# HCL MaxAI Workbench Installation Guide

# Contents

# Chapter 1. Introduction to MaxAI Workbench

MaxAI Workbench is an AI life cycle management and data science platform.

A typical data analytics cycle has the following stages:

- Data Ingestion
- Data Pre-processing
- Feature Selection
- Model Training
- Visualization and Explanation
- Model Testing
- Model Conversion and Deployment
- Model Monitoring and Observation

In MaxAI Workbench, all these stages are automated. When you provide input data and the use case to be solved, MaxAI Workbench runs through all the stages of the data analytics cycle and generates the best model for deployment.

For more information, see *MaxAI Workbench User's Guide*.

# Chapter 2. MaxAI Workbench Installation

## Prerequisites for Cloud Native MaxAI Workbench

Before installation, ensure that the systems meet the recommended requirements.

### Minimum Hardware Requirements

The minimum hardware requirements for Cloud Native MaxAI Workbench are as follows:

| | |
|---|---|
| **Memory** | 32 GB RAM |
| **Storage** | 250 GB Hard Disk Drive |
| **CPU** | 16 CPU Machines |

### Software Requirements

The recommended software requirements for Cloud Native MaxAI Workbench are as follows:

| **Software Type** | **Recommended Requirement** |
|---|---|
| **Operating System** | CentOS 8<br><br>Red Hat Enterprise Linux 8<br><br>SuSE Linux Enterprise Server 12 Service Pack 5 (hosted Operating System) |
| **Base Operating System** | The base Operating System used for all images is Red Hat Universal Base Image 9 |
| **Database** | Oracle<br><br>PostgreSQL |
| **Other Software** | Docker Enterprise 20.10.17<br><br>Kubernetes 1.21, 1.22, and above<br><br>Helm 3<br><br>EKS, AKS, and OpenShift<br><br>**Ingress**: New Ingress controller repo https://kubernetes.github.io/ingress-nginx NGINX Ingress Controller 2.0 |

## Supported Cloud Platforms

Cloud Native MaxAI Workbench supports the following Cloud Platforms:

- Amazon Elastic Kubernetes Service (Amazon EKS)
- Google Cloud Platform (GCP)
- Microsoft® Azure
- RedHat OpenShift®

## Minimum Required Resources for Containers

The following table lists the minimum resources required for each container:

| Resource Name | Resource Value |
|---|---|
| **CPU** | 2000m (2000 millicpu per container) |
| **Memory** | 2048 MB |
| **Storage** | NFS is supported with MaxAI as persistent volume<br><br>Openshift - NFS<br><br>Azure - Azure Files<br><br>Google - File storage<br><br>If the containers fail and restart, the installation and the database data persist. However, setups are not packaged with the solution, and they need to be placed at the mount point. |

**Note:** The configuration will impact the minimum hardware requirements. It is recommended that you use a configuration that is slightly higher than the minimum requirements. If a container does not get the required resources, it might hang.

## Docker Image Security Vulnerabilities

For Operating System package vulnerabilities, like NSS, SQLite, etc., run the yum updates within the containers.

## Downloading the Cloud Native MaxAI Workbench Image

Perform the following steps to complete the registration process and configure the deployment:

| Topic | Section |
|---|---|
| Register | https://support.hcl-software.com/csm?id=kb_article&sysparm_article=KB0109011 |

| Topic | Section |
| --- | --- |
| | https://support.hcl-software.com/csm?id=kb_article&sysparm_article=KB0069114 |
| Create Deployment | https://help.hcl-software.com/myhclsoftware/topics/introduction.html |
| Generate Deployment key | |
| Map Entitlements | |
| Download Product Installers and Complete the Installation and Deployment process | |

## Deploying MaxAI Workbench - OpenShift

To install MaxAI Workbench on Red Hat OpenShift, complete the following steps:

**Before you begin**

The `uid` and `gid` for MaxAI Workbench to work on a Kubernetes environment is `5005`. For the Docker image creation to work, pull in `gcr.io/kaniko-project/executor` with the image registry provided to you.

1. On OpenShift, deploy the MaxAI Workbench chart as follows:
   `<max-ai-namespace>/<project-name>`
2. On your local environment, setup the OpenShift command line interface.
3. Run the command `oc Project <project-name>` where *<project-name>* is the name of the project. For example, `max-ai`.
4. Apply the `pvc.yaml` part of the helm chart package.
5. Use one of the following deployment commands:
   - To install Helm charts, run the command `helm install aion ./aion-marketing -n "namespace name"`
   - To install or upgrade Helm charts, run the command `helm upgrade -i aion . -n "namepsace/project name"`
   - To upgrade Helm charts, run the command `helm upgrade -i aion . -n "namepsace/project name"`

## Configuring MaxAI Workbench after Installation

After installation or deployment of MaxAI Workbench in Red Hat OpenShift, complete the following configurations:

1. Extract the `prod_setup.zip` file to the required location.
2. Access the location of extraction and open the `prod_setup` folder.
3. Open the `aion_config.ini` file.
4. Within the **[path]** section of the parameters, if required, change the value of the **aion_base_mp_loc** parameter. The default value is `/home/aionuser/`. If you do not customize the value of the **aion_base_mp_loc** parameter, the default value is applied.
   The value set for **aion_base_mp_loc**, impacts all the following parameters:
   - **data_dir**
   - **deployment_location**
   - **target_location**

- **eda_location**
- **config_location**
- **log_location**
- **configtemplate_location**
-

5. Within the **[path]** section of the parameters, for the **CSRF_TRUSTED_ORIGINs** parameter, provide atleast one URL (HTTP and HTTPS) in the following format:

```
http://<domain-name>.com,https://<domain-name>.com
```

**Example**:

```
http://acme.company.aion.com,https://acme.company.aion.com
```

If you are maintaining two different UIs (for example, normal and Swagger) in two different servers, add the Swagger URL (HTTP and HTTPS) as well.

**Example**:

```
http://acme.company.aion.com,https://acme.company.aion.com,http://acme.company.swagger.aion.com,https://
acme.company.swagger.aion.com
```

6. In the `values.yaml` file, update the following environment variables as per your environment:

**Note:** The values provided here are example values. Change it as per your environment and settings.

```
serviceAccount:
   name: "aion-sa"
deploymentConfig:
    PV_CLAIM: 'aion-pvc-claim'
    MOUNT_POINT: '/mnt/aion_share'
    AION_SERVICE_ACCOUNT: 'aion-sa'
    ENV_Namespace: 'default'
    ENV_NFS_Share: '/mnt/aion_share'
    AION_BASE_MP_LOC: '/mnt/aion_share'
    ALLOWED_HOSTS: 'swagger2-marketing-aion.apps.com'
    CSRF_TRUSTED_ORIGINS: 'https://unicatango.cloud,http://localhost,https://localhost'
    CORS_ORIGINS_WHITELIST: 'https://unicatango.cloud,http://localhost,https://localhost'
    CORS_ALLOWED_ORIGINS: 'https://unicatango.cloud,http://localhost,https://localhost'
    unicaPlatformUrl: 'https://unicatango.cloud/unica' #'https://unicatango.hcltechsw.com/unica'
    platformurlinternal: 'https://unicatango.cloud/unica' #'https://unicatango.hcltechsw.com/unica'
    AION_AUTH: 'True'
    docker_package_image: 'gcr.io/kaniko-project/executor'
    docker_package_tag: 'latest'

 pvcspecs:
    pvcname: aion-pvc-claim
    storageclassname: manual

  volumes:
    - name: aion-pvc-claim
      persistentVolumeClaim:
        claimName: aion-pvc-claim
```

```
volumeMounts:
  - name: aion-pvc-claim
    mountPath: /mnt/aion_share/
```

For SSO, MaxAI Workbench uses `UNICA`. Based on your environment, update the following keys in the `values.yaml` file:

```
unicaPlatformUrl: 'https://platform-unica.apps.sysNNN.hostname.production.company.com/unica'
    platformUrlInternal: 'https://platform-unica.apps.sysNNN.hostname.production.company.com/unica'
```

If the environment `context_root` is "`/`", the login path will be `/platform/login`. If the environment `context_root` is `/`*aion* the login path will be `/aion/platform/login`.

7. As a part of the helm charts package, `ingress.yaml` and `route.yaml` are configured to access the application.

# Chapter 3. MaxAI Workbench Integration Single Sign-on

Long running trainings in MaxAI Workbench can time out if Unica Platform's session times out before the training is over in MaxAI Workbench. For on-premises Platform installation, manually run the `configTool` and `populateDb` commands to get the required MaxAI Workbench configuration node and **AIONAdmin** role in Platform.

## Configurations for Microsoft Windows

1. Open the command prompt.
2. Access the `Platform/tools/bin` directory.
3. Execute the following commands:

**Note:** Replace <*PLATFORM_HOME*> with the absolute path to Unica Platform installation home.

- ```
  ./populateDb.bat -n Manager
  ```
- ```
  ./populateDb.bat -n AION
  ```
- ```
  ./configTool.bat -vp -p "Affinium" -f
   "<PLATFORM_HOME>/Platform/conf/upgrade1218to1219/AION_configuration.xml"
  ```
- ```
  ./configTool.bat -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f
   "<PLATFORM_HOME>/Platform/conf/upgrade1218to1219/AION_Menu.xml"
  ```

## Configurations for UNIX-based Operating Systems

1. Open the command line interface.
2. Execute the following commands:

**Note:** Replace <*PLATFORM_HOME*> with the absolute path to Unica Platform installation home.

- ```
  cd Platform/tools/bin
  ```
- ```
  ./populateDb.sh -n Manager
  ```
- ```
  ./populateDb.sh -n AION
  ```
- ```
  ./configTool.sh -vp -p "Affinium" -f
   "<PLATFORM_HOME>/Platform/conf/upgrade1218to1219/AION_configuration.xml"
  ```
- ```
  ./configTool.sh -vp -p "Affinium|suite|uiNavigation|mainMenu|Analytics" -f
   "<PLATFORM_HOME>/Platform/conf/upgrade1218to1219/AION_Menu.xml"
  ```
  ```

# Chapter 4. Configuration on Unica Platform

After installing MaxAI Workbench, you must assign the MaxAI Workbench-specific role to the user to access MaxAI Workbench from Unica.

**About this task**

To assign the role, complete the following steps:

1. Log in to Unica.
2. Navigate to **Settings > Users**
3. Select the required username from the left panel.
4. On the right panel, select **Edit roles**.
5. Select the `AIONAdmin (Unica AION)` from the **Available roles** list and click the **>>** button to move the role to the **Selected roles** list.

   **Result**

   This assigns the MaxAI Workbench-specific role to the selected user.
6. Click **Save changes**.

   **Result**

   This activates the MaxAI Workbench menu on Unica.

# Chapter 5. MaxAI Workbench Uninstallation

To uninstall MaxAI Workbench deployment on OpenShift, run the following command:

```
helm uninstall aion . -n "namespace/project name"
```