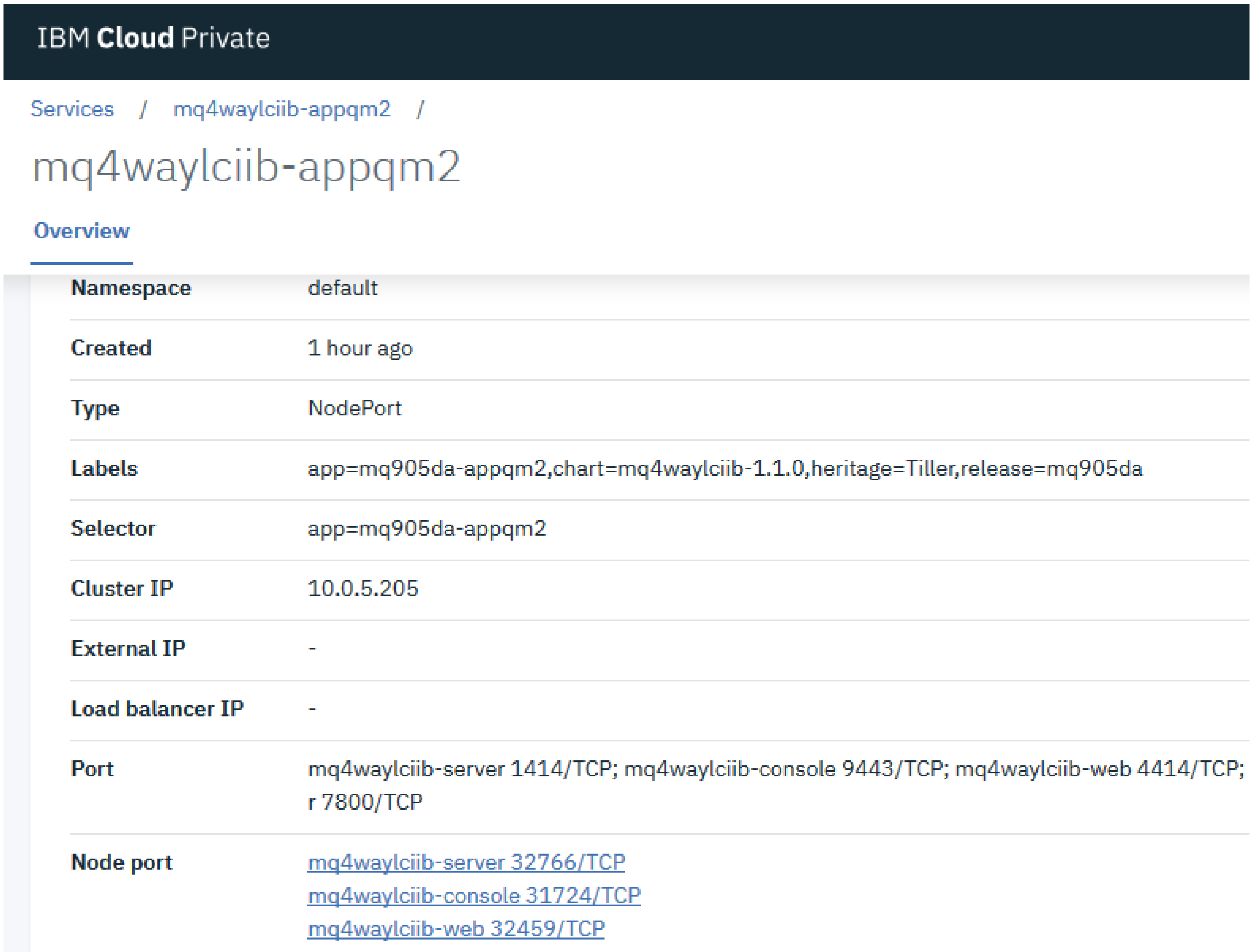
From the services in the MQ905da Helm Release



Service

NAME	TYPE	CLUSTER IP	EXTERNAL IP	PORT(S)
mq4waylciib-appqm2	NodePort	10.0.5.205	<none>	1414:32766/TCP,9443:31724/TCP,4414:32459/TCP,7080:31417/TCP,7800:30261/TCP
mq4waylciib-appqm1	NodePort	10.0.71.79	<none>	1414:32410/TCP,9443:30224/TCP,4414:30156/TCP,7080:31167/TCP,7800:31982/TCP

Select APPQM2 service


IBM **Cloud** Private

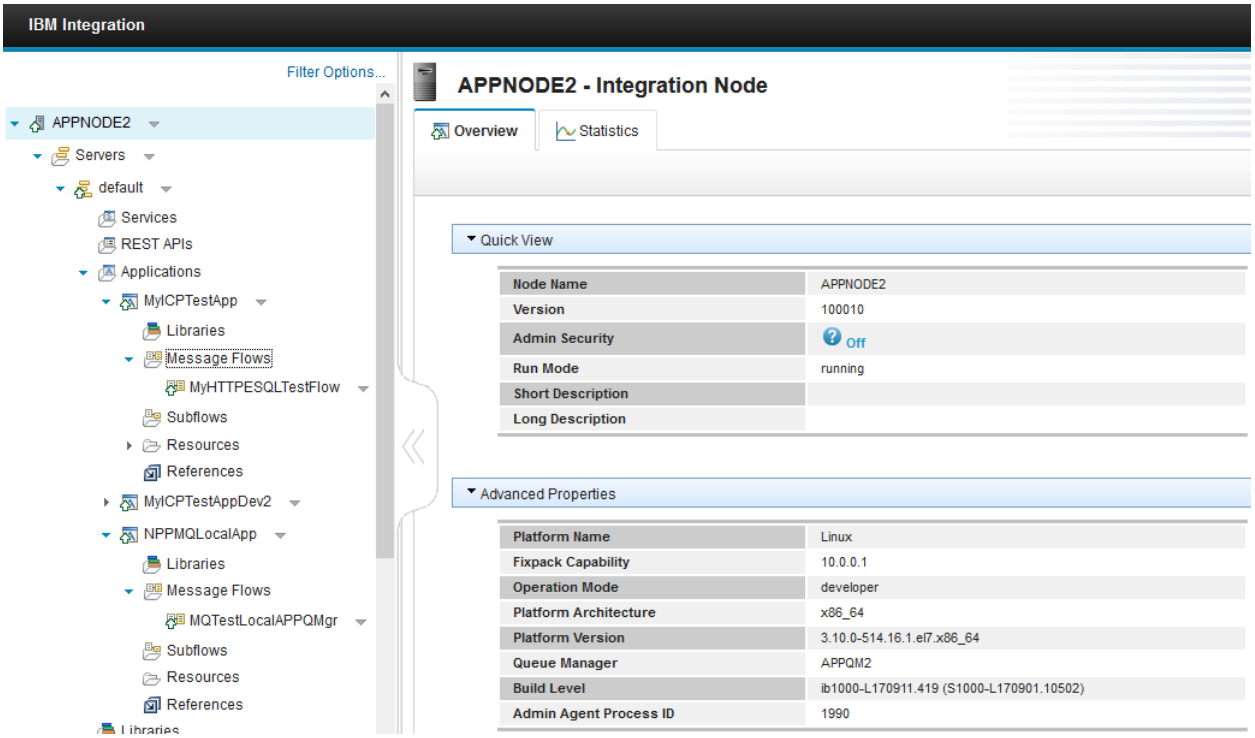
Services / mq4waylciib-appqm2 /

mq4waylciib-appqm2

Overview

Namespace	default
Created	1 hour ago
Type	NodePort
Labels	app=mq905da-appqm2,chart=mq4waylciib-1.1.0,heritage=Tiller,release=mq905da
Selector	app=mq905da-appqm2
Cluster IP	10.0.5.205
External IP	-
Load balancer IP	-
Port	mq4waylciib-server 1414/TCP; mq4waylciib-console 9443/TCP; mq4waylciib-web 4414/TCP; r 7800/TCP
Node port	mq4waylciib-server 32766/TCP mq4waylciib-console 31724/TCP mq4waylciib-web 32459/TCP

mq4waylciib-web nnnnn/TCP



APPNODE2 - Integration Node

Quick View

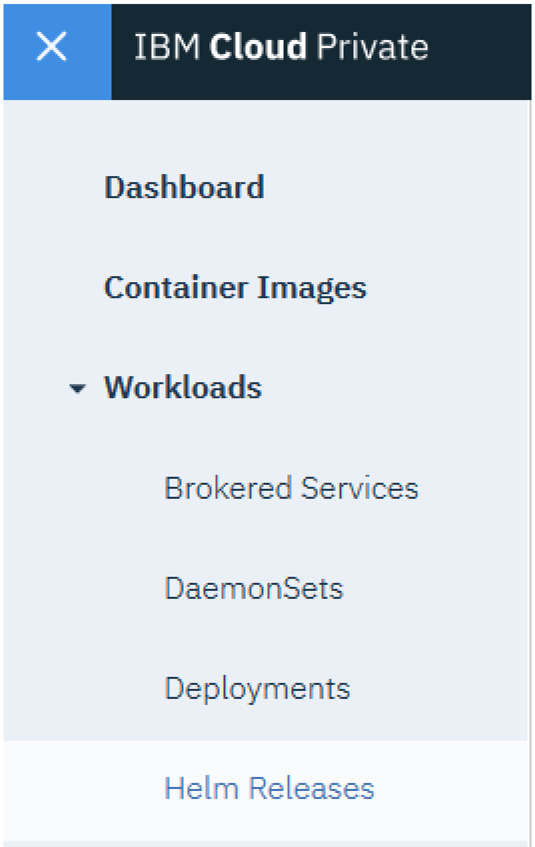
Node Name	APPNODE2
Version	100010
Admin Security	Off
Run Mode	running
Short Description	
Long Description	

Advanced Properties

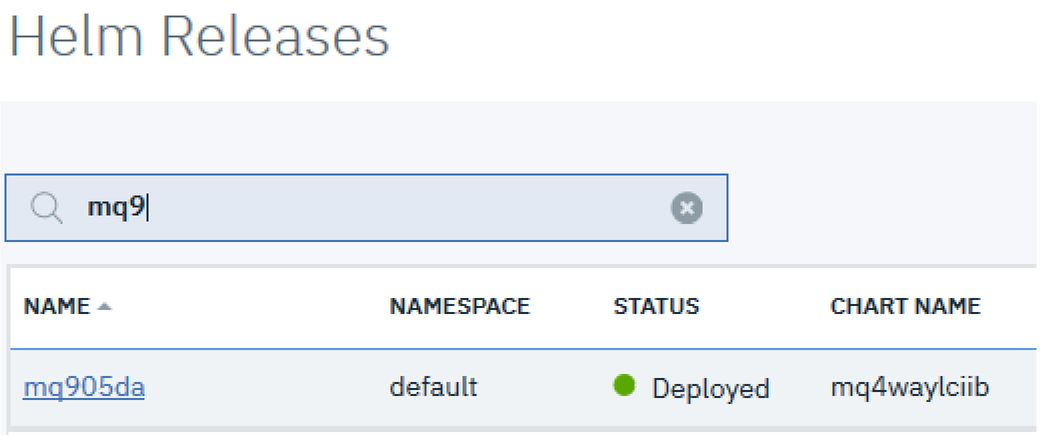
Platform Name	Linux
Fixpack Capability	10.0.0.1
Operation Mode	developer
Platform Architecture	x86_64
Platform Version	3.10.0-514.16.1.el7.x86_64
Queue Manager	APPQM2
Build Level	ib1000-L170911.419 (S1000-L170901.10502)
Admin Agent Process ID	1990

Removing the Helm Release and cleaning up

Select Workloads->Helm Release



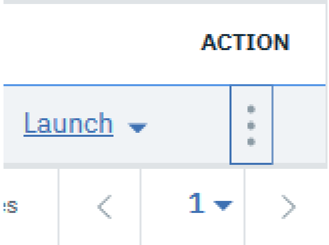
Filter on mq9



Helm Releases

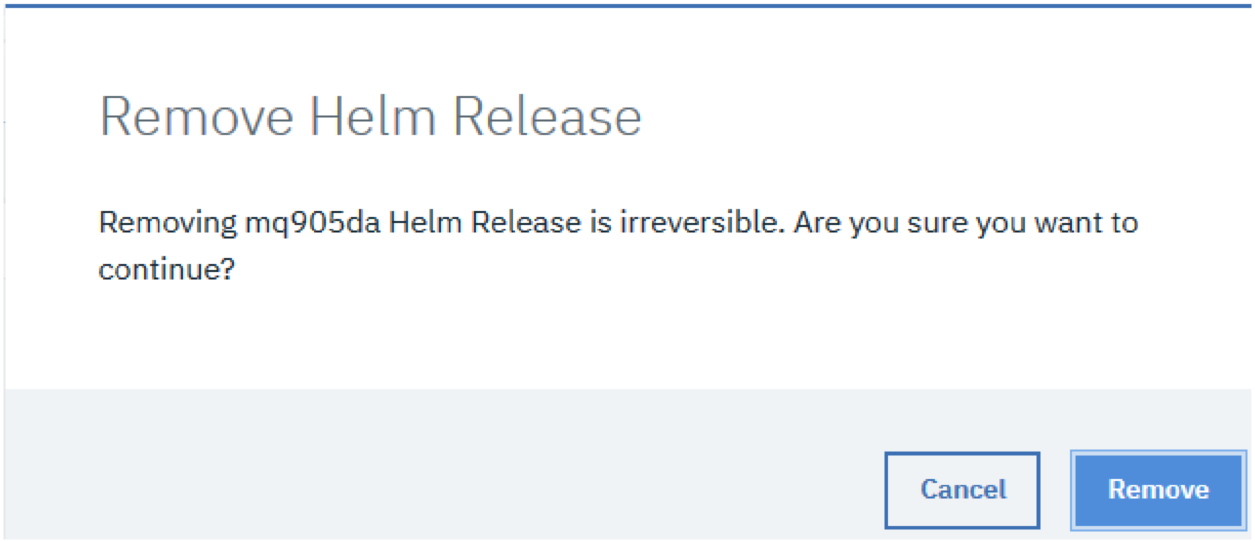
NAME	NAMESPACE	STATUS	CHART NAME
mq905da	default	● Deployed	mq4waylciib

On the mq905da release



Use the ACTION button (three vertical dots) to select DELETE

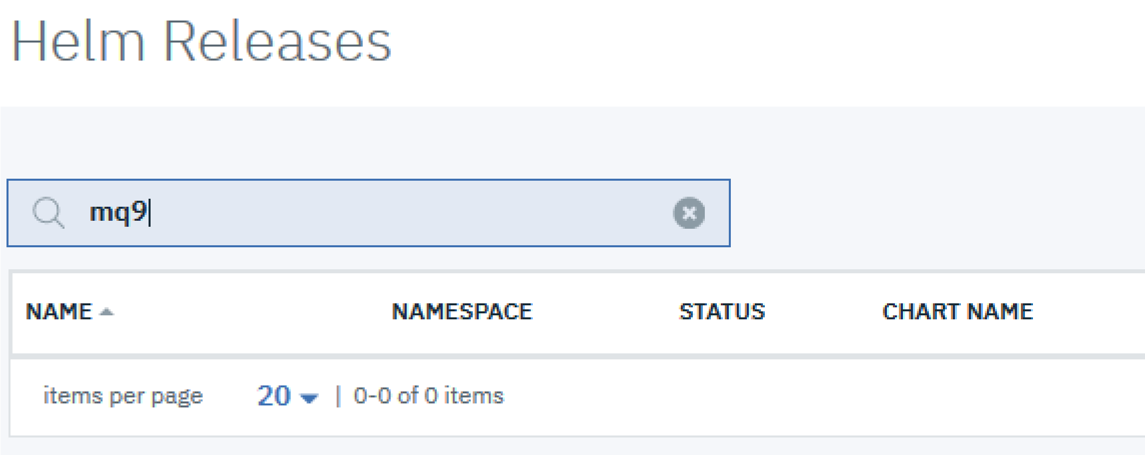
Select Remove at the dialog



Remove Helm Release

Removing mq905da Helm Release is irreversible. Are you sure you want to continue?

Refresh the browser and check the release has been removed



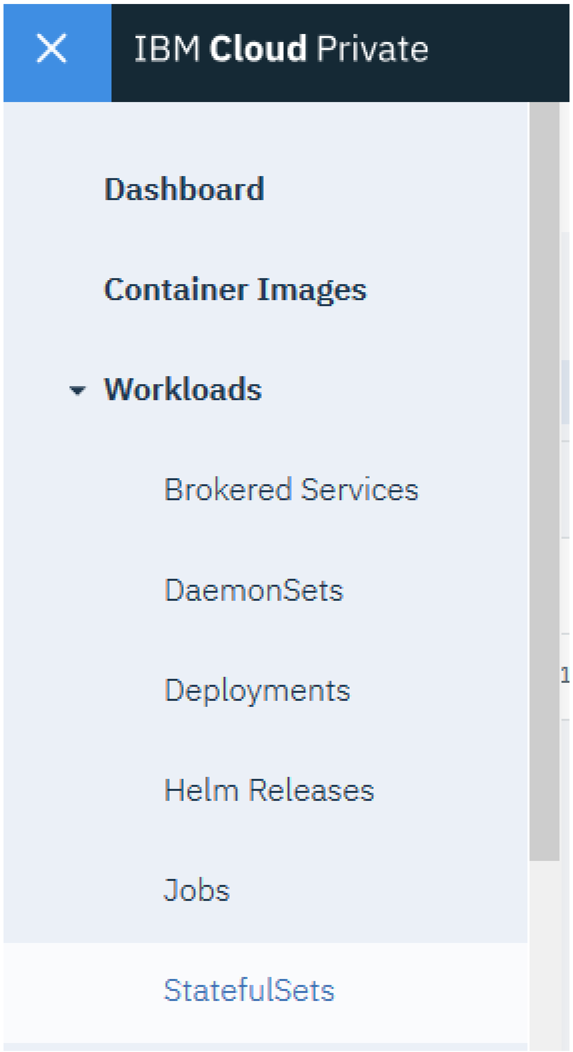
Helm Releases

mq9

NAME	NAMESPACE	STATUS	CHART NAME

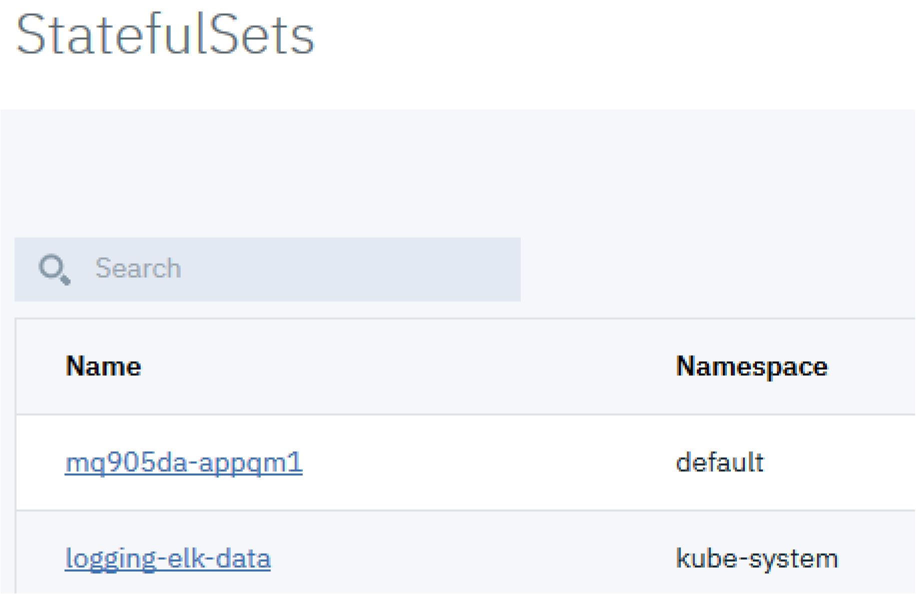
items per page 20 | 0-0 of 0 items

Check the stateful sets. Workloads->Stateful Sets



Some times it can take a couple of minutes for the stateful sets to be removed.

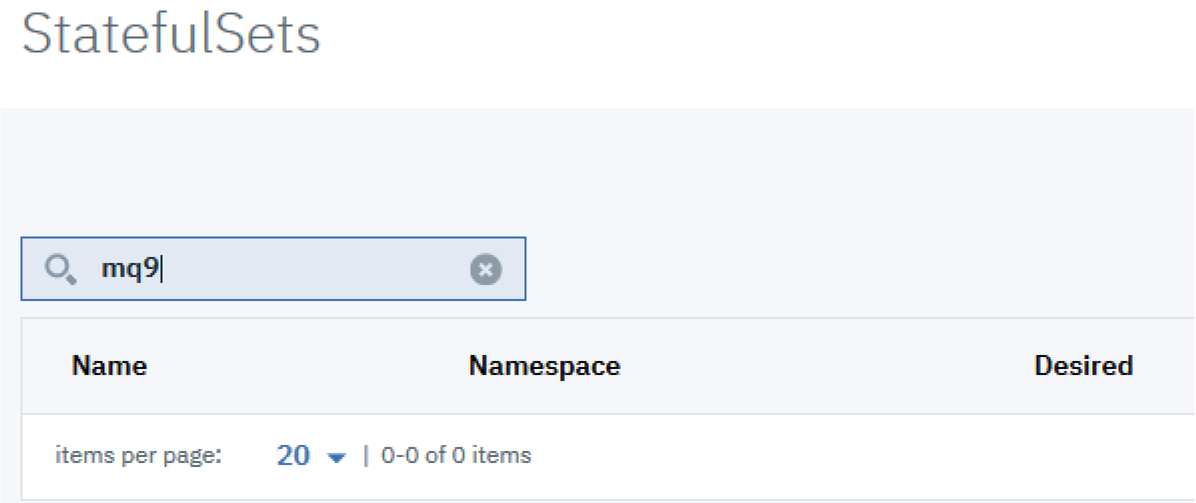
Filter on MQ9 and hit refresh a few times they should disappear.



StatefulSets

Name	Namespace
mq905da-appqm1	default
logging-elk-data	kube-system

You should see all the mq9 stateful sets disappear



StatefulSets

Search: mq9

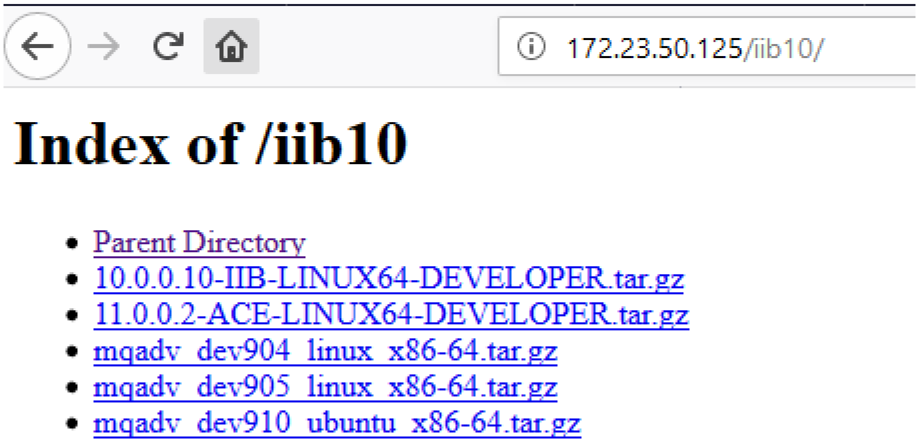
Name	Namespace	Desired

items per page: 20 | 0-0 of 0 items

Changing the version of MQ from 905 to 904 or vice versa for the GWQM1 and GWQM2 containers.

Updating the Github repository – MQ4WAYCLUS-LCIIB

I have copies of the MQ installation media on a localized drive rather than the public download URL (the public download site works fine but the builds are slower as a result).

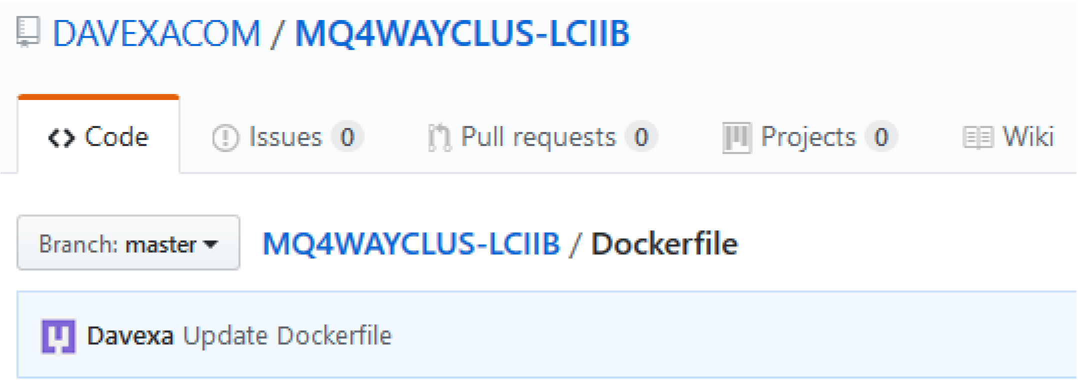


172.23.50.125/iib10/

Index of /iib10

- Parent Directory
- 10.0.0.10-IIB-LINUX64-DEVELOPER.tar.gz
- 11.0.0.2-ACE-LINUX64-DEVELOPER.tar.gz
- mqadv_dev904_linux_x86-64.tar.gz
- mqadv_dev905_linux_x86-64.tar.gz
- mqadv_dev910_ubuntu_x86-64.tar.gz

Edit the docker file to change the MQ installation version

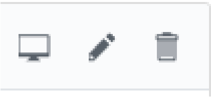


DAVEXACOM / MQ4WAYCLUS-LCIIB

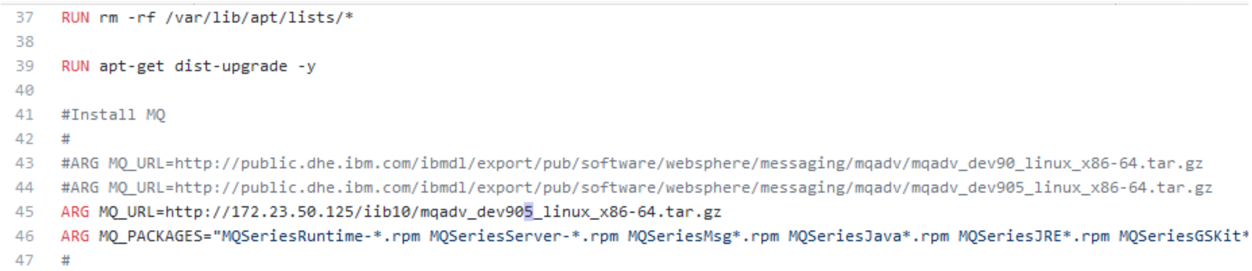
Branch: master ▾ MQ4WAYCLUS-LCIIB / Dockerfile

Davexa Update Dockerfile

Hit the Pencil Icon on the right



Scroll down and amend the ARG MQ_URL= line to point to different version of the MQ install media



```
RUN rm -rf /var/lib/apt/lists/*

RUN apt-get dist-upgrade -y

#Install MQ
#
#ARG MQ_URL=http://public.dhe.ibm.com/ibmdl/export/pub/software/websphere/messaging/mqadv/mqadv_dev90_linux_x86-64.tar.gz
#ARG MQ_URL=http://public.dhe.ibm.com/ibmdl/export/pub/software/websphere/messaging/mqadv/mqadv_dev905_linux_x86-64.tar.gz
ARG MQ_URL=http://172.23.50.125/iib10/mqadv_dev905_linux_x86-64.tar.gz
ARG MQ_PACKAGES="MQSeriesRuntime-*.rpm MQSeriesServer-*.rpm MQSeriesMsg*.rpm MQSeriesJava*.rpm MQSeriesJRE*.rpm MQSeriesGSKit*
#
```

Change the target MQ Installation media file,

ARG MQ_URL=http://172.23.50.125/iib10/mqadv_dev905_linux_x86-64.tar.gz

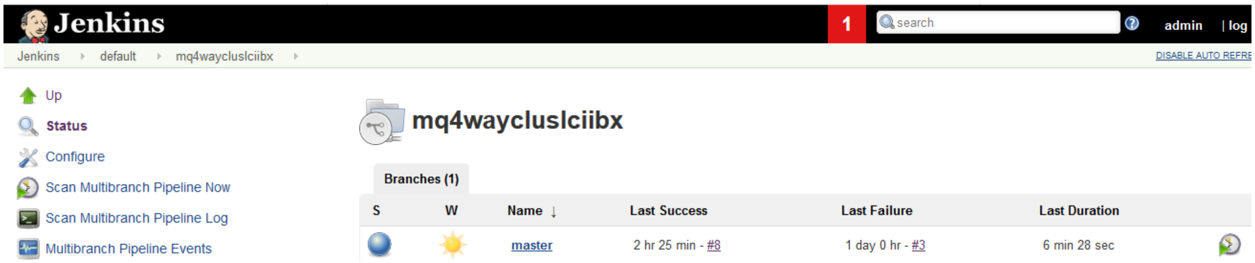
ARG MQ_URL=http://172.23.50.125/iib10/mqadv_dev904_linux_x86-64.tar.gz

And commit the change

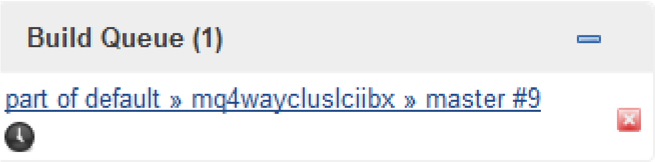
Perform a new build and Helm Release

Rerun the Jenkins pipeline

Goto you Jenkins console window and click the circle with the green triangle on the right

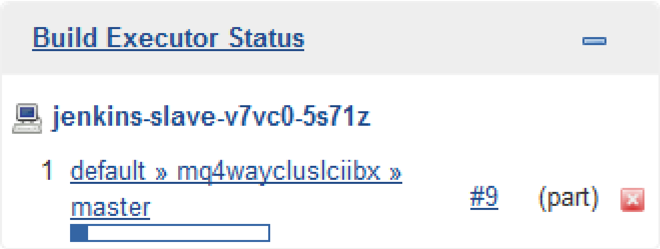


S	W	Name ↓	Last Success	Last Failure	Last Duration
●	☀	master	2 hr 25 min - #8	1 day 0 hr - #3	6 min 28 sec



part of default » mq4waycluslciibx » master #9

Wait for the slave to initiate the build

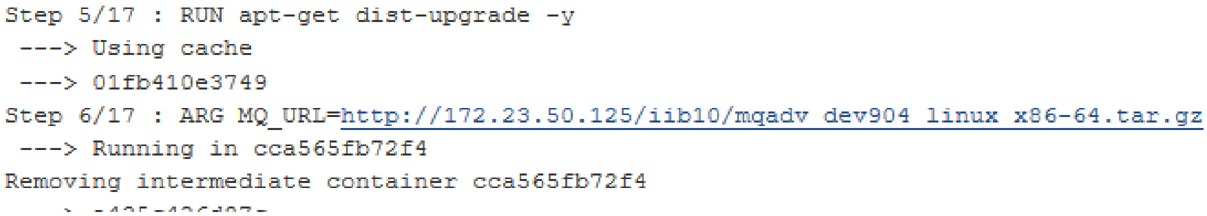


1 default » mq4waycluslciibx » master #9 (part)

Click on the #9 (in this example) and navigate to the console log.

select Console Output to review the build/deploy/release

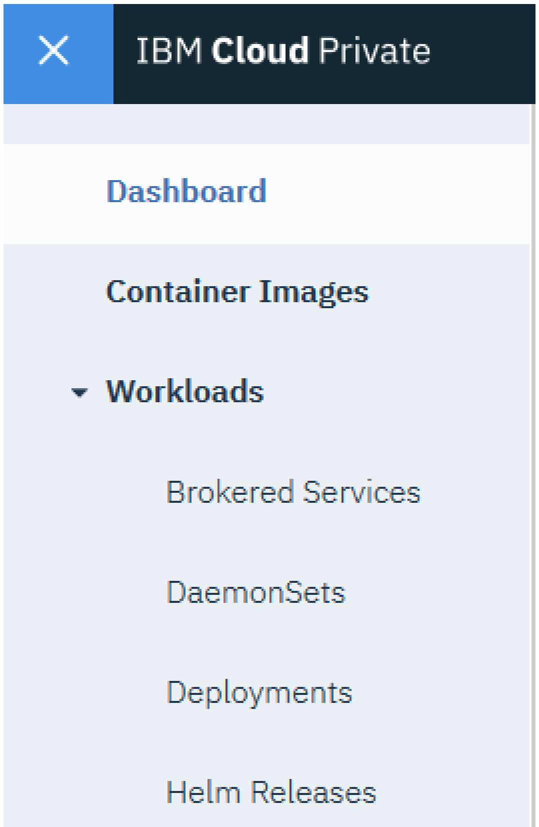
Note we are now building an image with MQ v904 in this example.



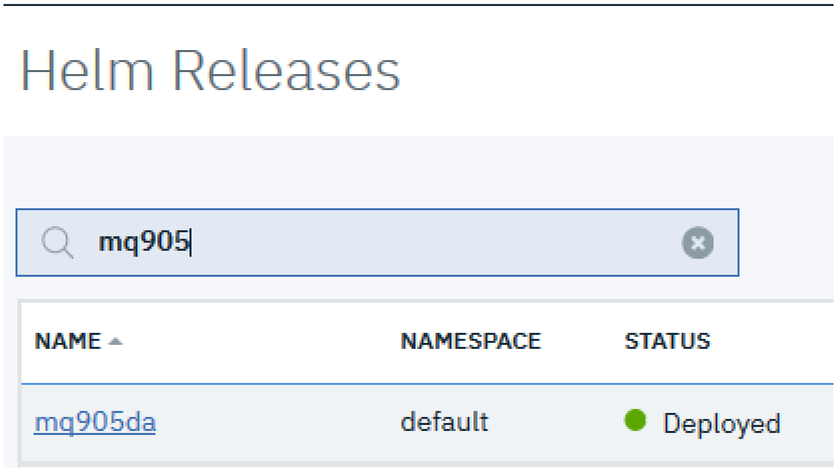
```
Step 5/17 : RUN apt-get dist-upgrade -y
 ---> Using cache
 ---> 01fb410e3749
Step 6/17 : ARG MQ_URL=http://172.23.50.125/iib10/mqadv_dev904_linux_x86-64.tar.gz
 ---> Running in cca565fb72f4
Removing intermediate container cca565fb72f4
```

Check the results of the Helm Release on ICP

Workloads->Helm Releases



Filter on mq905

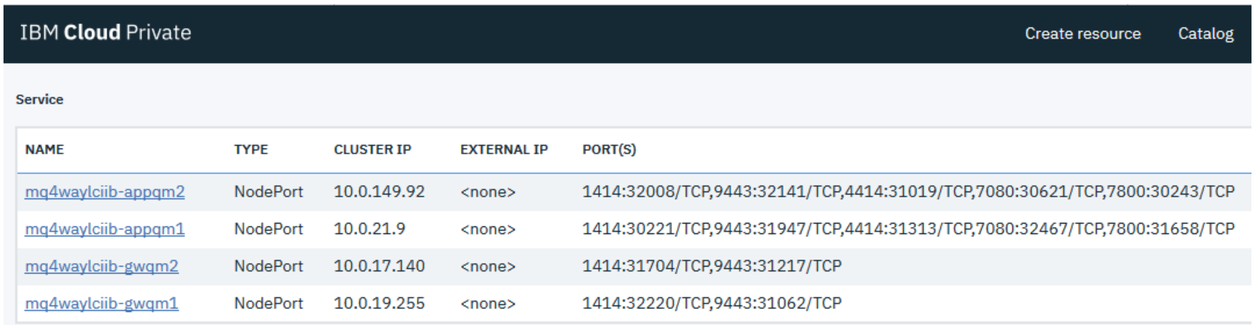


Helm Releases

mq905

NAME	NAMESPACE	STATUS
mq905da	default	● Deployed

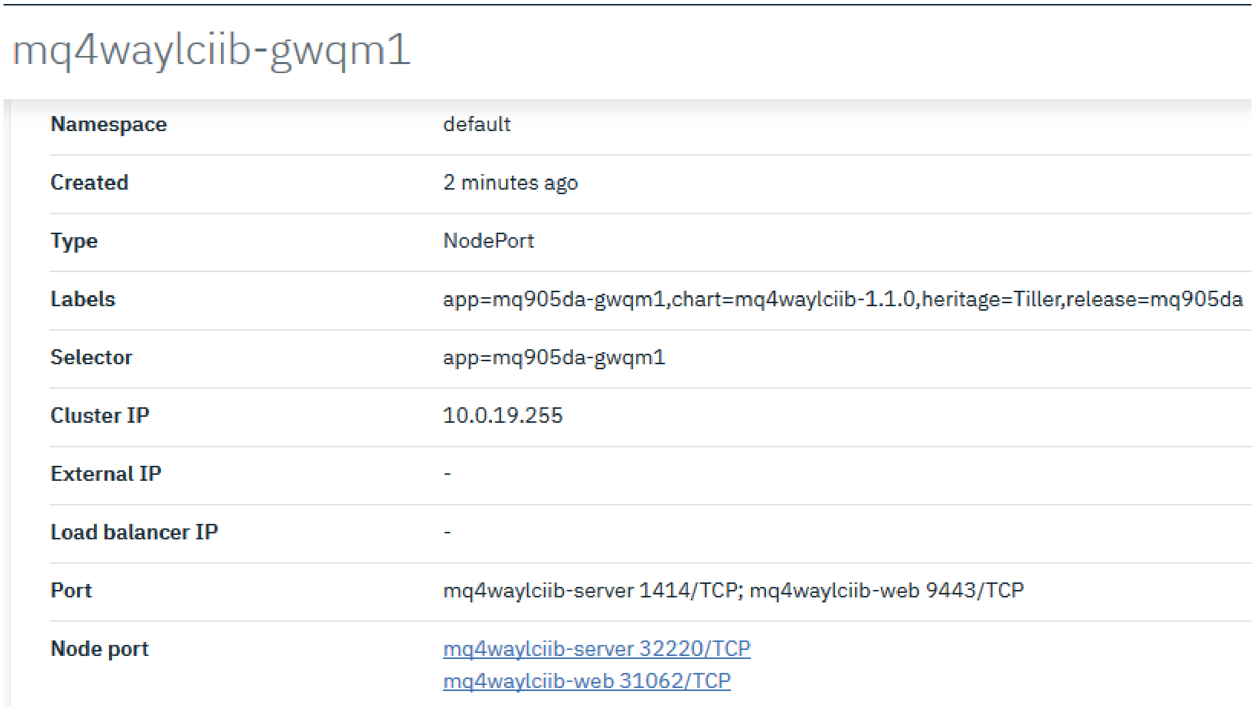
Click on mq905da and scroll down to the services



Service

NAME	TYPE	CLUSTER IP	EXTERNAL IP	PORT(S)
mq4waylciib-appqm2	NodePort	10.0.149.92	<none>	1414:32008/TCP,9443:32141/TCP,4414:31019/TCP,7080:30621/TCP,7800:30243/TCP
mq4waylciib-appqm1	NodePort	10.0.21.9	<none>	1414:30221/TCP,9443:31947/TCP,4414:31313/TCP,7080:32467/TCP,7800:31658/TCP
mq4waylciib-gwqm2	NodePort	10.0.17.140	<none>	1414:31704/TCP,9443:31217/TCP
mq4waylciib-gwqm1	NodePort	10.0.19.255	<none>	1414:32220/TCP,9443:31062/TCP

Click on mq4waylciib-gwqm1 link for GWQM1



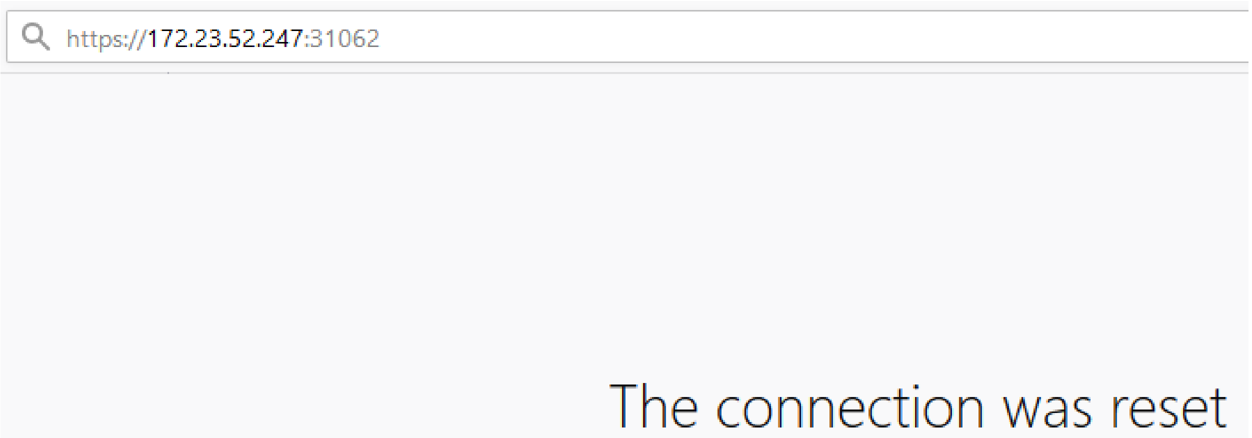
mq4waylciib-gwqm1

Namespace	default
Created	2 minutes ago
Type	NodePort
Labels	app=mq905da-gwqm1,chart=mq4waylciib-1.1.0,heritage=Tiller,release=mq905da
Selector	app=mq905da-gwqm1
Cluster IP	10.0.19.255
External IP	-
Load balancer IP	-
Port	mq4waylciib-server 1414/TCP; mq4waylciib-web 9443/TCP
Node port	mq4waylciib-server 32220/TCP mq4waylciib-web 31062/TCP

Validate the version of MQ on GWQM1

Select the mq4waylciib-web nnnnn/TCP link for the MQ console

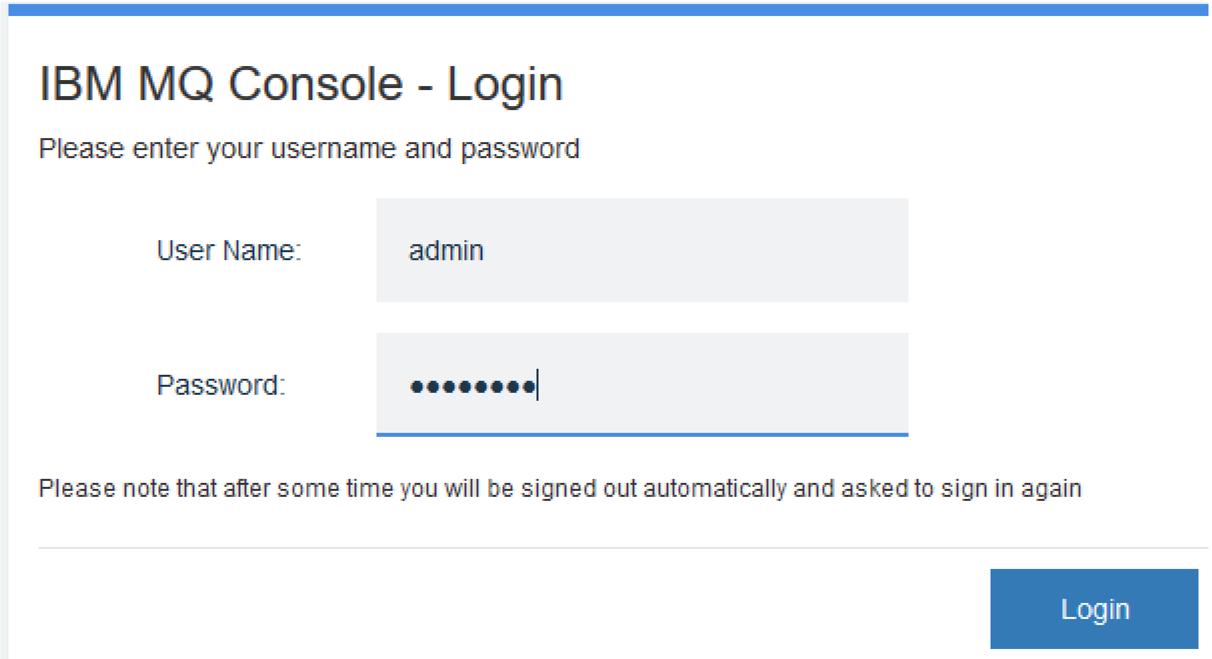
Add the HTTPS:// to the URL



https://172.23.52.247:31062

The connection was reset

Add and confirm the security exception in the browser



IBM MQ Console - Login

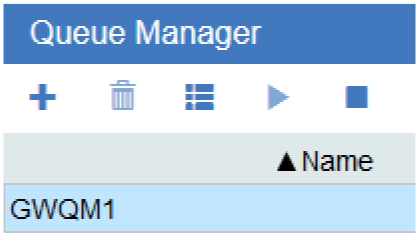
Please enter your username and password

User Name: admin

Password: ●●●●●●●●●

Please note that after some time you will be signed out automatically and asked to sign in again

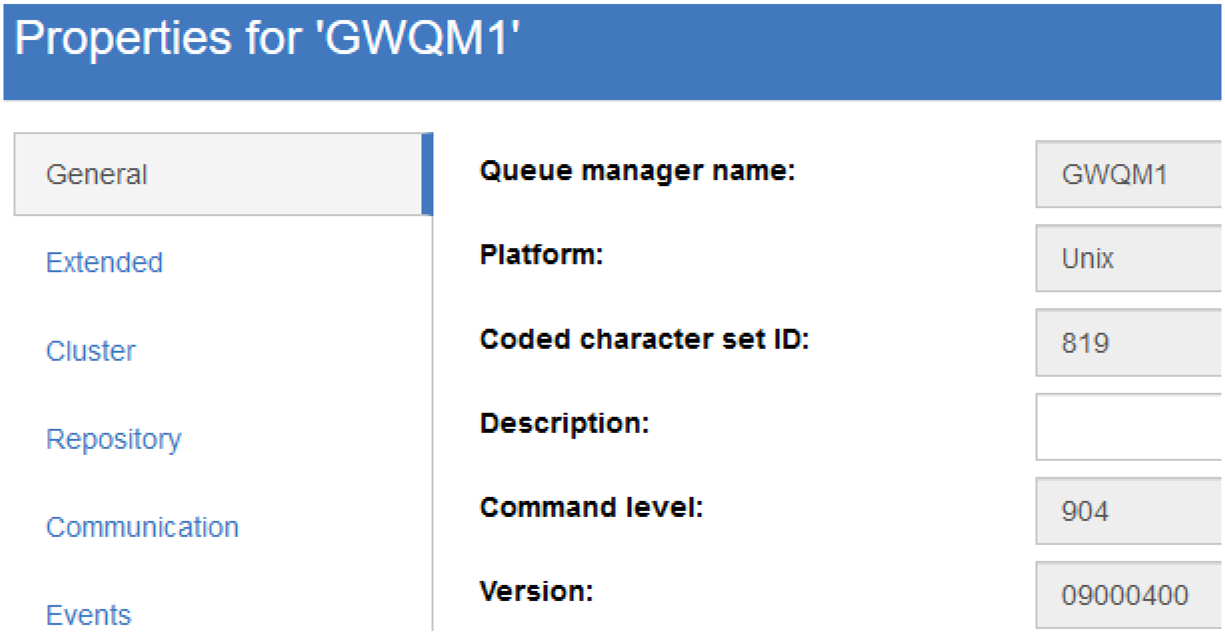
Login with admin/passw0rd



Queue Manager

▲Name
GWQM1

Select GWQM1 and Click on the properties button



Properties for 'GWQM1'

Queue manager name: GWQM1

Platform: Unix

Coded character set ID: 819

Description:

Command level: 904

Version: 09000400

Our Gateway Queue Managers has had its version changed.

The same will be true for GWQM2.