



# API Guide

Generated on: 2023-12-07 12:26:43 GMT+0000

SAP Cloud for Utilities Foundation | Cloud

## CONFIDENTIAL

Original content: [https://help.sap.com/docs/Cloud\\_for\\_Uilities\\_Foundation/964850841e2943659444c83c00f05dcc?locale=en-US&state=PRODUCTION&version=Cloud](https://help.sap.com/docs/Cloud_for_Uilities_Foundation/964850841e2943659444c83c00f05dcc?locale=en-US&state=PRODUCTION&version=Cloud)

## Warning

This document has been generated from the SAP Help Portal and is an incomplete version of the official SAP product documentation. The information included in custom documentation may not reflect the arrangement of topics in the SAP Help Portal, and may be missing important aspects and/or correlations to other topics. For this reason, it is not for productive use.

For more information, please visit the <https://help.sap.com/docs/disclaimer>.

# API Call Choreography for the Use Case Measurement Concept Instance Shutdown

The choreography described here should be treated as an example only regarding the sequence in which the available endpoints can be called, for example, to shut down a measurement concept instance.

## Start a Process to Shut Down a Measurement Concept Instance

You can use the `MCMService.initShutdown` OData function to trigger a shutdown process for a measurement concept instance. You must call the `/MCMInstance({InstanceUUID})` endpoint to address the version of the measurement concept instance for which the shutdown is to be performed.

The POST request to invoke the `initShutdown()` OData function addresses an existing measurement concept instance version that is identified by its UUID and stores it as the predecessor version in the data of the newly built version of the measurement concept instance.

You can start a shutdown process for a measurement concept instance that satisfies the following conditions:

- The version of the measurement concept instance has the status `INSTALLED` or `ACTIVE`
- No other version of the measurement concept instance has the status `INITIAL`, `IN_CHANGE`, or `IN_SHUTDOWN`

Parts of the `initShutdown()` request body are as follows:

- `changeProcesses`: Mandatory and used to identify the party who invoked the request
- `processReason_code`: Mandatory and used to indicate the reason for which the shutdown process is initiated
- `customerRequestDate`: Mandatory and used to indicate the date at which the shutdown process is initiated

### Sample Code

```
POST http://localhost:8080/odata/v4/api/mcm/v1/MCMInstances({instanceId})/MCMService.initShutdown
{
  "dataForNewInstanceVersion": {
    "changeProcesses": [{
      "externalOrderId": "111",
      "processReason_code": "GRID_HANDBOVER",
      "processData": {
        "customerRequestDate": "2022-08-01"
      }
    }]
  }
}
```

The measurement concept management component processes this request by:

- Checking whether the addressed measurement concept instance has the measurement concept instance status `ACTIVE` or `INSTALLED`. Starting a shutdown process for a measurement concept instance that has any other status is prohibited.
- Creating a new version of the measurement concept instance. This new version is a document in `MCMInstances` entity and has its own UUID as a key, a different value for the `version` parameter, but the same value for the `idText` parameter as that of the addressed version of the measurement concept instance.

- Setting the addressed version of the measurement concept instance as the predecessor of the new version
- Building up the structure of the new version by copying the metering locations, market locations, actors, and their sub-entities from the given measurement concept model
- Copying existing domain data (not process data) from the predecessor version to the new version. This is done based on a comparison of the parts of the measurement concept instance. The aim is to copy data only for elements that are the same between the predecessor version and the new version.

The response to the `initShutdown()` OData function contains a structure showing the difference between the previous version of the measurement concept instance and the new version.

## Process Shutdown - Step by Step

The process to shut down a measurement concept instance consists of steps, similar to the Create process, each of them having the same pattern of:

1. Setting data
2. Confirming data is complete
3. Status transfer to next Process status
4. Raising events to inform registered parties

Similar to the Create process, for the shutdown process you set data for the new instance version in the body of the request message.

### 1. [Set Notification That Final Data Entry Is Ready](#)

Having started a process to shutdown a measurement concept instance, you can continue the process on the measurement concept management component. The pre-system (for example, the DSO) notifies the measurement concept management component using the `MCMServices.notifyFinalDataEntryReady` OData action.

### 2. [Set Notification That a Single Device Is Removed](#)

When the measurement concept instance reaches the `IN_SHUTDOWN-STARTED` state, for each metering location, the downstream system sends the removal dates for physical devices and notifies that the devices have been removed. If there are no metering locations (for example, flat rate model), the downstream system instead notifies that all devices have been removed for the measurement concept instance.

### 3. [Set Notification That a Single Market Location Is Removed](#)

When the measurement concept instance reaches the `IN_SHUTDOWN-DEVICES_SHUTDOWN` state, for each market location and each actor (of types `CONSUMER`, `PRODUCER`, or `STORAGE`), the downstream system sends the removal dates for the market location and a notification that the market location and actors have been removed.

## Set Notification That Final Data Entry Is Ready

Having started a process to shutdown a measurement concept instance, you can continue the process on the measurement concept management component. The pre-system (for example, the DSO) notifies the measurement concept management component using the `MCMServices.notifyFinalDataEntryReady` OData action.

### Sample Code

```
POST http://localhost:8080/odata/v4/api/mcm/v1/$batch
--batch_id-1
Content-Type:application/http
Content-Transfer-Encoding:binary
```

12/7/2023

```
POST MCMInstances({{instanceInChangeId}})/changeProcesses({{instChange-ProcessId-1}})/processData:
Accept:application/json;odata.metadata=minimal;IEEE754Compatible=true
Content-Type:application/json;charset=UTF-8;IEEE754Compatible=true

{}

--batch_id-1--
```

As a result, the shutdown process is initiated, and the measurement concept management component notifies event subscribers on SAP Event Mesh with the `MCMInstance.ShutdownRequested.v1` event.

The measurement concept instance transitions to the `IN_SHUTDOWN-STARTED` state.

**Parent topic:** [Process Shutdown - Step by Step](#)

**Next:** [Set Notification That a Single Device Is Removed](#)

## Related Information

[List of Process Status Transitions](#)

# Set Notification That a Single Device Is Removed

When the measurement concept instance reaches the `IN_SHUTDOWN-STARTED` state, for each metering location, the downstream system sends the removal dates for physical devices and notifies that the devices have been removed. If there are no metering locations (for example, flat rate model), the downstream system instead notifies that all devices have been removed for the measurement concept instance.

For each metering location, SAP S/4HANA Utilities performs the following:

- Sends a date for the `removalDate` parameter
- Calls the `notifySingleDeviceRemoved()` OData action to set the `MCIMeteringLocation.locationRemoved` flag to true
- Calls the `notifyAllDevicesRemoved()` OData action to set the `MCIInstance.allDevicesRemoved` flag to true in case of no metering locations.

### Sample Code

```
POST http://localhost:8080/odata/v4/api/mcm/v1/$batch
--batch_id-1
Content-Type:application/http
Content-Transfer-Encoding:binary

PATCH MCMInstances({{instanceInChangeId}})/meteringLocations({{instCP-meloId-1}}) HTTP/1.1
Accept:application/json;odata.metadata=minimal;IEEE754Compatible=true
Content-Type:application/json;charset=UTF-8;IEEE754Compatible=true

{
  "removalDate": "2022-08-25"
}

--batch_id-1
Content-Type:application/http
Content-Transfer-Encoding:binary

POST MCMInstances({{instanceInChangeId}})/meteringLocations({{instCP-meloId-1}})/MCMService.notifi
```

This is custom documentation. For more information, please visit the [SAP Help Portal](#)

12/7/2023

```
Accept:application/json;odata.metadata=minimal;IEEE754Compatible=true
Content-Type:application/json;charset=UTF-8;IEEE754Compatible=true
```

```
{}
```

```
--batch_id-1--
```

The measurement concept management component considers the following validations before shutting down devices:

- The date that is provided for the `MCIMeteringLocations.removalDate` parameter is earlier than or the same as the date at which the shutdown is requested.
- For the most recent version of the measurement concept instance that has existing values for the `installedOn` and `commercialSetupOn` parameters, the measurement concept management component considers the following validations:
  - The date of the `MCIMeteringLocations.removalDate` parameter is later than or the same as the date of the `MeasurementConceptInstances.installedOn` parameter.
  - If the date of the `MeasurementConceptInstances.commercialSetupOn` parameter is earlier than or the same as the date at which the shutdown is requested, then the date of the `MCIMarketLocations.removalDate` parameter is later than or the same as the date of the `MeasurementConceptInstances.commercialSetupOn` parameter.

In any other case, the date of the `MCIMarketLocations.removalDate` parameter is either later than the date of the `MeasurementConceptInstances.commercialSetupOn` parameter or earlier than or the same as the date of the `MeasurementConceptInstances.commercialSetupOn` and later than or the same as the date of the `MeasurementConceptInstances.installedOn` parameter.

- All metering locations have the `MCIMeteringLocation.locationRemoved` flag set to true.

## i Note

It's not possible to validate against the `MCIMeteringLocationsPD.installationDate` property.

If there are no issues with the input validations, then as a result of removing devices, the measurement concept management component notifies event subscribers on SAP Event Mesh with the `MCMDevicesShutdownV1` event. The state of the measurement concept instance transitions to `IN_SHUTDOWN-DEVICES_SHUTDOWN`.

**Parent topic:** [Process Shutdown - Step by Step](#)

**Previous:** [Set Notification That Final Data Entry Is Ready](#)

**Next:** [Set Notification That a Single Market Location Is Removed](#)

## Related Information

[List of Process Status Transitions](#)

# Set Notification That a Single Market Location Is Removed

When the measurement concept instance reaches the `IN_SHUTDOWN-DEVICES_SHUTDOWN` state, for each market location and each actor (of types `CONSUMER`, `PRODUCER`, or `STORAGE`), the downstream system sends the removal dates for the market location and a notification that the market location and actors have been removed.

For each market location, SAP S/4HANA Utilities performs the following:

This is custom documentation. For more information, please visit the [SAP Help Portal](#)

- Sends a date for the `removalDate` parameter
- Calls the `notifySingleMarketLocationRemoved()` OData action to set the `MCIMarketLocations.locationRemoved` flag to `true`
- For each actor, SAP S/4HANA Utilities calls the `notifySingleActorRemoved()` OData action to set the `Actors.actorRemoved` flag to `true`.

### Sample Code

```
POST http://localhost:8080/odata/v4/api/mcm/v1/$batch
--batch_id-1
Content-Type:application/http
Content-Transfer-Encoding:binary

PATCH MCMInstances({{instanceInChangeId}})/marketLocations({{instCP-maloId-1}}) HTTP/1.1
Accept:application/json;odata.metadata=minimal;IEEE754Compatible=true
Content-Type:application/json;charset=UTF-8;IEEE754Compatible=true

{
  "removalDate": "2022-09-02"
}

--batch_id-1
Content-Type:application/http
Content-Transfer-Encoding:binary

POST MCMInstances({{instanceInChangeId}})/marketLocations({{instCP-maloId-1}})/MCMService.notify!
Accept:application/json;odata.metadata=minimal;IEEE754Compatible=true
Content-Type:application/json;charset=UTF-8;IEEE754Compatible=true

{}

--batch_id-1--
```

The measurement concept management component considers the following validations before finalizing the shutdown of a measurement concept instance:

- A date is provided for the `MCIMarketLocations.removalDate` parameter.
- All metering locations of the measurement concept instance have the `commercialShutdownOn` flag set to `true`.
- All market locations have the `locationRemoved` flag set to `true`.
- The date of the `commercialShutdownOn` parameter is equal to the most recent removal date among all market locations.

#### **i Note**

The date of the `commercialShutdownOn` parameter is later than or the same as the most recent removal date of all metering locations.

- The date of the `installedUntil` parameter of the current and predecessor versions of the measurement concept instance is set to the same date as that of the most recent removal date among all metering locations, if the instance has metering locations. If there are no metering locations (for eg., flat rate model), the `commercialShutdownOn` date is used.

#### **i Note**

The date of the `installedUntil` parameter has to be later than or the same as the date of the `installedOn` property.

- When shutting down a measurement concept instance that has a pending creation process, if the status of the predecessor version of the measurement concept instance is still in the `INSTALLED-DEVICES_CREATED` state, then the date of the `installedOn` parameters of the current and predecessor versions are set to the same date as the `installationDate` parameter that is the most recent out of all the metering locations of the predecessor version of the measurement concept instance.
- When shutting down a measurement concept instance that has a pending change process, if there is at least one version that is still in the `INSTALLED-DEVICES_CONFIRMED` state, then beginning with the current version, the date of the `installedOn` parameter is set as follows:
  - If a date exists for the `MCIMeteringLocations.installationDate` parameter and if it is later than or the same date as that of the `MCMInstances.installedOn` parameter of the predecessor version of the measurement concept instance, then the date of the `MCMInstances.installedOn` parameter is set to the same date as that which is most recent among the `installationDate` parameters of all versions of the measurement concept instance.
  - In any other case, the date of the `installedOn` parameter is the same as that date of the predecessor version. The date of the `installedUntil` parameter of the predecessor version is set to the same date as the `installedOn` parameter of the predecessor version minus one day. The date of the `installedUntil` parameter is the same as or later than the date of the `installedOn` parameter.

### i Note

If all versions of the measurement concept instance are in the `ACTIVE_FINALIZED` or `INSTALLED-LOCATIONS_CONFIRMED` states, then a date for the `installedOn` parameter is already set for those versions. Except for the most recent predecessor version of the measurement concept instance, the date of the `installedUntil` parameter is already set, and no further action is required.

### i Note

It's not possible to validate against the `MCIMarketLocations.commercialSetupDate` property.

If there are no issues with the input validations, then the following results from removing market locations:

- The measurement concept management component notifies event subscribers on SAP Event Mesh with the `MCMShutdownFinalizedV1` event.
- The measurement concept instance transitions to the `SHUTDOWN-FINALIZED` state.
- All the predecessor versions of the measurement concept instance transition to the `HISTORIC_DEACTIVATED` state.

Parent topic: [Process Shutdown - Step by Step](#)

Previous: [Set Notification That a Single Device Is Removed](#)

## Related Information

[List of Process Status Transitions](#)