

IBM Maximo Equipment Maintenance  
Assistant V1.1.1

*User Guide*





# Note

---

Before using this information and the product it supports, read the information in [“Notices”](#) on page 55.



---

April 2020



---

# Contents

- Chapter 1. Product overview and features..... 1**
  - How it works..... 1
  - What's new in Maximo Equipment Maintenance Assistant V 1.1.1..... 2
  - Integration with Enterprise Asset Management solutions..... 3
  - Accessibility features..... 3
  
- Chapter 2. Installing the solution..... 5**
  - System requirements..... 5
  - Preparing for the installation..... 7
  - Installing and configuring Helm charts..... 9
    - Configuring dependencies..... 9
    - Installing Helm charts manually..... 11
    - Installing Helm charts with deploy.sh on Linux..... 12
    - Verifying the Helm chart installation..... 13
    - Configuring Helm charts..... 13
    - Helm chart configurable parameters..... 14
    - Managing Helm releases..... 17
    - Uninstalling the solution..... 18
  - Installing prerequisites..... 18
    - Installing CouchDB on Red Hat OpenShift..... 18**
    - Installing Watson Discovery 2.1.2 for IBM Cloud Pak for Data..... 19
    - Installing S3 Compatible Object Store on Red Hat OpenShift..... 19
  
- Chapter 3. Administering the solution..... 21**
  - Generating API keys..... 21
  - Adding and managing users..... 21
  - Configuring SAML users..... 22
  - Monitoring and logging..... 23
  
- Chapter 4. Backing up and restoring data..... 25**
  - Backing up and restoring CouchDB data..... 25
  - Backing up and restoring Watson Discovery data..... 26
  - Backing up and restoring OpenShift GlusterFS or Ceph Object Storage data..... 26
  
- Chapter 5. Building models..... 29**
  - Manage documents in Watson Discovery..... 29
  
- Chapter 6. Using the product..... 31**
  - Initializing your environment..... 31
    - Activating your administrator account..... 31
  - Managing documents in Document Manager..... 31
    - Uploading documents to collections in Document Manager..... 32
    - Bulk loading files into a collection in Document Manager..... 33
    - Creating advanced query filters..... 33
    - Configuring results cards..... 33
  - Managing diagnosis models in Diagnosis Model Manager..... 34
    - Best practices for diagnosis models..... 34
    - Creating a model..... 35
    - Defining the elements of a model..... 36
    - Defining the stages of a model..... 37

Creating a graph.....	37
Training settings.....	37
Loading and mapping data.....	38
Downloading trained diagnosis models for use offline.....	38
Upgrading diagnosis models.....	39
Managing diagnosis models that use version 1.0 of the API.....	39
Viewing Maximo Equipment Maintenance Assistant as a technician user.....	41
Simulating the technician user experience.....	41
Using the sample application.....	42
Querying in natural language and filtering the document collection.....	42
Using the sample application to diagnose problems.....	43
Using the SDK to access Maximo Equipment Maintenance Assistant.....	43
<b>Chapter 7. Integrating with Maximo Asset Management.....</b>	<b>45</b>
Configuring Maximo Asset Management to work with Maximo Equipment Maintenance Assistant.....	45
Creating a Maximo work order collection.....	45
Using historical Maximo work orders.....	46
Required work order fields.....	46
Downloading work orders by using the Maximo MXAPIWODETAIL API.....	47
Uploading historical Maximo work orders.....	49
Opening Maximo Equipment Maintenance Assistant from Maximo Asset Management.....	50
Automatically refreshing Maximo work orders .....	50
Using the Maximo Integration Framework.....	50
<b>Chapter 8. Application programming interface.....</b>	<b>53</b>
Trademarks.....	56
Terms and conditions for product documentation.....	56
IBM Online Privacy Statement.....	57



---

# Chapter 1. Product overview and features

By using IBM® Maximo® Equipment Maintenance Assistant, you can improve the availability and performance of equipment while increasing operational efficiency.

Maximo Equipment Maintenance Assistant is a cloud-based platform that incorporates IBM Watson® technology to augment your asset maintenance program with machine learning techniques and cognitive tools. This powerful combination provides you with capabilities to optimize asset repairs based on prescriptive guidance. Maximo Equipment Maintenance Assistant applies cognitive methods to structured and unstructured data associated with repairs, maintenance, procedures, and techniques. The result is enhanced insights and recommended optimum repair methods and procedures, helping you to detect failure patterns, ensure optimal first-time fixes (FTF), reduce mean time to repair (MTTR), and extend the life of critical assets.

---

## How it works

IBM Maximo Equipment Maintenance Assistant performs the analysis and cognitive methods and provides insight and recommendations based on the data that it has collected and the models that you have in place.

There are four main highlighted focus areas:

### **Data and document collection management**

IBM Maximo Equipment Maintenance Assistant provides you with the ability to identify optimal repair by searching a broad range of both structured and unstructured data. Structured data sources include items such as historical work orders, predictive alerts, and IoT sensor readings. Unstructured data sources include items such as owner's manuals, repair manuals, knowledge base and help desk content, technical diagrams, maintenance standards, root-cause failure analysis, failure modes and effects analysis, and regulatory standards. A business user prepares, uploads, and marks documents for ingestion.

### **Model development and refinement**

IBM Maximo Equipment Maintenance Assistant uses advanced Watson technology to understand and reason large collections of your private and public data. You define, train, and manage machine learning models using Watson reasoning tools such as IBM Watson™ Discovery and IBM Watson Assistant.

### **User, role, API, and services management**

IBM Maximo Equipment Maintenance Assistant includes an administrator portal that your administrator can use to add new users, assign them roles, and manage APIs and services. When integrating with IBM Maximo Asset Management, the administrator would install the Maximo Equipment Maintenance Assistant license key in IBM Maximo Asset Management.

### **End user experience**

Once IBM Maximo Equipment Maintenance Assistant contains the structured and unstructured data, has received the uploaded models, and has been configured with users and roles, the end user experience is straightforward. The field technician queries the system or requests a diagnosis and follows the prompts to receive the requested information. A field technician receives work orders and can start a conversation with Maximo Equipment Maintenance Assistant to understand repair procedures, identify parts, materials, and tools needed. The field technician can also update environment or situation information, feeding valuable information back to the system.

## What's new in Maximo Equipment Maintenance Assistant V 1.1.1

---

Learn about the updates for the current release of IBM Maximo Equipment Maintenance Assistant.

### What's new in Version 1.1.1

- The following enhancements improve the diagnosis process for subject matter experts and technicians:
  - Subject matter experts can use enhanced rule-based diagnosis in the Diagnosis Model Manager to quickly define and capture models that help field technicians with advanced troubleshooting.
  - Subject matter experts can use the new 1.1 version of diagnosis models with automated knowledge extraction, enabling them to create diagnosis models quickly by using information from common industry content. Diagnosis models can be created from information in tables within PDF or Excel files. Diagnosis stages enable field technicians to have a guided troubleshooting experience.
  - Technicians can download Maximo Equipment Maintenance Assistant diagnosis models to use offline in mobile apps.
- Maximo Equipment Maintenance Assistant is updated to support IBM Watson Discovery 2.1.2. For more information, see the [IBM Watson Discovery 2.1.2 release notes](#).
- Maximo Equipment Maintenance Assistant is available for installation on IBM Marketplace by using the IBM Entitled Registry, which enables installation from a central repository rather than requiring large downloads from Passport Advantage.

### New in Version 1.1.0

- Version 1.1.0 is the initial release of IBM Maximo Equipment Maintenance Assistant.
- The IBM Maximo Equipment Maintenance Assistant On Premises for Cloud Pak for Data solution is renamed to IBM Maximo Equipment Maintenance Assistant Add-On in release V1.1.0.
- Subject matter experts (SMEs) can upload ZIP files with multiple documents to collections in Document Manager.
- Document query results for users are updated to have a look and feel that is consistent with the Sample app interface.
- Elements in the Maximo Equipment Maintenance Assistant user interface are updated to comply with the Carbon 10 design system.
- Maximo Equipment Maintenance Assistant is certified for use on the Red Hat OpenShift container platform.
- Maximo Equipment Maintenance Assistant is certified for use with IBM Cloud Pak for Data.

### New in Version 1.0.0

#### Red Hat OpenShift certification

- Maximo Equipment Maintenance Assistant is certified for use on the Red Hat OpenShift container platform. Client can leverage the following market-leading capabilities and flexible deployment models of Red Hat OpenShift:
  - Leverage an AI-powered insight engine. Maintenance technicians can find answers to their questions in both structured and unstructured data, unified from multiple data silos in the enterprise.
  - Easily integrate the solution on-premises to help scale the return of answers.
  - Optionally deploy the solution on IBM's enterprise-ready, containerized software solutions running on Red Hat OpenShift on IBM Cloud or Red Hat Enterprise Linux.

## Document management

- By using Document Manager, you can collect and manage data such as work orders, repair manuals, technical diagrams, and regulatory standards. Documents are uploaded and prepared for use by field technicians.
- Field technicians can identify optimal repair procedures by searching through the document knowledge base. They can run queries by using the web interface and can integrate the AI recommendations to existing workflows.

## Diagnosis models

- By using the Diagnosis Model Manager, you can create models, define graphs of asset failure symptoms and causes, provide inspection instructions, assign Watson Discovery collections to prepare the historical data, train the model, and deploy the model.
- Field technicians use the Diagnosis model to diagnose equipment by observing key problems, confirming the correct diagnosis results, modifying the incorrect problems and results, and providing feedback to help improve training model accuracy.

## Integration with IBM Maximo Asset Management

- Integration of Maximo Equipment Maintenance Assistant with all of your existing systems includes a ready-to-use integration with Maximo Asset Management.
- Maximo Equipment Maintenance Assistant extracts repair recommendations from structured IBM Maximo Asset Management work order data, IoT sensor data, analytics output, and metadata associated with the equipment.
- When a work order is updated in Maximo Asset Management, it is automatically updated in Maximo Equipment Maintenance Assistant.

## Red Hat OpenShift Certification

Maximo Equipment Maintenance Assistant is certified for use with Red Hat OpenShift.

## Integration with Enterprise Asset Management solutions

---

IBM Maximo Equipment Maintenance Assistant provides out of the box integration with IBM Maximo Asset Management and other Enterprise Asset Management (EAM) solutions.

The user interface for the web application is an HTML5 enabled screen that is compatible with many browser types. These browsers range from a desktop to smartphone and tablet. In addition to providing repair recommendations, you can use existing asset and maintenance data to provide a complete workflow for the field technician thus increasing productivity.

## Accessibility features

---

Accessibility features help users who have a physical disability, such as restricted mobility or limited vision, to use information technology products.

For information about the commitment that IBM has to accessibility, see the [IBM Accessibility Center](http://www.ibm.com/able) ([www.ibm.com/able](http://www.ibm.com/able)).

HTML documentation has accessibility features. PDF documents are supplemental and, as such, include no added accessibility features.



## Chapter 2. Installing the solution

When you install IBM Maximo Equipment Maintenance Assistant, you install and configuring helm charts, install prerequisites, and configure dependencies for Maximo Equipment Maintenance Assistant on OpenShift.

### System requirements

Before you install IBM Maximo Equipment Maintenance Assistant, ensure that your servers meet the hardware and software requirements.

#### Hardware requirements

Maximo Equipment Maintenance Assistant hardware requires at least three worker nodes. Each node must meet the following minimum requirements:

- CPU: 16 cores
- RAM: 64 GB

#### Software requirements

The minimum software requirements for Maximo Equipment Maintenance Assistant are as follows:

- IBM Cloud™ Pak for Data 2.5.0 is installed
- Red Hat OpenShift 3.11 is installed
  - - Persistent Storage (GlusterFS / Ceph RBD / NFS) is configured on OpenShift Container Platform. For more information, see [Configuring Persistent Storage on OpenShift Container Platform 3.11](#).
  - Storage Class is created for dynamic provisioning. For more information, see [Dynamic provisioning and creating storage classes](#).
  - The dependencies of Maximo Equipment Maintenance Assistant require both ReadWriteOnce (RWO) and ReadWriteMany (RWX) access modes for persistent storage.
- Helm Client / Tiller Server Version 2.9.1
- Linux® 64-bit Red Hat Enterprise Linux (RHEL) 7.5 and higher

#### Supported web browsers

You can access the Maximo Equipment Maintenance Assistant from Mozilla Firefox, Google Chrome, and Safari.

#### Installation notes

The docker images of Maximo Equipment Maintenance Assistant should be loaded in to an appropriate Docker Image Repository before installation. On OpenShift, images can be loaded to the default Docker registry (docker-registry.default.svc:5000) in the intended namespace (for example, “ema”).

The following table shows the docker images of Maximo Equipment Maintenance Assistant:

Docker Image	Tag	Description
ema-api	1.1.1	RESTful APIs
ema-admin-console	1.1.1	Administrator console UI
ems-auth	1.1.1	EMA user authentication

Docker Image	Version	Description
ema-crawler	1.1.1	Crawler for indexing documents
ema-diagnosis	1.1.1	Diagnosis Engine
ema-diagnosis-dataloader	1.1.1	Diagnosis Data Loader
ema-diagnosis-evaluation	1.1.1	EMA diagnosis evaluation
ema-diagnosis-learning	1.1.1	EMA diagnosis learning
ema-landing-page	1.1.1	Landing-page UI
ema-maximo-integration	1.1.1	Maximo integration
ema-monitoring	1.1.1	Monitoring
ema-multi-tenant	1.1.1	Multiple tenant management
ema-sample-app	1.1.1	Sample Application
ema-studio	1.1.1	Studio UI
ema-service-provider	1.1.1	EMA Service Provider
opencontent-common-utils	1.1.2	Common Utils Toolkit on OpenShift

### Additional requirements and resources

For information about installing OpenShift Container Platform 3.11, see [Installing OpenShift Container Platform 3.11](#). The following table shows the minimum configuration that is required for installing the OpenShift Container Platform 3.11 for Maximo Equipment Maintenance Assistant:

Node	Number of Nodes	CPU (core)	RAM (GB)	Disk (GB)
HA Proxy	1	4	8	System: 100
Master	3	8	32	System: 100 Docker Storage: 300 Docker Registry: 300 (minimum) 700 (recommended)
Worker	3	32	128	System: 100 Docker Storage: 300 Data Storage: 1000

For information about requirements for installing IBM Cloud Pak for Data 2.5.0 on OpenShift, see the installation section of [Installing Cloud Pak for Data 2.5.0 on Red Hat OpenShift](#).

Maximo Equipment Maintenance Assistant does not provide anti-virus scan capabilities.

## Preparing for the installation

---

Before you install Maximo Equipment Maintenance Assistant, you must complete pre-installation steps.

### Before you begin

The following installation procedure is performed by the admin user with OpenShift 3.11 cluster administrator privileges.

### Procedure

1. Set up the environment:
  - a) Download and set up the "oc" client of OpenShift 3.11. For more information, see [oc Client Downloads](https://docs.openshift.com/container-platform/3.11/cli_reference/get_started_cli.html) or [https://docs.openshift.com/container-platform/3.11/cli\\_reference/get\\_started\\_cli.html](https://docs.openshift.com/container-platform/3.11/cli_reference/get_started_cli.html).
  - b) Log in to the Red Hat OpenShift Container Platform with **oc login** command.
  - c) Set up the Helm client and tiller server versions 2.9.1+ on OpenShift. For more information, see [Getting started with Helm on OpenShift](#).
2. Create the project (namespace) that you want to deploy Maximo Equipment Maintenance Assistant to.
  - a) Run the following command to request a new project:  
**oc new-project <project-name>**  
For example, if you want to deploy Maximo Equipment Maintenance Assistant in to a target namespace "ema", you can create the "ema" project by running the following command:  
**oc new-project ema**
  - b) Run the following command to switch to the target project (namespace):  
**oc project <project-name>**
3. Configure the Red Hat OpenShift SecurityContextConstraints requirements to bind the SecurityContextConstraints (SCC) to the users or group of the target namespace:
  - a) Run the following command to create the SecurityContextConstraint resource manually:

```
oc create -f - << EOF
allowHostDirVolumePlugin: false
allowHostIPC: true
allowHostNetwork: false
allowHostPID: false
allowHostPorts: false
allowPrivilegedContainer: false
allowedCapabilities:
- '*'
allowedFlexVolumes: null
apiVersion: v1
defaultAddCapabilities: null
fsGroup:
  type: RunAsAny
groups: []
kind: SecurityContextConstraints
metadata:
  annotations:
    kubernetes.io/description: emauid provides all features of the restricted SCC but
allows users to run with any UID and any GID.
    name: ibm-ema-prod-scc
  priority: 10
readOnlyRootFilesystem: false
requiredDropCapabilities: null
runAsUser:
  type: RunAsAny
seLinuxContext:
  type: MustRunAs
supplementalGroups:
  type: RunAsAny
users: []
volumes:
- configMap
- downwardAPI
```

```
- emptyDir
- persistentVolumeClaim
- projected
- secret
EOF
```

- b) Check whether the SecurityContextConstraint has been created successfully by running the following command:

```
oc get scc ibm-ema-prod-scc
```

- c) Choose one of the following options:

- To bind the SCC to the user group of the target namespace, run the following command, replacing *\$namespace* with the namespace that you're deploying to: **oc adm policy add-scc-to-group ibm-ema-prod-scc system:serviceaccounts:\$namespace**
- To bind the SCC to the default service account of the target namespace, run the following command: **oc adm policy add-scc-to-user ibm-ema-prod-scc system:serviceaccount:{namespace}:default**

- d) Create the cluster role binding for your default service account by running the following command:

```
oc create clusterrolebinding admin-on-ema --clusterrole=admin
--user=system:serviceaccount:{namespace}:default
```

- e) Create the pull secret for the OpenShift Container Platform Docker registry by running the following command, replacing {docker-registry-url} with the OpenShift Container Platform registry URL:

```
docker login {docker-registry-url} -u $(oc whoami) -p $(oc whoami -t)
oc create secret -n {namespace} docker-registry ema-emauid-docker-pull --docker-
server={docker-registry-url} --docker-username={docker-login-user} --docker-
password={docker-login-password}
```

#### 4. Load and push docker images into OpenShift Container Platform registry.

On Linux OS platform, you can use the `deploy.sh` file contained in the PPA archive file of Maximo Equipment Maintenance Assistant to load the Docker images and push them into the integrated container image registry on OpenShift automatically. To load the Docker images into the OpenShift Container Platform registry manually, for example, if you have exposed the OpenShift Container Platform registry with a route to host name `docker-registry-default.apps.cocstag.xyz`, complete the following steps:

- a) After logging in to the cluster with **oc login** command, log in to the integrated OpenShift Container Platform registry by running the following command:

```
docker login docker-registry-default.apps.cocstag.xyz -u $(oc whoami) -p $(oc whoami -t)
```

- b) After extracting the PPA archive file of Maximo Equipment Maintenance Assistant, find the archive files of Docker images in `IBM_EMA_V1.1_LINUX_ML/ibm-ema-prod/images`. Load the images by running the following command:

```
docker load -i {image-name}_{image-tag}.tar.gz
```

- c) Re-tag the Docker images by running the following command:

```
docker tag {image-name}:{image-tag} docker-registry-
default.apps.cocstag.xyz/{namespace}/{image-name}:{image-tag}
```

For example, if you want to push the Docker image `ema-api:1.1.1` into the “ema” namespace, run the following command:

```
docker tag ema-api:1.1.1 docker-registry-default.apps.cocstag.xyz/ema/ema-
api:1.1.1
```

- d) Push images into the OpenShift Container Platform registry by running the following command:

```
docker push docker-registry-default.apps.cocstag.xyz/{namespace}/{image-
name}:{image-tag}
```

- e) Check the loaded images by running the following command .

```
oc get imagestreams -n {namespaces}
```



- f) Create and add the docker pull secret that is used for the EMA namespace and Cloud Pak for Data service account.

```
oc create secret docker-registry ema-anyuid-docker-pull --docker-server=docker-registry.default.svc:5000 --docker-username='oc whoami' --docker-password='oc whoami -t' -n {EMA_namespace}
oc secrets add serviceaccount/default secrets/ema-anyuid-docker-pull --for=pull -n {EMA_namespace}
oc create secret docker-registry ema-anyuid-docker-pull --docker-server=docker-registry.default.svc:5000 --docker-username='oc whoami' --docker-password='oc whoami -t' -n {ICP4D_Namespace}
oc secrets add serviceaccount/cpd-admin-sa secrets/ema-anyuid-docker-pull --for=pull -n {ICP4D_Namespace}
```

## Installing and configuring Helm charts

After the Helm charts are installed, IBM Maximo Equipment Maintenance Assistant is deployed in an IBM Cloud Private environment with a default pattern.

Only one instance of Maximo Equipment Maintenance Assistant can be deployed in a namespace, while multiple instances can be deployed across different namespaces.

When you configure the installed Helm charts, you change the values of the configurable parameters. For more information, see [“Helm chart configurable parameters”](#) on page 14. To change the value of a parameter from its default value, add `--set <parameter-name>=<parameter-value>` to the Helm **install** command:

```
helm install --name {release_name} --namespace {namespace_name} ibm-apm-ema-prod --tls
```

For example, you can change the ingress host by adding `--set ingress.hostname={your_host_name}` to the Helm **install** command:

```
helm install --name {release_name} --namespace {namespace_name} ibm-apm-ema-prod --tls --set ingress.hostname={your_host_name}
```

## Configuring dependencies

Before you install the helm charts for IBM Maximo Equipment Maintenance Assistant, you must configure information about the installed dependencies of an instance.

During the installation, the instance of Maximo Equipment Maintenance Assistant is provisioned automatically with the configured dependency information. Change dependency information, including service URL and credentials, in the `data/ema-instance-config.json` file in the helm chart.

After you extract the PPA archive file of Maximo Equipment Maintenance Assistant, the helm chart is in the `IBM_EMA_V1.1_LINUX_ML/ibm-ema-prod/charts` directory.

Run the **tar -zxvf ibm-apm-ema-prod-1.1.1.tgz** to extract the archive file `ibm-apm-ema-prod-1.1.1.tgz`.

In the `data/ema-instance-config.json` file, specify the required configuration for the instance of Maximo Equipment Maintenance Assistant:

`serviceInstanceDisplayName`: The instance name of Maximo Equipment Maintenance Assistant

Parameter	Description	Required
<code>couchdb.username</code>	The username of CouchDB for instance of Maximo Equipment Maintenance Assistant	Y

Table 2. Dependency configuration parameters (continued)

couchdb.password	The password of CouchDB for instance of Maximo Equipment Maintenance Assistant	Y
couchdb.serviceInstance	The CouchDB service URL. This value can be internal cluster IP https://{couchdb-service-name}. {namespace}.svc:5984 or host that is shown with a reencrypting route of CouchDB service.	Y
couchdb.certificate	The CouchDB certificate. The certificate is required if CouchDB is self-signed with a third-party tool.  <b>Note:</b> The certificate must begin with "-----BEGIN CERTIFICATE-----" and end with "-----END CERTIFICATE-----", replacing the line-break with '\n'.	N
domain	The IBM Cloud Pak for Data external domain host, such as "zen-lite-cpd-zen-lite.apps.cocstag.xyz".	Y
objectStorage.username	The username of S3 compatible object storage. The user must have permissions to create and delete buckets, put / get / list / delete objects.	Y
objectStorage.password	The password of S3 compatible object storage.	Y
objectStorage.serviceInstance	The S3 compatible object storage URL. For example, the shown host of the gluster S3 service with secure route: https://glusters3-app-storage.apps2.cocstag.xyz	Y
objectStorage.certificate	The S3 compatible object storage certificate. The certificate is required if CouchDB is self-signed with a third-party tool.  <b>Note:</b> The certificate must begin with "-----BEGIN CERTIFICATE-----" and end with "-----END CERTIFICATE-----", replacing the line-break with '\n'.	N
discovery.username	The username of Watson Discovery for IBM Cloud Pak for Data	Y

discovery.password	The password of Watson Discovery for IBM Cloud Pak for Data	Y
discovery.version	The API version of Watson Discovery for IBM Cloud Pak for Data. For example, "2019-06-28".	Y
discovery.serviceInstance	The instance URL of Watson Discovery for IBM Cloud Pak for Data.	Y

**Note:**

Ensure that the credentials and information of the dependencies are correct before you install IBM Maximo Equipment Maintenance Assistant. Otherwise, the connectivity check during the installation will fail.

**Note:**

If you need to ingest a large quantity of files in bulk to Watson Discovery by using the IBM Maximo Equipment Maintenance Assistant Studio UI or API, configure tuning parameters in the crawler deployment.

The following tuning configuration steps are recommended for Watson Discovery 2.1.2 with [High availability \(Production\) configuration](#):

1. Run the following command:

```
oc edit deployment <ema-release>-ema-crawler -n <ema-namespaces>
```

2. Enter `i` and increase the two environment variables in `spec.template.spec.containers`:

```
- name: CRAWL_BATCH
  value: "100"
- name: CHECK_BATCH
  value: "100"
```

3. Save the change and wait for the crawler pod to run.

## Installing Helm charts manually

Installing the Helm chart deploys a single instance of IBM Maximo Equipment Maintenance Assistant into the cluster on OpenShift Container Platform.

After you log in to the cluster on OpenShift Container Platform and load and pushing the Docker images, you can install the Helm charts.

If this is the first time that you are installing Maximo Equipment Maintenance Assistant, use the following command to install the Helm charts:

```
helm install --name {release_name} --namespace {namespace_name} ibm-apm-ema-prod --set global.license=accept
```

The `{release_name}` indicates the Helm release name and `{namespace_name}` indicates the target namespace that you plan to deploy Maximo Equipment Maintenance Assistant to.

If you installed previous versions of Maximo Equipment Maintenance Assistant, such as version 1.1.0, use the following command to upgrade the Helm charts installation:

```
helm upgrade {previous_release_name} -f values.yaml ibm-apm-ema-prod --set global.license=accept --namespace {namespace_name}
```

The {previous\_release\_name} indicates the Helm release name that you previously used for the old installation of Maximo Equipment Maintenance Assistant in the cluster. The {namespace\_name} indicates the target namespace that you plan to deploy Maximo Equipment Maintenance Assistant to.

To change the value of a parameter from its default value, add **--set <parameter-name>=<parameter-value>** or pass an overriding .yaml file with --values flag to the Helm install command.

## Installing Helm charts with deploy.sh on Linux

If you are installing IBM Maximo Equipment Maintenance Assistant on OpenShift Container Platform for the first time, you can deploy Maximo Equipment Maintenance Assistant by using the deploy.sh script that is contained in the PPA archive file.

### About this task

When you use deploy.sh, you do not need to load and push Docker images into OpenShift Container Platform Registry before you install the Helm charts. The Docker images that are contained in the PPA archive file are loaded and pushed into Docker registry automatically, and the Helm chart is installed automatically.

After you extract the PPA archive file of Maximo Equipment Maintenance Assistant, you can find the deploy.sh file in the IBM\_EMA\_V1.1\_LINUX\_ML/deploy directory.

### Procedure

1. Go to the IBM\_EMA\_V1.1\_LINUX\_ML/deploy/utils directory.
2. Run the **./helm list** command to check whether the Helm client binary file is executable and works properly.  
If the command doesn't run successfully, copy the Helm client binary file that was installed when you set up the environment into the utils folder to ensure that the command can be run successfully.
3. Complete a Docker login to the OpenShift Container Platform Registry with the following command:  
**docker login <docker-registry-url> -u \$(oc whoami) -p \$(oc whoami -t).**
4. Go to the IBM\_EMA\_V1.1\_LINUX\_ML/deploy/utils/deploy-utils directory.
5. Edit the install.yaml file. Change the following values in the file:

#### **docker.registry.externalAddress**

The external address of the OpenShift Container Platform registry.

#### **docker.registry.address**

The internal address of the OpenShift Container Platform registry.

#### **installer.namespace**

The target namespace (for example, "ema") into which you intend to deploy the IBM Maximo Equipment Maintenance Assistant On-Premises.

#### **helm.tillerNamespace**

The namespace in which the Helm tiller pod is running in.

#### **helm.tls.enabled**

Whether TLS is enabled for Helm.

Example:

```
docker:
  registry:
    externalAddress: "docker-registry.default.svc:5000/ema"
    address: "docker-registry.default.svc:5000/ema"
  installer:
    namespace: "ema"
  helm:
```

```
tillerNamespace: "tiller"
tls:
  enabled: false
```

6. Edit the `global.yaml` file. The `global.yaml` file contains the values that are defined in the charts/`ibm-apm-ema-prod/values.yaml` file, which is used for installing the Helm chart. For example, change the following values as necessary:

```
global:
  jksJob:
    image:
      repository: docker-registry.default.svc:5000/ema/opencontent-common-utils
      tag: 1.1.2
    icpDockerRepo: docker-registry.default.svc:5000/ema/
  spJob:
    image:
      repository: ema-service-provider
      tag: 1.1.1
  sch:
    enabled: true
    icp4dNamespace: zen
    license: accept
  ...
tenantMgmtDB:
  username: admin
  password: changeme
  endpoint: https://ema-tenantdb-route-ema.apps2.cocstag.xyz
  ...
```

7. Run the following command to load and push images and install the Helm chart:

```
./deploy.sh -d ../ibm-ema-prod -e {release-name} -n {tiller-namespace}
```

## Verifying the Helm chart installation

After the Helm chart installation is complete, you can verify the installation.

Verify the status of the Helm charts installation by running the following command:

```
helm status {release-name}
```

You also can test your Helm installation by running the following command:

```
helm test {release_name} --cleanup
```

**Note:** Add `--cleanup` to the command to delete the testing pods after the command is run.

You can also check the deployed Kubernetes resources by running one of the following commands:

- `oc get all -n {namespace}`
- `kubect1 get all -n {namespace}`

## Configuring Helm charts

When you configure the installed Helm charts, you change the values of the configurable parameters.

For more information, see [“Helm chart configurable parameters” on page 14](#).

To change the value of a parameter from its default value, add `--set <parameter-name>=<parameter-value>` to the Helm **install** command:

```
helm install --name {release_name} --namespace {namespace_name} ibm-apm-ema-prod --tls
```

For example, you can change the ingress host by adding `--set ingress.hostname={your_host_name}` to the Helm **install** command:

```
helm install --name {release_name} --namespace {namespace_name} ibm-apm-ema-prod --tls --set ingress.hostname={your_host_name}
```

## Helm chart configurable parameters

When you configure IBM Maximo Equipment Maintenance Assistant Helm charts, you can change the default parameter values.

The following section describes the configurable parameters and default values for Maximo Equipment Maintenance Assistant Helm charts.

Parameters	Description	Default
global.license	Users must read the license, set the value to accepted, and configure the correct value for the <b>global.isAddOnSwidTag</b> parameter based on the license.	not accept
global.isAddOnSwidTag	After users read and accept the license, they must set this value to <code>false</code> to install Maximo Equipment Maintenance Assistant or to <code>true</code> to install the Maximo Equipment Maintenance Assistant add-on.	false
global.jksJob.image.repository	The image name for common utilities for running Kubernetes jobs on OpenShift.	docker-registry.default.svc:5000/ema/opencontent-common-utils
global.jksJob.image.tag	The tag of common utilities image.	1.1.2
global.icpDockerRepo	The docker image repository. For each image, the full image name is prefixed by this value.	docker-registry.default.svc:5000/ema/
global.icp4dNamespace	The namespace into which IBM Cloud Pak for Data is deployed.	zen
global.spJob.image.repository	The docker image name of the EMA instance service provider.	ema-service-provider
global.spJob.image.tag	The image tag of the EMA instance service provider.	1.1.1
tenantMgmtDB.username	The administrator user name of the global tenant management DB.	empty
tenantMgmtDB.password	The administrator password of the global tenant management DB.	empty
tenantMgmtDB.endpoint	The endpoint for the global tenant management DB.  The endpoint can be an internal cluster IP of CouchDB service (for example, <code>https://{couchdb-service-name}.{namespace}.svc:5984</code> ), or the host exposed using a re-encrypting route.	empty

Table 3. Maximo Equipment Maintenance Assistant configurable Helm chart parameters (continued)

tenantMgmtDB.certificate	Required if the global tenant management CouchDB is self-signed with a third-party tool.	empty
emaAdminConsole.replicaCount	The replica count of deployment for Admin Console.	3
emaAdminConsole.image.repository	The image name of the Admin Console.	ema-admin-console
emaAdminConsole.image.tag	The image tag of Admin Console.	1.1.1
emaAdminConsole.image.pullPolicy	The Pull Policy of the Admin Console image.	IfNotPresent
emaApi.replicaCount	The replica count of deployment for API.	3
emaApi.image.repository	The image name of the API.	ema-api
emaApi.image.tag	The image tag of API.	1.0.0
emaApi.image.pullPolicy	The Pull Policy of the API image.	IfNotPresent
emaAuth.image.repository	The image name of the EMA user authentication.	ema-auth
emaAuth.image.tag	The image tag of the EMA user authentication.	1.1.1
emaAuth.image.pullPolicy	The image pull policy of the EMA user authentication.	IfNotPresent
emaCrawler.replicaCount	The replica count of deployment for Crawler.	3
emaCrawler.image.repository	The image name of Crawler.	ema-crawler
emaCrawler.image.tag	The image tag of Crawler	1.1.1
emaCrawler.image.pullPolicy	The Pull Policy of the Crawler image.	IfNotPresent
emaDiagnosis.replicaCount	The replica count of deployment for Diagnosis Engine.	3
emaDiagnosis.image.repository	The image name of Diagnosis Engine.	ema-diagnosis
emaDiagnosis.image.tag	The image tag of Diagnosis Engine.	1.1.1
emaDiagnosis.image.pullPolicy	The Pull Policy of the Diagnosis Engine image.	IfNotPresent
emaDiagnosisDataLoader.replicaCount	The replica count of deployment for Diagnosis Data Loader.	3
emaDiagnosisDataLoader.image.repository	The image name of Diagnosis Data Loader.	ema-diagnosis-dataloader
emaDiagnosisDataLoader.image.tag	The image tag of Diagnosis Data Loader.	1.1.1

Table 3. Maximo Equipment Maintenance Assistant configurable Helm chart parameters (continued)

emaDiagnosisDataLoader.image.pullPolicy	The Pull Policy of the Diagnosis Data Loader image.	IfNotPresent
emaDiagnosisLearning.replicaCount	The replica count of the Diagnosis Learning service.	3
emaDiagnosisLearning.image.repository	The repository for the Diagnosis Learning service.	ema-diagnosis-learning
emaDiagnosisLearning.image.tag	The tag for the Diagnosis Learning service.	1.1.1
emaDiagnosisLearning.image.pullPolicy	The image pull policy for the Diagnosis Learning service.	IfNotPresent
emaDiagnosisEvaluation.replicaCount	The replica count of the Diagnosis Evaluation service.	3
emaDiagnosisEvaluation.image.repository	The repository for the Diagnosis Evaluation service.	ema-diagnosis-evaluation
emaDiagnosisEvaluation.image.tag	The tag for the Diagnosis Evaluation service.	1.1.1
emaDiagnosisEvaluation.image.pullPolicy	The image pull policy for the Diagnosis Evaluation service.	IfNotPresent
emaLandingPage.replicaCount	The replica count of deployment for Landing Page.	3
emaLandingPage.image.repository	The image name of Landing Page.	ema-landing-page
emaLandingPage.image.tag	The image tag of Landing Page.	1.1.1
emaLandingPage.image.pullPolicy	The Pull Policy of the Landing Page image.	IfNotPresent
emaMaximoIntegration.replicaCount	The replica count of deployment for Maximo Integration.	3
emaMaximoIntegration.image.repository	The image name of Maximo Integration.	ema-maximo-integration
emaMaximoIntegration.image.tag	The image tag of Maximo Integration.	1.1.1
emaMaximoIntegration.image.pullPolicy	The Pull Policy of the Maximo Integration image.	IfNotPresent
emaSampleApp.replicaCount	The replica count of deployment for Sample Application.	3
emaSampleApp.image.repository	The image name of Sample Application.	ema-sample-app
emaSampleApp.image.tag	The image tag of Sample Application.	1.1.1
emaSampleApp.image.pullPolicy	The Pull Policy of the Sample Application image.	IfNotPresent



*Table 3. Maximo Equipment Maintenance Assistant configurable Helm chart parameters (continued)*

emaStudio.replicaCount	The replica count of deployment for Studio.	3
emaStudio.image.repository	The image name of Studio.	ema-studio
emaStudio.image.tag	The image tag of Studio.	1.1.1
emaStudio.image.pullPolicy	The Pull Policy of the Studio image.	IfNotPresent
emaMultiTenant.replicaCount	The replica count of deployment for Multi-tenant management.	3
emaMultiTenant.image.repository	The image name of Multi-tenant management.	ema-multi-tenant
emaMultiTenant.image.tag	The image tag of Multi-tenant management.	1.1.1
emaMultiTenant.image.pullPolicy	The Pull Policy of the Multi-tenant management image.	IfNotPresent
emaMonitor.replicaCount	The replica count of deployment for Monitoring.	3
emaMonitor.image.repository	The image name of Monitoring.	ema-monitor
emaMonitor.image.tag	The image tag of Monitoring.	1.1.1
emaMonitor.image.pullPolicy	The Pull Policy of the Monitoring image.	IfNotPresent

## Managing Helm releases

After the IBM Maximo Equipment Maintenance Assistant Helm chart installation is complete, you can view and manage the installed Helm release.

### View installed Helm release

To view the installed Helm release, run the following command:

```
helm list
```

### Uninstall the Helm release

To uninstall the Helm release, run the following command:

```
helm delete --purge {release_name}
```

### Upgrade the Helm release

To upgrade the installed Helm release, run the following command:

```
helm upgrade {release_name} ibm-apm-ema-prod --namespace {namespace_name}
```

### Rollback the Helm release

To roll back the Helm release to a previous revision, can run the following command:

```
helm rollback {release_name} {revision_name}
```

### View a list of revision numbers

To view a list of the revision numbers, run the following command:

```
helm history {release_name}
```

## Uninstalling the solution

You can uninstall the IBM Maximo Equipment Maintenance Assistant solution by uninstalling the Helm release.

To uninstall the deployment of Maximo Equipment Maintenance Assistant, uninstall the Helm release by running the following command:

```
helm delete --purge {release-name}
```

## Installing prerequisites

---

After you install the helm charts for Maximo Equipment Maintenance Assistant, you must install prerequisites before you can provision an instance of the solution on IBM Cloud Pak for Data.

### About this task

You must install the following prerequisites for each instance you provision:

- CouchDB 2.3.1 on Red Hat OpenShift
- Watson Discovery 2.1.2 for IBM Cloud Pak for Data 2.5.0
- S3 Compatible Object Store on Red Hat OpenShift

## Installing CouchDB on Red Hat OpenShift

You must install CouchDB 2.3.1 on Red Hat OpenShift 3.11 before you can provision an instance of the IBM Maximo Equipment Maintenance Assistant solution on IBM Cloud Pak for Data.

### Before you begin

#### Procedure

1. To install CouchDB 2.3.1 on Red Hat OpenShift 3.11, see the installation guide [Installing the Operator for Apache CouchDB on Red Hat OpenShift 3.11](#).
2. Deploy CouchDB cluster on Open Shift.  
For more information, see [Deploying a CouchDB Cluster on OpenShift](#).
3. Access CouchDB 2.3.1.  
You can expose the CouchDBCluster using a reencrypting route or by port-ward the service using kubectrl command.
4. Create a CouchDBCluster for global use of tenant management with the following minimum requirements:
  - Size: 3
  - CPU (core): 1
  - Memory: 1
  - Disk: 1Gi
5. Create a CouchDBCluster for use by the automatically provisioned instance of Maximo Equipment Maintenance Assistant during installation, with the following minimum requirements:
  - Size: 3
  - CPU (core): 1
  - Memory: 1

- Disk: 10Gi

## Installing Watson Discovery 2.1.2 for IBM Cloud Pak for Data

You must install Watson Discovery 2.1.2 for IBM Cloud Pak for Data 2.5.0 before you can provision an instance of the IBM Maximo Equipment Maintenance Assistant solution on IBM Cloud Pak for Data

To install IBM Watson Discovery 2.1.2 for IBM Cloud Pak for Data 2.5.0, refer to the “Installing” section of the document [Getting started with Watson Discovery for IBM Cloud Pak for Data](#), or refer to the [Helm chart README.md](#) for details.

## Installing S3 Compatible Object Store on Red Hat OpenShift

You must install S3 Compatible Object Store on Red Hat OpenShift before you can provision an instance of the IBM Maximo Equipment Maintenance Assistant solution on IBM Cloud Pak for Data.

You can use one of the following two types of S3-Compatible Object Store on OpenShift 3.11 Maximo Equipment Maintenance Assistant:

- Red Hat Ceph Storage: For more information, see [Installation Guide for Red Hat Enterprise Linux](#).
- Gluster S3 Object Store: For more information, see [S3 Compatible Object Store in a Red Hat Openshift Container Storage Environment](#).

**Note:** For Gluster S3 Object Store, GlusterFS storage must be configured on OpenShift Container Platform. For more information, see [Persistent Storage Using Red Hat Gluster Storage](#) for details.

It is recommended that you create a [secured route](#) for Gluster S3 Service on OpenShift with edge termination.



---

## Chapter 3. Administering the solution

Administration tasks for IBM Maximo Equipment Maintenance Assistant include managing API keys and users and performing monitoring and logging.

### Generating API keys

---

API keys are used to run the Maximo Equipment Maintenance Assistant API calls. You must generate API keys before you can use the API.

#### Procedure

1. Go to the **Admin Console** tab.
2. Launch the administration console.
3. On the **API Key** tab, click **Add New**.
4. Enter a name and optional description for the API key.
5. Specify one or more roles to be associated with the API key.
6. Save the API key.
7. Copy the API key. The API key is displayed only one time and can't be retrieved later.

#### What to do next

You can change the associated roles of an API key by clicking ... and selecting **Edit**. You can delete an API key by clicking ... and selecting **Delete**.

### Adding and managing users

---

New users can sign up to IBM Maximo Equipment Maintenance Assistant and administrators can then approve the users through IBM Cloud Pak for Data and assign them roles in Maximo Equipment Maintenance Assistant.

#### Procedure

1. A new user signs up for Maximo Equipment Maintenance Assistant:
  - a) On the **Login** page, click **Sign up**.
  - b) Enter the user information and log-in credentials.
  - c) Click **Sign up** to complete the sign-up process.

The sign-up request is sent from Maximo Equipment Maintenance Assistant to the Cloud Pak for Data Administrator console.

After the user signs up, they cannot access Maximo Equipment Maintenance Assistant until the administrator approves the new user and grants them user roles.
2. The Cloud Pak for Data administrator approves the new user in Cloud Pak for Data:
  - a) Log in to Cloud Pak for Data.
  - b) On the Cloud Pak for Data **Administer** page, select **Manage users**.
  - c) From the list of users, find the new user whose status is Pending.
  - d) Edit the user and change the **Account status** to Approved.
  - e) Save the changes to the user.

For more information about how to approve users, see "Managing users" in the Cloud Pak for Data Knowledge Center ([https://www.ibm.com/support/knowledgecenter/SSQNUZ\\_2.5.0/cpd/admin/users.html](https://www.ibm.com/support/knowledgecenter/SSQNUZ_2.5.0/cpd/admin/users.html)).

3. The Maximo Equipment Maintenance Assistant administrator grants the newly approved user permissions to use Maximo Equipment Maintenance Assistant:
  - a) Log in to Maximo Equipment Maintenance Assistant.
  - b) Click **Admin console** and then open **Administrator**.
  - c) On the **User** tab of the **Administrator** page, click **Add new**.
  - d) Type the username of the new user.
4. Choose one or more roles for the new user from the following options:
  - Admin Console: The user can access the **Admin console** user interface.
  - Studio: The user can access the **IBM® Maximo Equipment Maintenance Assistant Studio** user interface.
  - Sample Application: The user can access the **End user simulator** user interface.
5. Save the changes.

The user can now sign in and use Maximo Equipment Maintenance Assistant.

### What to do next

On the **User** tab of the **Administrator** page, the administrator can proceed to edit the user by changing their display name or roles or delete the user.

## Configuring SAML users

---

If you have a Security Assertion Markup Language (SAML) provider, you can configure IBM Cloud Pak for Data to enable SAML authentication to Cloud Pak for Data and IBM Maximo Equipment Maintenance Assistant.

### Before you begin

You must follow the steps in Cloud Pak for Data documentation to enable SAML authentication. For more information about enabling SAML, see "Configuring SSO" in the Cloud Pak for Data Knowledge Center: [https://www.ibm.com/support/knowledgecenter/en/SSQNUZ\\_2.5.0/cpd/install/saml-ss0.html](https://www.ibm.com/support/knowledgecenter/en/SSQNUZ_2.5.0/cpd/install/saml-ss0.html). Complete these steps before you begin the procedure to enable SAML:

1. Run the following command:

```
oc exec -it -n namespace $(oc get pod -n namespace -l component=usermgmt | tail -1 | cut -f1 -d \ ) -- bash -c "vi /user-home/_global_/config/saml/samlConfig.json"
```

If an error results from this command, run the following command to create saml folder:

```
oc exec -it -n namespace $(oc get pod -n namespace -l component=usermgmt|tail -1 | cut -f1 -d \ ) -- bash -c "mkdir -p /user-home/_global_/config/saml"
```

2. Run the following command, using the same namespace:

```
oc -n <namespace> delete pods -l component=usermgmt
```

### Procedure

1. The SAML administrator adds a user in SAML.

After the user is added in SAML, they cannot access Maximo Equipment Maintenance Assistant until the administrator approves the new user and grants them user roles.
2. The Cloud Pak for Data administrator approves the new user in Cloud Pak for Data:

- a) Log in to Cloud Pak for Data.  
If SAML is enabled, you can log in by using `<your-host>/auth/login/zen-login.html`.
- b) On the Cloud Pak for Data **Administer** page, select **Manage users**.
- c) Add the SAML user to the Cloud Pak for Data user list.

**Note:** Cloud Pak for Data accepts only email addresses that contain two or three letters for the top-level domain (for example, com, net, gov, ed, org, and other domain suffixes). Add a two- or three-letter domain to the end of addresses in the **Email** field. The username, not the email address, is used for the login and it does not need to follow the same format with a two- or three-letter domain at the end.

- d) Save the changes to the user.

For more information about how to approve users, see "Managing users" in the Cloud Pak for Data Knowledge Center ([https://www.ibm.com/support/knowledgecenter/SSQNUZ\\_2.5.0/cpd/admin/users.html](https://www.ibm.com/support/knowledgecenter/SSQNUZ_2.5.0/cpd/admin/users.html)).

3. The Maximo Equipment Maintenance Assistant administrator grants the newly approved user permissions to use Maximo Equipment Maintenance Assistant:
  - a) Log in to Maximo Equipment Maintenance Assistant.
  - b) Click **Admin console** and then open **Administrator**.
  - c) On the **User** tab of the **Administrator** page, click **Add new**.
  - d) Type the username of the new user.
4. Choose one or more roles for the new user from the following options:
  - Admin Console: The user can access the **Admin console** user interface.
  - Studio: The user can access the **IBM® Maximo Equipment Maintenance Assistant Studio** user interface.
  - Sample Application: The user can access the **End user simulator** user interface.
5. Save the changes.  
The user can now sign in and use Maximo Equipment Maintenance Assistant.

### What to do next

On the **User** tab of the **Administrator** page, the administrator can proceed to edit the user by changing their display name or roles or delete the user.

## Monitoring and logging

---

To perform monitoring and logging for IBM Maximo Equipment Maintenance Assistant, you must install a separate stand-alone Prometheus and Grafana server

OpenShift comes with a Prometheus instance available. However, this instance is optimized for instrumentation of the Kubernetes system itself. As a result, to monitor Maximo Equipment Maintenance Assistant, you must install and configure the separate stand-alone Prometheus server in the same project where Maximo Equipment Maintenance Assistant is installed.

Maximo Equipment Maintenance Assistant provides application and instance level monitoring metrics and export to `/metrics` by using the **npm prom-client** module in the `ema-monitor` service so that the stand-alone Prometheus server can pull it after your configuration. You can check the exported metrics details by using the `ema-monitor` service: <https://ema-monitor-service:443/metrics>. The `ema-monitor` service is not shown as an OpenShift container platform route, so you can use the port forwarding feature to export to your local port.

You also can deploy the separate Grafana/prometheus instance for Maximo Equipment Maintenance Assistant.

## Prometheus installation and configuration

OpenShift provides [prometheus templates](#) and [grafana templates](#) to support the installation of Prometheus and Grafana on OpenShift.

After you have Prometheus or Grafana installed, config your Prometheus scrape config file to contain the ema-monitor-service metrics.

```
scrape_configs:
  - job_name: 'ema'
    static_configs:
      - targets: ['ema-monitor-service:443']
    scheme: https
    tls_config:
      cert_file: /etc/tls/private/tls.crt
      key_file: /etc/tls/private/tls.key
      insecure_skip_verify: true
```

### Tips

- Install the stand-alone Prometheus server to the same namespace as Maximo Equipment Maintenance Assistant so that you do not have to replace `ema-monitor-service:443` with `ema-monitor-service.<EMA_NAMESPACE>.svc.cluster.local:443` in the configuration. Use the EMAMaximo Equipment Maintenance Assistant namespace to replace `<EMA_NAMESPACE>`.
- The certificate (`tls.crt`, `tls.key`) is the OpenShift self-signed certificate and is created as secrets `prometheus-tls` during the installation of the Prometheus server. The certificate is usually mounted to the path in the Prometheus oauth proxy container `prom-proxy`. Modify the statefulset by running **`oc edit statefulset prom`** and then mount it to the Prometheus container `prometheus`.
- Ensure that the Args `-openshift-delegate-urls={"/": {"resource": "namespaces", "verb": "get"}}` is configured to the Prometheus oauth proxy container `prom-proxy`. Give the Prometheus service account `prom` the auto delegation role by running **`oc adm policy add-cluster-role-to-user system:auth-delegator system:serviceaccount:<EMA_NAMESPACE>:prom`** so the Prometheus server can work within the OpenShift auth framework.
- You also can add Prometheus to Grafana as a data source and import the predefined Maximo Equipment Maintenance Assistant dashboard data/`ema-grafana-dashboard.json` in the Maximo Equipment Maintenance Assistant packages.

### Logging

Maximo Equipment Maintenance Assistant sends its logs to the standard output, and you can deploy the cluster logging to see the Maximo Equipment Maintenance Assistant logs. You can import the predefined Maximo Equipment Maintenance Assistant kibana dashboard data/`ema-kibana-dashboard.json`.



---

## Chapter 4. Backing up and restoring data

You must back up IBM Maximo Equipment Maintenance Assistant data manually in case you need to recover it after a disaster.

The backup and restore process includes the following data:

- Global embedded CouchDB databases and instance-level CouchDB databases
- [IBM Watson Discovery for IBM Cloud Pak for Data](#)
- OpenShift GlusterFS or Ceph Object Storage

For backup and restore for IBM Cloud Pak for Data, refer to [Back up and restore your Cloud Pak for Data](#).

### Process Overview

Before you perform the backup of CouchDB, IBM Watson Discovery for IBM Cloud Pak for Data, and Object Storage, ensure that there are no written tasks for Maximo Equipment Maintenance Assistant

Reinstall Maximo Equipment Maintenance Assistant and restore the data to the corresponding CouchDB, IBM Watson Discovery for IBM Cloud Pak for Data, and Object Storage.

### Required tools

Before you begin backing up or restoring data, ensure that you have the following tools installed on your local machine:

- The OpenShift Client command line interface .CLI.
- The Kubernetes command line tool kubectl.

---

## Backing up and restoring CouchDB data

To back up CouchDB data, use CouchDB replication to back up the data to another CouchDB installation. You can choose between a normal (one-shot) replication or continuous replications.

Set up a separate CouchDB instance as the target backup instance that your existing CouchDB databases will be replicated to. Create replication tasks on the replication portal by using the following URL:

```
https://<Your_CouchDB_URL>/_utils/index.html#/replication
```

The following types of CouchDB databases must be backed up:

### Global embedded CouchDB database

The global CouchDB database is preinstalled and configured by the user to save global tenants and configuration information. The global CouchDB has the following five databases: "api-keys", "roles", "tenants", "token", "users".

### Instance-level CouchDB database

The instance-level CouchDB database is preinstalled and configured by user before the instance provisioning, and used when the instance is created. The instance-level CouchDB has the following databases: "crawler", "diagnosis-history", "diagnosis-model", "document-management" and "api-calls", and several "api-calls-xxxx-xx".

Use CouchDB replication to recover the backup databases from the backup CouchDB instance. For more information about CouchDB backup and Recovery, see the [CouchDB Documentation](#).

### Note:

The CouchDB database encryption password DB\_ENCRYPT\_SECRET in the kube secret ema-secret needs to be backed up manually because the CouchDB data is encrypted with it and must use the same password during the EMA recovery. For example:

```
$ oc get secret ema-secret -o yaml -n emadev
apiVersion: v1
data:
DB_ENCRYPT_SECRET: NMMUIDkwMkMtQzhDMi00NDg0LTkyMDQtQTM1RkMxNDEwQjQ0
```

As an alternative solution for CouchDB backup and recovery, you can copy the actual (.Couch) files from the CouchDB data directory (by default, data/) at any time. For more information, see the [CouchDB Documentation](#).

## Backing up and restoring Watson Discovery data

You must back up your Watson Discovery for IBM Cloud Pak for Data instance in case you need to restore it.

For more information about backing up and restoring data for Watson Discovery, see [Backup and restore data of Watson Discovery for IBM Cloud Pak for Data](#).

## Backing up and restoring OpenShift GlusterFS or Ceph Object Storage data

To back up and restore Ceph Object Storage or GlusterFS data, set up the IBM Maximo Equipment Maintenance Assistant instance-level data store, and then back up the data manually.

### Before you begin

Complete step 1 to back up and restore Ceph Object Storage data and complete step 2 to back up and restore GlusterFS data.

### Procedure

1. For Ceph Object Storage, complete the following steps to use the minIO client to back up all of the buckets in the object storage:
  - a) Add the ceph service and use the **mc mirror** command to back up all buckets to local computers:

```
mc config host add <ALIAS> <YOUR-S3-ENDPOINT> <YOUR-ACCESS-KEY> <YOUR-SECRET-KEY> <API-SIGNATURE>
```

For example:

```
$ mc config host add ceph http://IP_Address:7480 52YSQ3BBB4KE7YD09UJB
n0wweoKuKZUs6j0AWiNN7NC3DLb1zBzc0vccSKaX
$ mc ls ceph
$ mc mirror ceph/collection.9118a890-c7d3-11e9-9fee-dd2153b895b1.testxw01/
collection.9118a890-c7d3-11e9-9fee-dd2153b895b1.testxw01/
```

**Note:** You might need to add the option `--insecure` if your host certificate is self signed.

- b) Repeat to mirror all Ceph service buckets to local computers.
- c) To restore the local backed up to the new Ceph services buckets, use the following command:

```
$ mc mb ceph/collection.9118a890-c7d3-11e9-9fee-dd2153b895b1.testxw01/
$ mc mirror collection.9118a890-c7d3-11e9-9fee-dd2153b895b1.testxw01 ceph/
ccollection.9118a890-c7d3-11e9-9fee-dd2153b895b1.testxw01/
```

For more information, see [MinIO Client Complete Guide](#).

2. For Gluster S3 Object Storage, complete the following steps to use the S3cmd tool to back up all of the buckets in the object storage:

- a) Directly add the corresponding entry to the `~/ .s3cfg` file and set “signature\_v2 = True”. Gluster S3 supports S3V2 only.
- b) To list the buckets in the remote gluster S3 endpoint, run the following command:  
**s3cmd ls s3:// --no-check-certificate**
- c) To sync a specific bucket from a remote gluster S3 endpoint to a local system for backup, run the following command:  
**s3cmd sync s3://collection.6c1afec0-2873-11ea-a894-a37bc870f378.1577428678828 collection.6c1afec0-2873-11ea-a894-a37bc870f378.1577428678828 --no-check-certificate**
- d) To sync a specific bucket from a local gluster S3 to a remote gluster S3 for recovery, run the following command:  
**s3cmd sync collection.6c1afec0-2873-11ea-a894-a37bc870f378.1577428678828 s3://collection.6c1afec0-2873-11ea-a894-a37bc870f378.1577428678828 --no-check-certificate**

For more information, see [S3cmd tool portal](#).



---

## Chapter 5. Building models

Train models and configure documentation so that Maximo Equipment Maintenance Assistant has better query accuracy. Use Watson Discovery to configure and query document collections. Document collections can be associated with collections that you create in Maximo Equipment Maintenance Assistant.

### Manage documents in Watson Discovery

---

Use Watson Discovery to manage your document collections and to create, configure, and query documents in Maximo Equipment Maintenance Assistant. Watson Discovery provides custom entity and relations enrichment.

Watson Discovery allows you to run queries with more in-depth search capabilities for aggregate and analysis results to better understand your documentation.

When you create a collection in the Maximo Equipment Maintenance Assistant studio, you can associate it with a Watson Discovery document collection.

For more information about how to use Watson Discovery, go to the [Watson Discovery](#) documentation page.

**Note:** Maximo Equipment Maintenance Assistant is not able to identify text in image files that are used in a Watson Discovery collection.



---

## Chapter 6. Using the product

IBM Maximo Equipment Maintenance Assistant uses natural language for you to query the documentation collections for the location, equipment, and parts of equipment.

Maximo Equipment Maintenance Assistant needs a model that describes the items that are referenced in the documentation and the relationship between the items.

The relationships in the model can be visualized graphically.

---

### Initializing your environment

To initialize your environment, you must activate your administrator account and access Watson Discovery and Watson Assistant.

#### Activating your administrator account

Before you use Maximo Equipment Maintenance Assistant for the first time, you must activate the administrator account.

##### Procedure

1. Go to the Maximo Equipment Maintenance Assistant home page.
2. Click **Sign up**.
3. Enter the name, email address and password for the administrator account. The email address must be the administrator email address you specified when you requested the environment.
4. Click **Sign up**.  
You'll receive a confirmation email.
5. Click **Verify** in the confirmation email to verify the administrator account.

##### Results

After you complete these steps, you can log in to the product using the administrator email address.

---

### Managing documents in Document Manager

You manage documents in Maximo Equipment Maintenance Assistant. These documents are used to query information that helps field technicians have accurate information for repairs.

#### About this task

You must create collections to manage documents in the Document Manager.

##### Procedure

1. From the Maximo Equipment Maintenance Assistant landing page, click **Launch** under **Document Manager**.
2. Click **Create** to create a collection.
3. In the **Create Collection** box, specify a name for the collection.
4. Specify a collection type:
  - Select **Maximo Work Orders** if you plan to create a collection for uploading Maximo work order data.
  - Select **Documents** if you plan to upload unstructured data, including PDF, Word, HTML, and JSON files.

5. In the **Associated Watson Discovery data collection** field, you must select a data collection. If you do not have a data collection, select **Create Watson Discovery collection** and create one. If there already is an associated Watson Discover collection, you can enable the collection for use in queries.
6. Click **Create** to create the collection.

## Results

After you create a document collection, you can add documents to the collection, upload new document versions, manage the document metadata, and delete documents. You can create advanced query filters, and you can configure how the results are shown to users on cards. You can also expand a query to define synonyms to make the query more accurate and split large documents into smaller parts to make queries more accurate. For example, set **Unify Segment Format** to **On** to enable document splitting and select the heading level to split the document into segments. If you want to apply changed configurations to old documents, you can re-index the documents. Select **Clear all documents** to clear all the indexed documents in Watson Discovery.

## Related concepts

### Building models

Train models and configure documentation so that Maximo Equipment Maintenance Assistant has better query accuracy. Use Watson Discovery to configure and query document collections. Document collections can be associated with collections that you create in Maximo Equipment Maintenance Assistant.

## Uploading documents to collections in Document Manager

You can upload documents to work order or document collections in the Maximo Equipment Maintenance Assistant Document Manager. You can also upload .zip files that contain documents.

### Procedure

1. After you create a document collection in the Document Manager, select **Add new** to upload documents to the collection.
2. Select up to 10 PDF, Word, HTML, or JSON files that are 32 MB or smaller to upload, or select one .zip file to upload. The .zip file must be 50 MB or smaller, can contain up to 1000 files that are 32 MB or smaller, and cannot contain any files that are already in the collection.

If you upload a .zip file to a document type collection, ensure that the following requirements are met:

- The .zip file can contain only PDF, Word, HTML, and JSON files.
- If you want to specify metadata, the document ID, and the description, you must create an `index.csv` file and add it to the .zip package. The `index.csv` must contain a header and content, as shown in the following example. The documentID must be a number, letter, or dash (-), and the length is 1 - 36. Metadata keys cannot start with an underscore (\_). Use ";" as the metadata separator and use ":" between the metadata key and value.

```
fileName,documentId,description,metadata
Sample_ABB Drive_ACS880_FW_manual-1.pdf,1234567-ABB,it's a description,zztest:zztest;inZip:true
Sample_ABB Drive_ACS880_FW_manual-2.pdf,,zztest:zztest;inZip:true
Resp-1.json,wo-1234567-Resp-1,it's a description,zztest:zztest;inZip:true
Resp-2.json,,
```

*Figure 1. Index CSV example*

If you upload a .zip file to a work order type collection, ensure that the following requirements are met:

- The .zip file can contain only JSON files.
- The use of an `index.csv` file is not supported with work order collections.



## Bulk loading files into a collection in Document Manager

You can load multiple files into a Maximo Equipment Maintenance Assistant Document Manager collection at one time.

### Procedure

Run the python **import** command, as shown the following sample code for importing JSON files:

```
import requests

url = 'https://<host>/api/v1/document-management/collections/0b1e1930-
bf33-11e9-93b0-67c9323a4ec3/documents'
files = {'file': ('W02001.json', open('<filepath>/<filename>'), 'application/json')}

headers = {
    'X-API-KEY': <api key>,
    'X-TENANT-ID': <tenant id>,
    'accept': 'application/json'
}

r = requests.post(url, files=files, headers=headers)
print(r.json())
```

## Creating advanced query filters

You can set up advanced query filters in IBM Maximo Equipment Maintenance Assistant to provide users with more accurate search results.

### Procedure

1. Open the document collection that you want to create an advanced query for.
2. In the Document Manager main navigation pane, click **Advanced query filter**.
3. Click **Add filter** to open the filter editor.
4. Type a filter name that is no more than 60 characters long.
5. Add rules to the filter:
  - a) In the Configure filter rules section, click **Add rule** or **Add group of rules**.
  - b) Specify whether the filter must satisfy all of the rules you are adding or any of the rules.  
If you add a group of rules, you can specify whether the rules within that group must satisfy "or" or "and" conditions.
  - c) Select the field that the rule applies to, the operator, and the value type.  
A value type of **Constant** means that users who are running the query must select or enter a value.  
A value type of **Variable** means that users must enter a variable name and select value criteria.
6. Save the filter.

## Configuring results cards

You can configure query results cards to determine how results are shown to users.

### Procedure

1. Open the document collection that you want to configure results cards for.
2. In the Document Manager main navigation pane, click **Configure result card**.
3. Select whether you want the results cards to show structured data.
4. If structured data is turned on, enter at least one row of structured data results configuration:
  - a) Click **Add result**.
  - b) Type a name that is no more than 60 characters long.
  - c) Select the data field.
  - d) Optional: Add more results fields and reorder the results.

5. Select whether you want the results cards to show a highlighted summary.
6. Save the results configuration.
7. Test the configuration by typing a keyword to search for and press Enter.  
You can view the search results, and you can determine whether you need to modify them.

## Managing diagnosis models in Diagnosis Model Manager

---

Manage diagnosis models and user feedback data in the Diagnosis Model Manager. You create models, define the model elements and stages, test and complete graphs, load training data, and train and deploy the models.

Diagnosis models use the structured and unstructured data that is associated with repairs, maintenance, procedures, and techniques to provide insights in to problems and recommend solutions and repair methods to field technicians. Diagnosis models that use the version 1.1 API use the following elements:

- Observations are elements that can be directly observed and measured, or inferred from another observation. Observations can help field technicians to identify failure causes.
- Causes identify the reason for the failure.
- Recommendations identify potential solutions and courses of action to deal with the failure.

When you define a diagnosis model, you can use a graph that is based on the domain knowledge and then train the model with historical and feedback data to enhance the model accuracy. You also can manage a diagnosis model by extracting data from documents such as equipment handbooks and training materials and by using the extracted knowledge to define the relationships between observations and causes.

The tasks in this section describe how to use the Diagnosis Model Manager with models that use the current version of the API - version 1.1. For instructions on how to use Diagnosis Model Manager with models that use the previous version of the API - version 1.0, see [“Managing diagnosis models that use version 1.0 of the API”](#) on page 39. For instructions on how to upgrade models to the new version, see [“Upgrading diagnosis models”](#) on page 39.

### Best practices for diagnosis models

When you develop a diagnosis model, you create symptoms and causes, connect the causes to symptoms that they can manifest, define probabilities, and train the model. You can improve model accuracy by following modeling best practices.

When you create your model, start with issues that are challenging for the customers. These issues might be symptoms or causes that result in loss of time, can be costly if the diagnosis is wrong, or are difficult to diagnose. Collect a small set of related symptoms and causes first. Start with two or three related symptoms. Next, add the common causes. Then, add any other symptoms of the causes. Continue with this model structure as you build your model. You must first build your model structure. After you build your model structure, you add probability estimates.

In a model for diagnosis, a symptom is an observable phenomenon. A cause is a problem that can cause symptoms or other problems to occur. For example, when you diagnose a computer problem, you might notice that the computer is slow to start. A slow startup might be an observable symptom. A cause for the slow startup might be a hard disk failure. A root cause is a problem that has nothing that causes it that is listed in the model.

When you are modeling for diagnosis, you must follow the principles of causality and conditional independence. Causality means that edges connect causes to their effects, which typically means symptoms. Conditional independence means that symptoms and causes are independent of all other nodes except their descendants given the value of their direct parent nodes.

It is important to use a strategy when modeling for diagnosis so that you can improve model accuracy, which can result in a better diagnosis and can save unnecessary costs. Start with a small model, evaluate the model with an expert, and extend the model incrementally. Ask the expert for probability estimates to get a usable model quickly. After you get a usable model, use feedback to improve the model. Also, use

historical data to extract symptoms and causes to train the model and to improve the accuracy of models that are already running.

Modeling for diagnosis includes defining probability estimates. You define the probability configuration to describe how likely it is that a symptom or cause occurs. When you define the estimate, ask for numeric estimates and not vague designations. When you input the information for conditional probability and probability due to other causes, you must be input the conditional probability first. If a probability estimate for a symptom or cause that has a parent is above 10 percent, there are likely missing causes. You might also be missing causes if you make your estimates too soon. Do not use probability estimates of 0 percent or 100 percent because no observations are absolutely certain.

Use the following probability definitions when you are defining your probability configurations:

<i>Table 4. Probability Definitions</i>		
<b>Probability type</b>	<b>Network element</b>	<b>Definition</b>
Edge probability	Cause to symptom	The probability that the symptom will occur if the cause is the only parent cause of the symptom that occurred.
Probability due to other causes	Symptom	The probability that the symptom will occur when none of its parent causes in the model occurred, which means the symptom occurred because of some cause other than the parent causes.
Root-cause probability	Cause without parents	The prior probability that the root cause will occur when there is an issue, but no information about the symptoms is available.
Edge probability	Cause to cause	The probability that the child cause will occur when the parent cause is the only cause that occurred.
Probability due to other causes	Cause with parents	The probability that the child cause will occur when none of its parents occurred, which means the child cause occurred because of some cause other than the parent causes.

## Creating a model

Create a model from an existing template, a local file, or from scratch by starting with a blank form.

### Procedure

1. In the Diagnosis Model Manager, on the **All Models** tab, click **Create**.
2. In the **Create new model** window, type the model name.
3. Select whether you want to create a model from an existing template or file, and then select the template or file. If you want to create a model from scratch, select **Blank**.
4. Click **Create**.

## Results

You can view the new model in models table. You can proceed to edit, duplicate, export, or delete existing models. Editing the model includes defining the model elements and stage, creating graphs, loading training data, and managing instances.

## Defining the elements of a model

After you create a model or view an existing model, define the elements of the model. Elements include observations that help to prompt technicians when they are diagnosing problems, possible causes of the problems, and recommendations with instructions for resolving problems.

### Procedure

1. In the model, select **Definition** and then select **Model elements**.
2. Click **Add** to add elements to the model.
3. Select **Create observation** and complete the fields to add the observation element to the model. Add a name, description, tags and supporting data that technicians use to diagnose and address problems in the field, including the following details:

#### Data type

- **Yes/No** Indicates that the value of the observation can be Yes or No only. Only this type of observation can be treated as a symptom that is used by the graph. For example, you can create an observation that is called "engine overheat" and then select **Yes/No** so that a technician in the field selects an answer based on whether the symptom is present or not.
- **List**: Indicates that the value of the observation is a string of characters. For example, you can create an observation that is called "asset type" and then a technician in the field might type "gould pump" as the asset type value.
- **Numeric**: Indicates that value of the observation is a measurable numeric value. For example, you might create an observation that is called "pressure" and then a technician in the field might enter a pressure value of 34.5 with a unit of kPa.

#### Observation type

- **Observation**: Indicates a raw observation in which the value is decided on and entered by the technician.
- **Inferred observation**: Indicates an observation that is calculated based on other raw observations. For more information about how to define an inferred observation formula, including comparison operators and expressions, see [JSONata Documentation](#). Use the observation name in the formula definition, and use "\$" if the name contains a blank character or a non-English character. The following list shows examples of inferred observations:
  - Pressure > 30
  - \$. 'driver temperature' > 100
  - Pressure > 34
  - Pressure > 34 and Pressure < 60
  - (Pressure + 100)/2.4

#### Choices

Specify possible choices or options of observation values that a technician in the field can select.

4. Select **Create cause** and complete the fields to add the cause element to the model. If one observation or a combination of observations can have one definite cause, you can complete the **Cause configuration** section to make the cause a rule. To define a rule-based cause, you enter the same type of formulas that you use to define inferred observations. For more information, see step 3. However, you must define rule-based causes as Yes/No questions. Link to related recommendations for a cause. This helps technician users to get more detailed repair or inspection instructions after the root causes are diagnosed.

5. Save the added element.

### What to do next

You can view, edit, and activate the elements that you add and you can add more elements or remove elements from the model. Proceed to add stages to the model.

## Defining the stages of a model

After you create a model or when you view an existing model, define the stages of the model. Stages are used to narrow down potential problems to help technicians diagnose and resolve issues in the field.

### Procedure

1. In the model, select **Stage**.
2. Click **Create stage** to get started and generate the first stage for the model.
3. Edit the details of the first stage, adding in related observations and any other stages that this stage is dependent on.
4. Save the new stage.

### What to do next

You can add more stages, remove stages, edit the details of stages, and turn on or turn off stages to be used with the model. Proceed to add graphs to the model.

## Creating a graph

After you create a model with defined elements and stages, or when you view an existing model, you can create and test graphs that describe failure symptoms and problems and provide repair instructions for each problem.

### About this task

When you create a graph, improve model accuracy by following modeling best practices. For more information about the best practices for diagnosis models, go to the [Best practices for diagnosis models](#) topic.

### Procedure

1. In the model, select **Graph**.
2. Click **Add nodes** to add causes or symptoms to the graph.
3. Select problems and symptoms in the graph to edit information about them, such as probability.

## Training settings

Define the training settings to determine the data that you want to use to train the model. After the model is trained, it is used by field technicians. You can review the diagnosis logs and continuously fine-tune the model.

In a model, on the **Training Settings** tab, input the equivalent sample size, specify whether user feedback and historical data should be used when training the model, and whether the model needs to be run offline.

After you define the training settings, select **Train** and create a model instance. On the **All Instances** tab, view a summary of the trained model instance. Select **Activate** to activate the model. When a model is activated, it is used by the field technicians to diagnose problems.

## Loading and mapping data

You load data that you want to use to train the model. To train the model, map historical, unstructured data to the symptoms and problems that were defined in the graph. Define mapping rules to increase training accuracy. You can also use previous feedback and examples from technicians to train the model.

### Procedure

1. In the model, select **Data**.
2. On the **Historical data** tab, select **Load historical data**.
3. Assign a collection. The assigned collections are based on the historical data that you uploaded in the Document Manager.
4. Configure the mapping rules. Select a symptom or cause and select **Mapping**.
5. In the **Edit Mapping Rules** pane, define the mapping rules.
  - a) Select **Enhance by Watson Natural Language Query** and input keywords.
  - b) Select **Enable mapping rules** and add or edit the mapping rules.
  - c) Select **Test** to view the historical data that matches the mapping rules that you selected. If the results are not accurate, update the mapping rules.
  - d) Select **Save and Next** to map another symptom or cause.
  - e) Select **Ok** when you are finished mapping.
6. Select **Load Data**. After the data loads, view the mapped historical data.
7. On the **Feedback** tab, review the field technician feedback. The feedback helps train the model and increases model accuracy. You can confirm the feedback and use only the confirmed feedback when you define the training settings for the model. You can also use feedback to retrain the model.

## Downloading trained diagnosis models for use offline

An offline diagnosis model is a neural network model in formats that can be directly deployed on iOS or Android mobile devices. You can download Maximo Equipment Maintenance Assistant diagnosis models for technicians to use offline in mobile apps.

### Before you begin

To be able to download trained diagnosis models for offline use, the models must be generated with the offline use setting on during the training process.

The training data that is used for estimating its parameters is simulated based on the trained Bayesian network. As a result, the expected prediction that is made by the neural network model is close to the Bayesian network's prediction. However, factors, such as the training data size and sampling algorithms that the neural network depends on, might influence the gap between the prediction results from neural network and the Bayesian network models.

### Procedure

1. After a training process ends, find the model that you want to download in the **All trained instances** list.
2. Find the correct instance of the model, click the **More** action, and select **Download offline model**.  
A .zip file is downloaded with the following contents:
  - For iOS apps, the neural network model converted from a trained Bayesian network in Core ML format.
  - For Android apps, the neural network model converted from a trained Bayesian network in TensorFlow Lite format.
  - A JSON configuration file with information about the mapping between the names and numeric IDs of nodes.
  - The neural network model.

3. Insert these neural network models into your customized app.

Follow the guidelines on that are outlined in the Core ML and TensorFlow Lite documentation to perform inference tasks using these models.

## Upgrading diagnosis models

If you have diagnosis models that use the version 1.0 API, you can upgrade the models to use the version 1.1 API.

### About this task

You upgrade the diagnosis models by creating new 1.1.1 models based on 1.1.0 models or templates. The **API Version** column on the **All Models** tab in the Diagnosis Model Manager indicates the API version that is used with each model.

### Procedure

1. In the Diagnosis Model Manager, on the **All Models** tab, click **Create**.
2. In the **Create new model** window, type the name of the model you are upgrading.
3. Select whether you want to create the 1.1.1 model from an existing 1.1.0 model or from a 1.1.0 template, and then upload the model or select the template.
4. Select **Convert it to newest API(V1.1)**.
5. Click **Create**.

### Results

The new 1.1.1 diagnosis model, which uses the version 1.1 API, is created based on the existing 1.1.1 model. You can proceed to edit and manage the upgraded model.

## Managing diagnosis models that use version 1.0 of the API

These tasks describe how to use the Diagnosis Model Manager with models that use the previous version of the API- version 1.0.

To take advantage of added features, upgrade your existing models so that they use the latest API version. For instructions on how to use Diagnosis Model Manager with models that use the upgraded API - version 1.1, including instructions on how to upgrade models to the new version, see [“Managing diagnosis models in Diagnosis Model Manager” on page 34](#).

### Creating a model

Create a model from an existing template, a local file, or from scratch.

### Procedure

1. In the Diagnosis Model Manager, on the **All Models** tab, select **Create**.
2. In the **Create new model** box, input the model information, and select **Create**.

### Results

Select the **All models** tab to view the models. You can edit, duplicate, export, or delete existing models.

### Creating a graph

After you create a model or view an existing model, create and test graphs that describe failure symptoms and problems, and provide repair instructions for each problem.

### About this task

When you create a graph, improve model accuracy by following modeling best practices. For more information about the best practices for diagnosis models, go to the [Best practices for diagnosis models](#) topic.

## Procedure

1. In the model, select **Graph**.
2. Select **Add nodes** to add a cause node or a symptom node to the graph.
  - a) On the **General** tab, input the information.
  - b) On the **Cause** tab, select the possible causes and then the prior probability or the probability due to other causes. The probability is set to unknown by default. You can edit the probability if necessary.
  - c) On the **Reference** tab, input keywords for the querying support documents and add repair instructions.
3. Select **Test Graph**. Select problems and symptoms to view how the causes and symptoms appear in the graph, how they are connected, and to see if any causes or symptoms are missing.
4. Select **Notifications** to view error and warning messages about the graph.

## Training settings

Define the training settings to determine the data that you want to use to train the model. After the model is trained, it is used by field technicians. You can review the diagnosis logs and continuously fine-tune the model.

In a model, on the **Training Settings** tab, input the training settings for the graph, the training settings for the data, and the data training policies. You can select which settings you want to define.

For the graph training settings, define the training settings for the expert estimation that you defined on the **Graph** tab, such as the cause probability. By default, you can use only the expert estimation weight to train the model.

For the data training settings, define the settings for the information that you defined on the **Data** tab, such as the user feedback, the mapped historical data, and the pre-defined examples.

For each weight value, define the value as a relative number to the other weight values that you defined. For example, if the weight value for the expert estimation is 10 and the weight value for pre-defined examples is 100. These weight values would show that the pre-defined examples are 10 times more important than the expert estimation.

For the train policies setting, define the setting for the number of cross validations. The cross validation value combines the measure of fitness in a prediction to derive an accurate estimate of the model prediction performance. The value represents the training process that is repeated to improve accuracy.

After you define the training settings, select **Train** and create a model instance. On the **All Instances** tab, view a summary of the trained model instance. Select **Activate** to activate the model. When a model is activated, it is used by the field technicians to diagnose.

## Loading and mapping data

You load data based on the data that you want to use to train the model. To train the model, map historical data that is unstructured to the symptoms and problems that were defined in the graph. Define mapping rules to increase training accuracy. You can also use feedback and examples to train the model.

## Procedure

1. In the model, select **Data**.
2. On the **Historical data** tab, select **Load historical data**.
3. Assign a collection. The assigned collections are based on the historical data that you uploaded in the Document Manager.
4. Configure the mapping rules. Select a symptom or cause and select **Mapping**.
5. In the **Edit Mapping Rules** pane, define the mapping rules.
  - a) Select **Enhance by Watson Natural Language Query** and input keywords.
  - b) Select **Enable mapping rules** and add or edit the mapping rules.
  - c) Select **Test** to view the historical data that matches the mapping rules that you selected. If the results are not accurate, update the mapping rules.



- d) Select **Save and Next** to map another symptom or cause.
  - e) Select **Ok** when you are finished mapping.
6. Select **Load Data**. After the data loads, view the mapped historical data and define the weight.
  7. On the **Feedback** tab, review the field technician feedback. The feedback helps train the model and increases model accuracy. You can confirm the feedback and use only the confirmed feedback when you define the training settings for the model. You can also use feedback to retrain the model. You can update the data policy to view or edit the technician weight policy. If a diagnosis record is common, select **Save as example** and input the information on the **Create Example** pane.
  8. On the **Example** tab, add examples by selecting the add button. Examples help train the model if there is a lack of historical data and feedback.

### Using the sample application with API version 1.0 to diagnose problems

Use IBM Maximo Equipment Maintenance Assistant to diagnose problems and to find possible causes.

#### About this task

Follow this procedure when you are using diagnosis models that use version 1.0 of the API. To diagnose problems by using version 1.1 of the API, see [“Using the sample application to diagnose problems”](#) on page 43.

#### Procedure

1. Select **Diagnosis**.
2. Select the diagnosis type.
3. Investigate the equipment, select and search the symptoms that you see, and select **Start**.
4. On the **Results** page, review the possible causes and the probability. Select or remove any possible causes or add additional symptoms.
5. Select a possible cause to view more detailed information.
  - a) Select **Repair instruction** to review the repair instructions. The repair instructions are predefined in the model.
  - b) Select **Evidence** to view the probability tree. The probability tree describes the problems, symptoms, and the relationship and probability of those symptoms.
  - c) Select **Support documents** to view returned results. If you think that the returned result is accurate, select the thumbs up icon. If you think that the returned result is not accurate, select the thumbs down icon. The feedback helps refine the models.
6. After you review the repair instructions, evidence, and support documents for a possible cause, go to the **Results** page and select another possible cause or symptom and select **End**. Input the feedback information and select **Send**.

## Viewing Maximo Equipment Maintenance Assistant as a technician user

---

IBM Maximo Equipment Maintenance Assistant is available for various media formats, desktop, mobile phone and tablet.

If you want to view what the technician user experiences, Maximo Equipment Maintenance Assistant simulates and formats the screen for either mobile phone and tablet format.

### Simulating the technician user experience

IBM Maximo Equipment Maintenance Assistant simulates the technician user experience for either a tablet or mobile phone formats.

#### Procedure

1. From the home page, click the **End User Simulator** tab.
2. Click either **Mobile Simulator** or **Tablet Simulator**.

## Results

You view Maximo Equipment Maintenance Assistant depending on the format you choose.

## Using the sample application

---

Use IBM Maximo Equipment Maintenance Assistant to search documentation collections with natural language queries and to diagnose problems.

### Querying in natural language and filtering the document collection

IBM Maximo Equipment Maintenance Assistant uses natural language queries to search previous work orders, key performance indicators, and document collections for a type of equipment.

#### About this task

You can use a filter to restrict a search to certain collections. However, there must always be one collection that is selected. Maximo Equipment Maintenance Assistant uses a term frequency-inverse document frequency algorithm. The term frequency returns a higher weighting score to documents that return the more frequently used terms throughout the document collection. The inverse document frequency removes words from the search that are common to all documents.


If the search returns many instances of the search item from the database, you can define the document collections that are referenced by the search in the source filter feature. Use the search function to type your own contextual words or phrases. For example, to know more about an installation or a piece of equipment, you can type `install equipment ID` where the equipment ID is the number of the equipment that you are interested in.

#### Procedure

1. Select the collection that you want to query on.
2. In the **What do you want to search for** field, type a keyword or phrase to do a search.
3. Click **Query**.

The related results are returned by the greatest similarity values first.

4. If the search returns many results, you can filter the results by specifying the document collections to

do the search on. To change the collection settings, click the filter icon  and set the filter option to off for collections as necessary.

5. Click **Save**.

The filter reduces the search items.

## Results

For each collection, the query results are displayed in each tab.

### Running advanced query filters

Advanced query filters can help you obtain more accurate search results. If your administrator set up advanced query filters, you can set the values to search on for certain fields, and you can enable the filters to run when you search for keywords.

#### Procedure

1. At the top of your query results, click **Configure**.
2. In the **Query filter** window, turn on filters that you want to use and turn off filters that you don't want to use.
3. Optional: Modify the filter values.
4. Click **Apply** to run the query with the advanced filters.

### What to do next

If you think that the returned result is accurate, select the thumbs up icon. If you think that the returned result is not accurate, select the thumbs down icon. The feedback helps the train the models. After the training, the model is given a confidence score.

You can download the original document or select **Read all** to open the document online.

## Using the sample application to diagnose problems

Use IBM Maximo Equipment Maintenance Assistant to diagnose problems and to find possible causes and solutions.

### About this task

Follow this procedure when you are using diagnosis models that use version 1.1 of the API. To diagnose problems by using version 1.0 of the API, see [“Using the sample application with API version 1.0 to diagnose problems”](#) on page 41.

### Procedure

1. In Maximo Equipment Maintenance Assistant, select **Diagnosis**.
2. Select the type of diagnosis that you want to perform on the asset.
3. Select the diagnosis stage that you want to start at.
4. Investigate the equipment and identify the symptoms that you notice. Follow the prompts to add more information about the symptoms, and go back and investigate symptoms as instructed.
5. Start the diagnosis.
6. On the **Results** page, review the possible causes and the probability of each cause. You can go back and add or remove symptoms and run the diagnosis again.
7. From the results, select a possible cause to view more detailed information.
8. Select and view the instructions and supporting documentation for possible solutions.
9. After you review the instructions, evidence, and support documents and have attempted to complete the solution, answer the questions and provide information about whether the cause was accurate and you were able to resolve the issue.

The information that you enter helps to train the model and provide more accurate results for future diagnosis.

## Using the SDK to access Maximo Equipment Maintenance Assistant

---

You can use Maximo Equipment Maintenance Assistant SDK to access Maximo Equipment Maintenance Assistant capabilities.

### About this task

The Maximo Equipment Maintenance Assistant node.js SDK is provided out of the box and can be downloaded from the Maximo Equipment Maintenance Assistant landing page. For other programming language SDKs, you can obtain the Swagger definition files from the Maximo Equipment Maintenance Assistant swagger API UI and generate the corresponding SDKs by using the Swagger Editor. Before you use the Maximo Equipment Maintenance Assistant node.js SDK, ensure that node.js is installed on your computer.

### Procedure

1. Download Maximo Equipment Maintenance Assistant node.js SDK, which is a file such as `equipment-advisor-node-sdk-1.1.0.tgz` by clicking **Download SDK** on the Maximo Equipment Maintenance Assistant landing page.
2. Copy the file `equipment-advisor-node-sdk-1.1.0.tgz` to your node.js project and add it to the "dependencies" of `package.json`.

For example:

```
"dependencies":{ "equipment-advisor-node-sdk":"./equipment-advisor-node-sdk-1.1.0.tgz" }
```

3. Run **npm install** to install the Maximo Equipment Maintenance Assistant EMA SDK dependencies.

## Results

You can start writing code to access Maximo Equipment Maintenance Assistant capabilities. The following sample code shows how to use the Maximo Equipment Maintenance Assistant node.js SDK:

```
const SDK = require('equipment-advisor-node-sdk');
const sdk = new SDK({
  endpoint: '<EMA endpoint, e.g. https://ea-prod-cluster.us-south.containers.appdomain.cloud >',
  tenantId: '<tenant id>',
  apiKey:
    '<API Key generated in EMA Administor Console>'
});
const DocumentManagementAPI = sdk.DocumentManagementV1;
DocumentManagementAPI.listCollections()
  .then(collections => console.log(collections.map(coll => coll.name)))
  .catch(err => console.log(err));
```

## Note:

The capabilities of Maximo Equipment Maintenance Assistant APIs are different for SaaS and on-premises products. For more information, see the Maximo Equipment Maintenance Assistant API Swagger UI in your environment.

---

# Chapter 7. Integrating with Maximo Asset Management

You can integrate Maximo Equipment Maintenance Assistant with Maximo Asset Management. When the products are integrated, you can import work orders from Maximo Asset Management into Maximo Equipment Maintenance Assistant, and you can start Maximo Equipment Maintenance Assistant from a Maximo Asset Management work order.

## Configuring Maximo Asset Management to work with Maximo Equipment Maintenance Assistant

---

You can configure Maximo Asset Management to work with Maximo Equipment Maintenance Assistant by setting a few system properties. This integration enables the Seek Advice button on work orders that starts Maximo Equipment Maintenance Assistant.

### Before you begin

Before you start this task, get a Maximo integration API key from the Maximo Equipment Maintenance Assistant administration console.

### About this task

Maximo Equipment Maintenance Assistant integrates with Maximo Asset Management version 7.6.1 or later.

### Procedure

1. Log in to Maximo Asset Management as a user with administrative privileges.
2. Open the System Properties application.
3. Filter the system properties by using the string "ea .".
4. For the ea.url system property, enter the Maximo application URL for Maximo Equipment Maintenance Assistant:  
`https://ema_url/maximo`  
where *ema\_url* is the URL for the Maximo Equipment Maintenance Assistant server.
5. For the ea.apikey system property, enter the Maximo integration API key that you obtained from the Maximo Equipment Maintenance Assistant administration console.

## Creating a Maximo work order collection

---

Before you add Maximo work orders to Maximo Equipment Maintenance Assistant, you must create a collection for the work orders.

### Procedure

1. In Equipment Maintenance Assistant Studio, launch the Document Manager.
2. Select **Create** to create a collection.
3. In the **Create Collection** dialog box, enter the collection information. For collection type, specify Maximo Work Orders.
4. Select an associated Watson Discovery data collection. If there is no associated Watson Discovery data collection, create one.
5. Select **Create** to complete the collection creation.

## Using historical Maximo work orders

You can download work orders from Maximo to a JSON file. You can then upload the work orders to a Maximo work order collection in Maximo Equipment Maintenance Assistant.

### Required work order fields

Certain fields are required to use Maximo work orders with Maximo Equipment Maintenance Assistant.

The following work order fields are required:

Field	Description
wonum	Work order number.
description	Description of the work order.
description_longdescription	Long description of the work order.
status	Work order status.
status_description:	Description of the work order status.
owner	Owner of the work order.
reportdate	Reported date.
actstart	Actual start date.
targstartdate	Target start date.
schedstart	Scheduled start date.
rel.failure{failurecode,description}	Failure class of the work order. Related attributes.
failurecode	Failure code.
description	Failure description.
rel.location{location,description}	Location of the work order. Related attributes.
location	Location code.
description	Location description.
rel.classtructure{classificationid}	Classification of the work order. Related attribute.
rel.assignment{laborcode}	Assigned labor code of the work order. Related attribute.
rel.asset{assetnum,assettype,description,rel.classtructure{classificationid}}	Related attributes of the work order asset.
assetnum	Asset number.
assettype	Asset type.
description	Description of the asset.
rel.classtructure{classificationid}	Classification of the asset. Related attribute.
href	The href link to the work order details in Maximo. <b>Note:</b> This field is required to view work order details for the query results in Maximo Equipment Maintenance Assistant.

## Downloading work orders by using the Maximo MXAPIWODETAIL API

Use the Maximo MXAPIWODETAIL API to download work orders. You can set filter, query, and sort parameters to download specific sets of work orders.

### User name and password parameters

You must specify the Maximo user name and password with the `_lid` and `_lpwd` parameters. For example, using `_lid=id&_lpwd=password`. The user must have an administrator user account.

### Site ID parameter

By default, work orders from all sites are downloaded. You can use the `siteid` query parameter to select work orders from a specified site.

### Href parameter

The `href` field is selected by default according to the `mxe.oslc.webappurl` Maximo system property which indicates the public URI provider of Maximo. The `mxe.oslc.webappurl` property is in the following format `http://maximo_server:maximo_server_port/maximo_context_root/oslc`. To ensure that this field value is a publicly accessible URI, use one of the following options.

1. Log in Maximo work center with an administrator user account, and change value of the `mxe.oslc.webappurl` system property to be a publicly accessible URI.
2. Add `--header x-public-uri:` when calling the Maximo REST API MXAPIWODETAIL to download work orders from Maximo server. For example, use the following command.

```
curl --header "x-public-uri:http://maximo_server_port/  
maximo_server/oslc" "http://maximo_server_port/  
maximo_server/oslc/os/MXAPIWODETAIL?_lid=id&_lpwd=  
password&lean=1&ignorecollectionref=1&oslc.select=wonum,description,  
description_longdescription,status,owner,rel.failure\{failurecode,description\}  
,rel.location\{location,description\},reportdate,actstart,targstartdate,schedstart,  
rel.classtructure\{classificationid\},rel.assignment\{laborcode\},rel.asset\  
\{assetnum,assettype,description,rel.classtructure\{classificationid\}\}"
```

### Field parameters

By using the MXAPIWODETAIL API, you can use the `oslc.select` query parameter to select the required fields of a work order and to download work logs in addition to the required fields.

Use the following command to select the required fields of a work order.

```
oslc.select=wonum,description,description_longdescription,  
status,owner,rel.failure\{failurecode,description\},rel.location\{location,  
description\},reportdate,actstart,targstartdate,schedstart,  
rel.classtructure\{classificationid\},rel.assignment\{laborcode\},  
rel.asset\{assetnum,assettype,description,  
rel.classtructure\{classificationid\}\}
```

Use the following command to download work logs in addition to the required fields.

```
oslc.select=wonum,description,description_longdescription,status,owner,  
rel.failure\{failurecode,description\},rel.location\{location,description\},  
reportdate,actstart,targstartdate,schedstart,rel.classtructure\{classificationid\},  
rel.assignment\{laborcode\},rel.asset\{assetnum,assettype,description,  
rel.classtructure\{classificationid\}\},rel.worklog\{*\}
```

### Additional parameters

By using the MXAPIWODETAIL API, you can download Maximo work orders and save them to a JSON file. For example, by using the following command, you can download the 10th of 1000 work orders from Maximo server and save it to the `test.json` file.

```
curl --header "x-public-uri:http://maximo_server:maximo_server_port/  
maximo_context_root/oslc" "http://maximo_server:maximo_server_port/  
maximo_context_root/oslc/os/MXAPIWODETAIL?_lid=wilson&_lpwd=wilson&lean  
=1&ignorecollectionref=1&pageno=10&oslc.pageSize=1000&oslc.select=wonum,description,  
description_longdescription,status,owner,rel.failure\{failurecode,description\},
```

```
rel.location\{location,description\},reportdate,actstart,targstartdate,schedstart,
rel.classtructure\{classificationid\},rel.assignment\{laborcode\},rel.asset\
{assetnum,assettype,description,rel.classtructure\{classificationid\}\},
rel.worklog\{*\}" -o test.json
```

You can filter work orders by time range. In the following example, "oslc.where=reportdate>="date" and reportdate<"date"" are used to filter work orders in a certain date range. This example downloads the first 100 work orders that are reported during the time period from "2009-01-06T00:00:00.000" to "2018-02-06T00:00:00.000".

```
curl --header "x-public-uri:http://maximo_server:maximo_server_port/
maximo_context_root/oslc" "http://maximo_server:maximo_server_port/
maximo_context_root/oslc/os/MXAPIWODETAIL?_lid=wilson&_lpwd=wilson&lean=1
&ignorecollectionref=1&pageno=1&oslc.pageSize=100&oslc.where=reportdate>=
"2009-01-06T00:00:00.000"&%20and%20reportdate<"2018-02-06T00:00:00.000"
&oslc.select=wonum,description,description_longdescription,status,owner,
rel.failure\{failurecode,description\},
rel.location\{location,description\},reportdate,actstart,targstartdate,schedstart,
rel.classtructure\{classificationid\},rel.assignment\{laborcode\},rel.asset\
{assetnum,assettype,description,rel.classtructure\{classificationid\}\},
rel.worklog\{*\}" -o test.json
```

You can also filter work orders by time range by using the **tlrange** and **tlattribute** parameters to filter work orders with time range based query. For example, you can find work orders that are reported in the past 3 months with the query parameter **tlrange=-3M&tlattribute=reportdate**. And using the query parameter **tlrange=+3Y&tlattribute=schedstart**, you can filter work orders that are scheduled to start 3 years in the past and 3 years in the future indexed on the current date. For example, using the following command, you can download the first 100 work orders that are reported in the past 3 months.

```
curl --header "x-public-uri:http://maximo_server:maximo_server_port/
maximo_context_root/oslc" "http://maximo_server:maximo_server_port/
maximo_context_root/oslc/os/MXAPIWODETAIL?_lid=wilson&_lpwd=
wilson&lean=1&ignorecollectionref=1&pageno=1&oslc.pageSize=100&tlrange=-3M&tlattribute
=reportdate&oslc.select=wonum,description,description_longdescription,status,owner,
rel.failure\{failurecode,description\},rel.location\{location,description\},reportdate,
actstart,targstartdate,schedstart,rel.classtructure\{classificationid\},rel.assignment
\{laborcode\},rel.asset\{assetnum,assettype,description,rel.classtructure\
{classificationid\}\},rel.worklog\{*\}" -o test.json
```

Using the following command, you can download the first 1000 work orders that are scheduled to start between the last 3 years and the next 3 years from the current date. Note that you must use **%2B** instead of **+** in the curl request.

```
curl --header "x-public-uri:http://maximo_server:maximo_server_port/
maximo_context_root/oslc" "http://maximo_server:maximo_server_port/
maximo_context_root/oslc/os/MXAPIWODETAIL?_lid=wilson&_lpwd=
wilson&lean=1&ignorecollectionref=1&pageno=1&oslc.pageSize=1000&tlrange=%2B-
3Y&tlattribute=schedstart&oslc.select=wonum,description,description_longdescription,
status,owner,rel.failure\{failurecode,description\},rel.location\{location,description\},
reportdate,actstart,targstartdate,schedstart,rel.classtructure\{classificationid\},
rel.assignment\{laborcode\},rel.asset\{assetnum,assettype,description,rel.classtructure\
{classificationid\}\},rel.worklog\{*\}" -o test.json
```

To download a specific work order, you can add the **oslc.where=wonum=workorder\_number** parameter to get the work order. For example, using the following command, you can download work order 43041.

```
curl --header "x-public-uri:http://maximo_server:maximo_server_port/
maximo_context_root/oslc" "http://maximo_server:maximo_server_port/
maximo_context_root/oslc/os/MXAPIWODETAIL?_lid=wilson&_lpwd=wilson&lean=1&
ignorecollectionref=1&oslc.where=wonum=43041&oslc.select=wonum,description,
description_longdescription,status,owner,rel.failure\{failurecode,description\},
rel.location\{location,description\},reportdate,actstart,targstartdate,schedstart,
rel.classtructure\{classificationid\},rel.assignment\{laborcode\},rel.asset\
{assetnum,assettype,description,rel.classtructure\{classificationid\}\},
rel.worklog\{*\}" -o test.json
```



To download a set of work orders, you can add the **oslc.where=wonum in [, , ...]** parameter. For example, using the following command, you can download work orders 1158, 1161, and T1056.

```
curl --header "x-public-uri:http://maximo_server:maximo_server_port/  
maximo_context_root/oslc" "http://maximo_server:  
maximo_server_port/maximo_context_root/oslc/os/MXAPIWODETAIL?  
_lid=wilson&_lpwd=wilson&lean=1&ignorecollectionref=1&oslc.where=wonum%20in%20\  
[1158,1161,\"T1056\"]&oslc.select=wonum,description,description_longdescription,  
status,owner,rel.failure\{failurecode,description\},rel.location\{location,  
description\},reportdate,actstart,targstartdate,schedstart,rel.classtructure\  
\{classificationid\},rel.assignment\{laborcode\},rel.asset\{assetnum,assettype,  
description,rel.classtructure\{classificationid\}\},rel.worklog\{*}\}" -o test.json
```

To sort the work orders by attributes, you can add query parameter **oslc.orderBy=+attr1, -attr2**. Attributes prefixed with the plus sign (+) are sorted in ascending order, and the attributes prefixed with the minus sign (-) are sorted in descending order. Related attributes are not supported to be sorted. For example, using the following command, you can download 100 work orders that are reported most recently in the past 3 months.

```
curl --header "x-public-uri:http://maximo_server:maximo_server_port/  
maximo_context_root/oslc" "http://maximo_server:maximo_server_port/  
maximo_context_root/oslc/os/MXAPIWODETAIL?_lid=wilson&_lpwd=wilson&lean=1&  
ignorecollectionref=1&pageno=1&oslc.pageSize=100&oslc.orderBy=-reportdate&tltange=  
-3M&tlattribute=reportdate&oslc.select=wonum,description,description_longdescription,  
status,owner,rel.failure\{failurecode,description\},rel.location\{location,  
description\},reportdate,actstart,targstartdate,schedstart,rel.classtructure\  
\{classificationid\},rel.assignment\{laborcode\},rel.asset\{assetnum,assettype,  
description,rel.classtructure\{classificationid\}\},rel.worklog\{*}\}" -o test.json
```

Using the following command, you can download the first 1000 work orders after sorting the work orders in ascending order. Note that you must use **%2B** instead of **+** in the curl request.

```
curl --header "x-public-uri:http://maximo_server:maximo_server_port/  
maximo_context_root/oslc" "http://maximo_server:maximo_server_port/  
maximo_context_root/oslc/os/MXAPIWODETAIL?_lid=wilson&_lpwd=wilson&lean=  
1&ignorecollectionref=1&pageno=1&oslc.pageSize=1000&oslc.orderBy=%2Bwonum&oslc.  
select=wonum,description,description_longdescription,status,owner,rel.failure\  
\{failurecode,description\},rel.location\{location,description\},reportdate,actstart,  
targstartdate,schedstart,rel.classtructure\{classificationid\},rel.assignment\  
\{laborcode\},rel.asset\{assetnum,assettype,description,rel.classtructure\  
\{classificationid\}\},rel.worklog\{*}\}" -o test.json
```

## Uploading historical Maximo work orders

After you download Maximo work orders to a JSON file, you can upload the work orders into a Maximo work order collection. You can upload a maximum of 10,000 work orders at one time.

### Procedure

1. In Maximo Equipment Maintenance Assistant Studio, launch the Document Manager.
2. Select the Maximo work order collection.
3. Select **Add New** and choose the JSON file that contains the downloaded work orders.
4. Input the information and then select **Upload**.
5. You can add new work orders or update the existing work orders by selecting **Add New**.

## Opening Maximo Equipment Maintenance Assistant from Maximo Asset Management

---

After integration is enabled, you can open Maximo Equipment Maintenance Assistant from Maximo Asset Management by clicking the **Seek Advice** button in a work order.

### Procedure

1. In Maximo Asset Management, open a work order.
2. Click **Seek Advice**.
3. Select work order information from the dropdown menu or input search text.
4. Filter the collection.
5. Click **Search**.
6. You can filter the work order collection by asset or time range.
7. In case of document collections, relevant documents are shown. Click **Read All** to see all documents in a new browser window or click the file name to download the document.

## Automatically refreshing Maximo work orders

---

You can integrate Maximo Equipment Maintenance Assistant with Maximo so that when a work order is updated in Maximo, it is updated automatically in Maximo Equipment Maintenance Assistant.

### Using the Maximo Integration Framework

Use Maximo Integration Framework to configure Maximo Equipment Maintenance Assistant so that work orders automatically refresh when they are updated in Maximo.

#### Creating a Java Message Service queue

Before Maximo work orders can refresh automatically in Maximo Equipment Maintenance Assistant, you must create a Java™ Message Service (JMS) queue.

#### About this task

For more information about setting up a JMS queue, see: [Configuring Java Message Service for WebSphere® Application Server](#) and [Configuring the Java Message Service for WebLogic Server](#) topics.

#### Adding a template to the MXAPIWODetail object structure

Before Maximo work orders can refresh automatically in Maximo Equipment Maintenance Assistant, you must add an Maximo Equipment Maintenance Assistant template to Maximo.

### Procedure

1. In Maximo, go to the **Object Structures** application.
2. Search for the MXAPIWODETAIL object structure, and select **Query Template**.
3. In the **Query Templates for MXAPIWODETAIL** box, input the following information:
  - a) In the Template name field, input *EMA*.
  - b) In the Page Size field, set the page size to *1*.
4. In the **Query Template Attributes for EMA** box, add the following attributes:
  - actstart
  - description
  - description\_longdescription
  - owner

- rel\$asset.assetnum\*
- rel\$asset.assettype\*
- rel\$asset.description\*
- rel\$asset.rel\$classtructure.classificationid\*
- rel\$assignment.laborcode\*
- rel\$classtructure.classificationid\*
- rel\$failure.description\*
- rel\$failure.failurecode\*
- rel\$location.description\*
- rel\$location.location\*
- reportdate
- schedstart
- status
- targstartdate
- wonum

### Creating a publish channel

Before Maximo work orders can refresh automatically in Maximo Equipment Maintenance Assistant, you must create a publish channel.

#### Procedure

1. In Maximo, go to the **Publish Channel** application.
2. Select **New Publish Channel** and input the following information:
  - a) In the Name field, input **MXINTEMA**.
  - b) In the Object Structure field, input **MXAPIWODETAIL**.
  - c) In the template name field, input **EMA**.
  - d) Select **Publish JSON?**
  - e) Select **Enable Listener?**
3. Save the publish channel.

### Adding an end point

Before Maximo work orders can refresh automatically in Maximo Equipment Maintenance Assistant, you must add an end point.

#### Procedure

1. In Maximo, go to the **End Point** application.
2. Select **New End Point** and input the following information:
  - a) In the Name field, input **EMA**.
  - b) In the Handler field, input **HTTP**.
  - c) In the Headers field, input X-TENANT-ID:*tenantId*X-API-KEY:*API Key for document management*Content-Type:*application/json*.
  - d) In the HTTPMETHOD field, input **POST**.
  - e) In the URL field, input `https://ea-prod-cluster.us-south.containers.appdomain.cloud/api/v1/document-management/collections/collectionId/workorders/json`.

### Adding an external system

Before Maximo work orders can refresh automatically in Maximo Equipment Maintenance Assistant, you must add a new external system.

#### Procedure

1. In Maximo, go to the **External Systems** application.
2. Select **New External System** and input the following information:
  - a) In the Name field, input **EMA**.
  - b) In the End Point field, input **EMA**.
  - c) In the Outbound Sequential Queue field, input `jms/maximo/int/queues/ema`.
  - d) Select **Enabled?**.
3. On the **Publish Channels** tab, select the following information:
  - a) Select **Channel**.
  - b) Select **MXINTEMA**.
  - c) Select **Enabled?**.
4. Save the external system.

### Adding a cron task instance

Before Maximo work orders can refresh automatically in Maximo Equipment Maintenance Assistant, you must add a cron task instance.

#### Procedure

1. In Maximo, go to the **Cron Task Setup** application.
2. Filter and select the JMSQSEQCONSUMER cron task.
3. In the **Cron Task Instances** box, select **New Row** and input the following information:
  - a) In the Cron Task Instance Name field, input **EMAQ**.
  - b) In the Schedule field, set the schedule or interval time. For example, `1h,*0,*,*,*,*,*`.
  - c) In the Run as User field, input the username.
  - d) Select **Active**.
4. Set the following parameters:
  - MESSAGEPROCESS : `psdi.iface.jms.QueueToDestManagerProcessor`
  - QUEUENAME: `"jms/maximo/int/queues/ema"` or `"jms/maximo/int/queues/sqout"`
5. Select **Save Cron Task Definition**.
6. Select **Reload Request**.

---

## Chapter 8. Application programming interface

The application programming interface enables you to manage users, get usage information, manage knowledge graphs, manage documents, manage document queries, and manage diagnosis.

To use an API, you need to generate an API key from the administration console.

The [user management API](#) enables you to perform tasks such as managing user authorization, roles, and API keys.

The [usage API](#) enables you to perform tasks such as getting collection usage information, document usage information, and document query usage information.

The [document management API](#) enables you perform tasks such as listing, creating, and deleting document collections and uploading and downloading documents.

The [document query API](#) enables you to perform tasks such as listing all document collections, getting detailed information about a collection, and querying documents in multiple collections.

The [diagnosis API version 1.1](#) enables you to perform tasks for diagnosis model management such as listing, creating, updating, deleting diagnosis templates, and managing diagnosis model instances. The diagnosis API also enables you to perform tasks for diagnosis data management such as listing, creating, updating, and deleting diagnosis data, and triggering diagnosis. If you have models that use the [1.0](#) version of the diagnosis API, you can upgrade the models to use the latest API version.



## Notices

---

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

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

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

*IBM Director of Licensing  
IBM Corporation  
North Castle Drive, MD-NC119  
Armonk, NY 10504-1785  
US*

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

*Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan Ltd.  
19-21, Nihonbashi-Hakozakicho, Chuo-ku  
Tokyo 103-8510, Japan*

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

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

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

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

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

*IBM Director of Licensing  
IBM Corporation  
North Castle Drive, MD-NC119  
Armonk, NY 10504-1785  
US*

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

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

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

#### COPYRIGHT LICENSE:

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

## Trademarks

---

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

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

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

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

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

## Terms and conditions for product documentation

---

Permissions for the use of these publications are granted subject to the following terms and conditions.

### **Applicability**

These terms and conditions are in addition to any terms of use for the IBM website.



## Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

## Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

## Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

## IBM Online Privacy Statement

---

IBM Software products, including software as service solutions, ("Software Offerings") may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering's use of cookies is set forth below.

Depending upon the configurations deployed, this Software Offering may use session and persistent cookies that collect each user's name, user name, password, or other personally identifiable information for purposes of session management, authentication, single sign-on configuration or other usage tracking or functional purposes. These cookies can be disabled, but disabling them will also likely eliminate the functionality they enable.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM's [Privacy Policy](http://www.ibm.com/privacy) at <http://www.ibm.com/privacy> and IBM's [Online Privacy Statement](http://www.ibm.com/privacy/details) at <http://www.ibm.com/privacy/details> in the section entitled "Cookies, Web Beacons and Other Technologies" and the "[IBM Software Products and Software-as-a-Service Privacy Statement](https://www.ibm.com/privacy)" at <https://www.ibm.com/privacy>.







Part Number:

(1P) P/N: