

**Oracle Utilities Smart Grid Gateway
Adapter for Itron OpenWay**

Administrative User Guide

Release 2.2.0.2

E92031-01

January 2018

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners. Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Itron OpenWay Adapter Overview	4
Itron OpenWay Adapter Processing	6
Initial Measurement Data and Device Event Loading.....	6
Scheduled Read Commands.....	6
Initial Measurements.....	9
Device Events.....	11
Base Package Business Objects.....	13
Device Communication.....	14
Communication Flows.....	14
Device Communication Base Package Business Objects.....	14
External System.....	16
Outbound Message Types.....	16
Inbound / Outbound Service Configuration.....	16
BPEL Processes.....	17
Configuring an Itron OpenWay Head-End System	20
Master Configurations.....	20
Inbound Web Services.....	20
Message Senders.....	21
Outbound Message Types.....	22
External System.....	23
Service Provider.....	23
Processing Methods.....	24
Configuring Endpoint URIs.....	25
Configuring Itron OpenWay Extendable Lookups	27
Extending the Itron OpenWay Adapter	30
The Itron OpenWay Test Harness	31
Test Harness Design.....	31
Locating the WSDL for the Test Harness.....	32
Web Services.....	32
General Services.....	32
Locate Meter Services.....	33
Meter Administration Services.....	35
Meter Attribute Administration Services.....	38

Chapter 1

Itron OpenWay Adapter Overview

The Oracle Utilities Smart Grid Gateway Adapter for Itron OpenWay supports communication with the Itron OpenWay application, including measurement data and device event loading, and command messaging in support of commissioning, connect, disconnect, decommissioning, status check, and on-demand read commands. The following table describes the attributes of the adapter:

Attribute	Details
Currently Supported Versions	6.6 Itron OpenWay Operations Center SR 4.1
Protocol	Proprietary
Market(s)	Worldwide
Architecture	RF Mesh

The adapter uses Oracle Service Bus (OSB) and Oracle Business Process Execution Language (BPEL) to facilitate communication between Oracle Utilities Smart Grid Gateway and the Itron OpenWay application.

The following functionality is included:

Measurement Data and Device Event Loading - data parsing and transformation via Oracle Service Bus from Itron OpenWay format into the Oracle Utilities Service and Measurement Data Foundation unified format for measurement data and device events.

Schedule Read Commands - Measurement data and device events are not automatically sent to Oracle Utilities Smart Grid Gateway, and must be requested from the Itron head-end system via Scheduled Read commands. Upon receipt of a Scheduled Read command, the Itron OpenWay application deposits a usage and/or device event payload so that Oracle Service Bus can pick up and process the file. See [Scheduled Read Commands](#) for more information about Scheduled Read commands.

Measurement Data and Device Event Processing - configurable mapping for Itron OpenWay status codes and device event names to Oracle Utilities Service and Measurement Data Foundation standard values.

Smart Meter Command Processing - sending/receiving messages to/from the Itron OpenWay application to initiate smart meter commands from Oracle Utilities Smart Grid Gateway. The Itron OpenWay adapter supports the following types of commands and communications:

- **Meter Commissioning** - business objects and BPEL processes to support issuing meter commissioning commands.
- **Remote Connect** - business objects and BPEL processes to support issuing remote connect commands.

- **Remote Disconnect** - business objects and BPEL processes to support issuing remote disconnect commands
- **Meter Decommissioning** - business objects and BPEL processes to support issuing meter decommissioning commands.
- **On-Demand Read** - business objects and BPEL processes to support issuing on-demand read commands.
- **Device Status Check** - business objects and BPEL processes to support issuing device status check commands for both single and multiple devices.

Chapter 2

Itron OpenWay Adapter Processing

This section provides details concerning the OSB processing, BPEL processes, and OUAF objects supplied as part of the base package for the adapter for Itron OpenWay. This information illustrates how the base package objects were designed, and can serve as the basis for any customizations you create as part of your implementation.

Initial Measurement Data and Device Event Loading

Initial measurement data and device event loading and processing use OSB to poll for, parse, and transform the head-payloads into the Oracle Utilities Smart Grid Gateway service format. Payloads contain measurements and meter events in some head-end specific format OSB then places each service call into a JMS queue within the Oracle Utilities applications. The JMS client consumes the entries and invokes the respective services in parallel then a service creates initial measurements with data in a common format with head-end-specific processing as needed. A second service creates device events with data in a common format.

As noted above, measurement and device event data must be requested from the Itron OpenWay head-end system via Scheduled Read commands. See [Scheduled Read Commands](#) for more information about Scheduled Read commands.

Scheduled Read Commands

As noted above, measurement and device event data must be requested from the Itron OpenWay head-end system via Scheduled Read commands.

Schedule read commands are used to request meter reads and/or device events for a group of devices for a specified date and time. Scheduled read commands can be created such that recurring requests based on the same command parameters are sent to the head-end system at regular intervals. For example, a command could be configured to request daily (24 hour) scalar meter readings once per day for a specified group of devices. Another command could be configured to request weekly (168 hours) device events once per week for a group of devices.

Scheduled read commands should be created for all devices from which meter readings and/or device events are to be uploaded via Oracle Utilities Smart Grid Gateway. Separate scheduled read commands can be created for specific groups and for specific types of data. For example, for a given group of scalar devices, you might create separate commands to request scalar usage data and to request device events from the same group (both of which can be based on different

recurrence patterns). The specifics concerning request specifics and recurrence patterns should be based on the requirements of the implementation.

Attributes used to define a schedule read command include the following:

- **Status:** the status of the activity
- **Schedule Read Type:** the type of scheduled read
- **Schedule Information:** information about the schedule for the read, including:
 - **One Time Read:** indicates if the request is a one-time request. One-time requests have a defined date/time range during which request and recurrences can take place.
 - **One Time Start Date Time:** specifies the start time for one-time requests (applicable only if **One Time Read** is set to “Yes”)
 - **One Time End Date Time:** specifies the start time for one-time requests (applicable only if **One Time Read** is set to “Yes”)
 - **First Daily Measurement Time:** Indicates starting time for measurements for each day. This field, along with the value from **IMD Length** , will be used to determine when interrogation requests should be made. For example, if the **First Daily Measurement Time** is set to 08:00AM and **IMD Length** is set to 12 hours (12:00:00) then there would be two requests made per day:
 - 8:00AM - 8:00PM
 - 8:00PM - 8:00AM
 - **Interrogation Buffer:** This parameter impacts when interrogation requests are sent. It is entered in the format of hours:minutes:seconds. For example, if set to 1 hour (01:00:00) for our above example, the first request of the day for 8:00AM - 8:00PM would be sent to Itron OpenWay at 7:00PM. The buffer time will decrease if the activity is monitored after 7:00PM, so if the monitor process is executed at 7:30PM the buffer time would only be 30 minutes and if the monitor process were not executed until after 8:00PM there would be no buffer time at all.
 - **IMD Length:** Defines the number of hours, minutes and seconds of data that should be retrieved for each request. It is entered in the format of hours:minutes:seconds. This number must be divisible into 24 since it will determine how many requests will be made per day (i.e. if set to 8 hours (08:00:00) there will be 3 requests per day).
 - **Disable Extended IMD Length Recovery:** By default the schedule read activity will attempt to "catch-up" when interrogation requests have been missed by requesting the entire period from the last interrogation request made. Setting this parameter to "Yes" will turn off this functionality and interrogation requests will only be made in for the exact amount of time defined in the **IMD Length**
 - **Range of Recurrence:** indicates how to define request recurrences. Valid values include “Maximum Recurrence”, “No End Date”, and “Recurrence End Date”
 - **Maximum Recurrence:** The activity will make a set number of requests, once the set number has been reached the activity will expire. When selected, **Maximum Recurrences** must also be provided.
 - **No End Date:** The activity will never expire and will continue to make requests until it is manually transitioned to inactive.
 - **Recurrence End Date:** The activity will continue to make requests until a specified date. When selected, **Recurrence End Date Time** must also be provided.

NOTE: When “No End Data” is selected, the recurrences of the request continue until a user manually changes the status of the command.

- **Maximum Recurrence:** defines the maximum number of recurrences before the command’s status is changed to “Inactive”.
- **Recurrence End Date Time:** The date after which no more recurrences are attempted. After this date and time, the command’s status is changed to “Inactive”.

- **Group:** the group of devices used by requests for the command
- **Group Type:** indicates the type of device group to use for the command. Valid options include Application Group or Configuration Group.
- **Application Group/Configuration Group Name:** defines the specific group (based on the group type).

NOTE: Groups referenced by Scheduled Read commands and the devices that belong to each are defined in Itron OpenWay head-end system. Each group to be used with these commands must also be defined in either the “Itron — Application Group Lookup” or “Asset Specification” extendable lookup. See the Oracle Utilities Smart Grid Gateway Adapter for Itron OpenWay Configuration Guide for more information about these extendable lookups.

- **Default Read Parameters:** specifies details concerning the data to be requested. The read parameters define the exact data that will be requested from the meters. Please consult the Itron OpenWay documentation for the specifics of each parameter.
 - **Interrogation Window Hours:** the number of hours used to calculate when to submit requests. For example, to submit requests once per day, this should be set to 24. To submit requests once per week (7 days), this should be set to 168. The current date time (or the Latest Request End Date Time) plus this equals interrogation window end date time.
 - **Override Read Parameters:** an alternative set of parameters that can be defined. To leverage this group each parameter must be provided, no parameter is allowed to be left blank. This set of parameters will be used when the request end time is equal to the **First Daily Measurement Time**. For example, a schedule read with a **First Daily Measurement Time** of 12:00AM and an **IMD Length** 12 hours (12:00:00) would make the following requests each day:
 - 12:00AM - 12:00PM
 - 12:00PM - 12:00AM

The **Override Read Parameters** would be used for the 2nd request where the request end time of 12:00AM matches the **First Daily Measurement Time** of 12:00AM. If an extended request is made to "catch up" (i.e. it makes a request that is larger in duration than the configured **IMD Length**), and the request start and request end date time includes the **First Daily Measurement Time** then the **Override Read Parameters** will be used.

- **Outbound Communication Exception Handling Overrides:** parameters used to control retry attempts and expiration for the outbound communication. These parameters take precedence over their outbound communication counterparts
- **Last Read Details:** details concerning the most recent read request sent for the command. This information is used to calculate the time for the next request (based on the **Interrogation Window Hours** parameter).
 - **Latest Request Start Date Time:** the start date and time of the last request
 - **Latest Request End Date Time:** the end date and time of the last request
 - **Recurrence Count:** the number of recurrences of the last request

About Ad Hoc Requests

While scheduled read commands are typically configured to create recurring requests for meter reads and device events, users can also submit ad hoc requests for a specific date and time range based on the parameters of a currently active scheduled read command. This is useful if meter read data or device events are needed before the next request for a command would be submitted based on the command’s recurrence settings.

Ad hoc requests are created as separate one-time request commands, with a start and end time based on parameters supplied by the user when initiating the ad hoc request. The group and read parameters for ad hoc requests are the same as those for the command used to send the ad hoc request.

Error Handling

If errors occur when processing scheduled read commands, errors are logged as follows:

- If there is a single error it will create a single log entry
- If there is more than one error, the log entry will perform a count of devices and create a single error message stating the number of devices that were in error.

Initial Measurements

The usage data received from the AMI head-end system as a file in Itron OpenWay XML format is loaded into Oracle Utilities as initial measurement data. The following OSB projects, delivered in the base product, help manage the usage processing:

1. **SGG-D8-ITRONXML-BASE** contains components responsible for “actual” processing of incoming data. It should not be modified during configuration. This project can be upgraded without affecting the customization and environment settings added to **SGG-D8-ITRONXML-CM**.
2. **SGG-D8-ITRONXML-CM** allows for customization and simplifies future upgrades.

The runtime configuration settings for the **SGG-D8-ITRONXML-CM** project are stored in the xquery file `EnvironmentSettings.xq`. You can use this file to adjust initial measurement data processing. For example, if you want to load raw data you would specify “true” for the content of the `populateRawIMD` element.

The following table describes the elements included in the `EnvironmentSettings.xq` file:

Element	Description	Valid Values
<code>populateRaw</code>	Determines if the initial measurement data is populated as raw data.	true false
<code>callPreProcessing</code>	Determines if the preprocessing proxy service is called.	true false
<code>callPostProcessing</code>	Determines if the postprocessing proxy service is called.	true false
<code>destinationRootElementInterval</code>	Holds the name of inbound web service for the interval IMD seeder.	
<code>destinationRootElementScalar</code>	Holds the name of inbound web service for the scalar IMD seeder. In most cases it is the same as <code>destinationRootElementInterval</code> .	
<code>destinationRootElementEvent</code>	Holds the name of inbound web service for the device event seeder.	
<code>publishServices/service</code>	Specifies the name of the business service within the OSB project used to publish data for external systems (such as Oracle DataRaker).	
<code>filterEvents</code>	Determines if events should be filtered.	true false
<code>filterUsage</code>	Determines if usage should be filtered.	true false
<code>filterRegisterSource</code>	Determines if source register data should be filtered.	true false

Publishing Initial Measurement Data

The Itron OpenWay adapter can be configured to publish initial measurement data for use in Oracle DataRaker or other external systems. This functionality is supported through a combination of OSB components and BPEL composites.

Enabling Initial Measurement Data Publishing

Publishing data is enabled by referencing a publisher business service in the `publishServices/service` element in the `EnvironmentSettings.xq` file as follows:

```
<publishServices>
  <service>[publisherBusinessService]</service>
</publishServices>
```

The following components provided with the SGG-D8-ITRONXML-CM OSB project are used in publishing measurement data to Oracle DataRaker:

- The **DataRakerBusinessService** business service is used to send data to a pre-configured JMS queue (defined as an Endpoint URI), from which the data will be published. This is the business service that should be specified in the `EnvironmentSettings.xq` file.
- The **DataRakerServiceAccount** service account is used to define and maintain the user name and password needed to access the JMS queue defined in the **DataRakerBusinessService** business service.

Configuring Initial Measurement Publishing Output

The SGGDRIntegration BPEL composite handles publishing the data to Oracle DataRaker or other systems.

Initial measurement data is published in the “native” initial measurement data format (the format of the initial measurement seeder business object). This format includes normalized unit of measure and condition codes. See the *Oracle Utilities Smart Grid Gateway Adapter Development Kit Administrative User Guide* for more details about this format.

NOTE: Initial measurement data published via this feature is published prior to VEE processing. In addition, filtering can NOT be applied to data published via this feature.

The following parameters can be used to configure details of how the data is provided to Oracle DataRaker, including the directory where files are posted for Oracle DataRaker to consume, number of records per file, polling frequency, etc. These parameters are defined during installation. See the *Oracle Utilities Smart Grid Gateway Installation Guide* for more details about defining values for these parameters.

Parameter	Description	Default Value
SGG_DR_INT_QUEUE	JNDI name of queue to publish SGG payloads. This is the JMS queue defined in the DataRakerBusinessService business service. This should NOT be changed.	DataRakerQueue
SOA_DR_PUBLISH_SIZE	The number of records (SGG payloads) to accumulate in a published file.	100
SOA_DR_FILE_SIZE	The maximum file size for the accumulated (SGG payloads) file in kilobytes.	524288
SOA_DR_ELAPSED_TIME	The period of time in second which, when exceeded, causes a new outgoing file to be created.	600
SOA_DR_POLLING_FREQ	The polling frequency in seconds of the staging directory for new files.	60
SOA_DR_STAGING_DIR	Mount point/directory for the staging directory for accumulated SGG payload files. This is used internally and should NOT be changed.	/spl/sploutput/staging

Parameter	Description	Default Value
SOA_DR_INTEGRATION_DIR	Mount point/directory from which Oracle DataRaker will consume the converted XML files.	/spl/sploutput/int

Filtering Initial Measurement Data

The Itron OpenWay adapter can be configured to filter initial measurement data passed into Oracle Utilities Smart Grid Gateway and Meter Data Management. Filtering data is enabled by setting the <filterUsage> element in the EnvironmentSettings.xq file to “true” as follows:

```
<filterUsage>true</filterUsage>
```

When filtering is enabled, only measurements whose <externalUOM> matches one of the values defined in the **Itron - UOM Code to Standard UOM Mapping** extendable lookup (D8-HeadendUOMLookup) are passed into the system for processing.

Processing Large Input Files

In some environments, the OSB project may begin processing a large input file before it has been completely copied to the input directory. To prevent this, configure the MinimumAge property in the “InboundProxyService” proxy service for the SGG-D8-ITRONXML-CM project. The MinimumAge property specifies the minimum age of files to be retrieved, based on the last modified time stamp. This enables large files to be completely copied to the input directory before they are retrieved for processing.

Prioritized Initial Measurement Processing

The Itron OpenWay adapter prioritizes processing of initial measurements created from smart meter commands and/or completion events by setting the Execution Method flag in these types of initial measurements to “Real Time” (D1RT). In addition, initial measurements received with the Execution Method flag set to “Real Time” will be processed in real time rather than via batch processing.

See **Initial Measurement Data Prioritization** in the *Oracle Utilities Meter Data Management / Smart Grid Gateway Business User Guide* for more information.

Device Events

Itron OpenWay can send real-time events such as meter outage alarms to Oracle Utilities Smart Grid Gateway. These events are referred to as exceptions in the Itron architecture, and are sent to the BPEL ExceptionSubscriberService service by Itron OpenWay without requiring an explicit request from Oracle Utilities Smart Grid Gateway. Exceptions are loaded as device events data and provided to the edge applications.

The required functionality is delivered in the base product as two OSB projects:

1. **SGG-D8-EXCEPTION-BASE** contains components responsible for “actual” processing of incoming data. It should not be modified during configuration. This project can be upgraded without affecting the customization and environment settings added to the SGG-D8-EXCEPTION-CM project.
2. **SGG-D8-EXCEPTION-CM** allows the customization and simplifies the future upgrades.

The following table describes the elements included in the EnvironmentSettings.xq file:

Element	Description	Valid Values
populateRaw	Determines if the device event data is populated as raw data.	true false
callPreProcessing	Determines if the preprocessing proxy service is called.	true

Element	Description	Valid Values
		false
callPostProcessing	Determines if the postprocessing proxy service is called.	true false
destinationRootElement	Holds the name of inbound web service for the device event seeder.	
publishServices/service	Specifies the name of the business service within the OSB project used to publish data for external systems (such as Oracle DataRaker).	
filterEvents	Determines if events should be filtered.	true false

Publishing Events

The Itron OpenWay adapter can be configured to publish device events for use in Oracle DataRaker or other external systems. This functionality is supported through a combination of OSB components and BPEL composites.

Enabling Device Event Publishing

Publishing data is enabled by referencing a publisher business service in the publishServices/service element in the EnvironmentSettings.xq file as follows:

```
<publishServices>
  <service>[publisherBusinessService]</service>
</publishServices>
```

The following components provided with the SGG-D8-EXCEPTION-CM OSB project are used in publishing device events data to Oracle DataRaker:

- The **DataRakerBusinessService** business service is used to send data to a pre-configured JMS queue (defined as an Endpoint URI), from which the data will be published. This is the business service that should be specified in the EnvironmentSettings.xq file.
- The **DataRakerServiceAccount** service account is used to define and maintain the user name and password needed to access the JMS queue defined in the **DataRakerBusinessService** business service.

Configuring Device Event Publishing Output

The SGGDRIntegration BPEL composite handles publishing the data to Oracle DataRaker or other systems.

Device event data is published in the “native” device event data format (the format of the device event seeder business object). This format includes normalized device event codes. See the *Oracle Utilities Smart Grid Gateway Adapter Development Kit Administrative User Guide* for more details about this format.

NOTE: In addition, filtering can NOT be applied to device events published via this feature.

The following parameters can be used to configure details of how the data is provided to Oracle DataRaker, including the directory where files are posted for Oracle DataRaker to consume, number of records per file, polling frequency, etc. These parameters are defined during installation. See the *Oracle Utilities Smart Grid Gateway Installation Guide* for more details about defining values for these parameters.

Parameter	Description	Default Value
SGG_DR_INT_QUEUE	JNDI name of queue to publish SGG payloads. This is the JMS queue defined in the DataRakerBusinessService business service. This should NOT be changed.	DataRakerQueue
SOA_DR_PUBLISH_SIZE	The number of records (SGG payloads) to accumulate in a published file.	100

Parameter	Description	Default Value
SOA_DR_FILE_SIZE	The maximum file size for the accumulated (SGG payloads) file in kilobytes.	524288
SOA_DR_ELAPSED_TIME	The period of time in second which, when exceeded, causes a new outgoing file to be created.	600
SOA_DR_POLLING_FREQ	The polling frequency in seconds of the staging directory for new files.	60
SOA_DR_STAGING_DIR	Mount point/directory for the staging directory for accumulated SGG payload files. This is used internally and should NOT be changed.	/spl/sploutput/staging
SOA_DR_INTEGRATION_DIR	Mount point/directory from which Oracle DataRaker will consume the converted XML files.	/spl/sploutput/int

Filtering Events

The Itron OpenWay adapter can be configured to filter device events passed into Oracle Utilities Smart Grid Gateway and Meter Data Management. Filtering data is enabled by setting the <filterEvents> element in the EnvironmentSettings.xq file to “true” as follows:

```
<filterEvents>true</filterEvents>
```

When filtering is enabled, only device events whose <externalEventName> matches one of the values defined in the **Itron - Device Event Mapping** extendable lookup (D8-DeviceEventMappingLookup) are passed into the system for processing.

Prioritized Device Event Processing

The Itron OpenWay adapter prioritizes processing of device events created from smart meter commands and/or completion events by setting the Execution Method flag in these types of device events to “Real Time” (DIRT). In addition, device events received with the Execution Method flag set to “Real Time” will be processed in real time rather than via batch processing.

See **Device Event Prioritization** in the *Oracle Utilities Meter Data Management / Smart Grid Gateway Business User Guide* for more information.

Base Package Business Objects

The Itron OpenWay adapter base package includes the following initial measurement and device event business objects:

Business Object Name	Description
D8-InterrogateByGroup	Itron - Interrogate By Group Performs a Schedule Read for a group of devices (either Application or Configuration) in the Itron OpenWay API.
D8-InterrogateByGroupResult	Itron - Interrogate by Group Result This BO is the inbound asynchronous response to an Interrogate By Group (Schedule Read) outbound communication.
D8-InitialLoadIMDInterval	Itron - Initial Load IMD - Interval Used when loading Itron OpenWay interval measurements into the system for the first time.
D8-InitialLoadIMDScalar	Itron - Initial Load IMD - Scalar

Business Object Name	Description
	Used when loading Itron OpenWay scalar measurements into the system for the first time.
D8-ScheduleRead	Itron - Schedule Read Requests IMD and Event data from the Itron OpenWay Network. This data will later be returned and placed in files to be uploaded.

Device Communication

The basic communication for all business processing is essentially the same. A communication request is sent from the Oracle Utilities application to Itron OpenWay. This request would be for a connect/disconnect, commission/decommission, measurement data, device status check, or an on-demand read. The designated BPEL process transforms the request from Oracle Utilities format to Itron OpenWay format and invokes the related Itron OpenWay web service. Itron OpenWay then returns a reply, and the BPEL process transforms the reply message back to the appropriate format so that Oracle Utilities can receive the response.

Communication Flows

The table below lists the communications created for each Itron OpenWay command:

Command	Outbound Communication	Inbound Communication	Completion Event
Remote Connect	Itron - Reconnect Meter	Itron - Reconnect Meter Result	Connect Device Completion Event
Remote Disconnect	Itron - Disconnect Meter	Itron - Disconnect Meter Result	Disconnect Device Completion Event
Device Commissioning	Itron - Add Meter Definition		Device Commissioning Completion Event
Device Decommissioning	Itron - Deregister Meter		Device Decommissioning Completion Event
On-Demand Read (Scalar)	Itron - Contingency Read (Scalar)	Itron - Contingency Read Result (Scalar)	Create IMD Completion Event
On-Demand Read (Interval)	Itron - Contingency Read (Interval)	Itron - Contingency Read Result (Interval)	Create IMD Completion Event
Scheduled Read (Scalar)	Itron - Interrogate by Group (Scalar)	Itron - Interrogate by Group Result (Scalar)	Create IMD Completion Event
Scheduled Read (Interval)	Itron - Interrogate by Group (Interval)	Itron - Interrogate by Group Result (Interval)	Create IMD Completion Event
Device Status Check	Itron - Ping by Endpoints (Status Check)	Itron - Get Ping by Endpoints Result	

Device Communication Base Package Business Objects

The Itron OpenWay Adapter base package includes the following communication business objects:

Business Object Name	Description
D8-AddMeterDefinitions	Itron - Add Meter Definition (Commission)
D8-DeregisterMeter	Itron - Deregister Meter (Decommission)
D8-DetectLoadSideVoltageByMtr	Itron - Detect Load Side Voltage
D8-DetLoadSideVoltageMtrRslt	Itron - Detect Load Side Voltage Result

Business Object Name	Description
D8-DisconnectMeter	Itron - Disconnect Meter (Remote Disconnect)
D8-DisconnectMeterResult	Itron - Disconnect Meter Result
D8-PingByEndpoints	Itron - Ping By Endpoints (Device Status Check)
D8-PingByEndpointsMDResponse	Itron - Multi-Device Ping Response
D8-PingByEndpointsMultiDevice	Itron - Ping By Endpoints Multi-Device
D8-PingByEndpointsResponse	Itron - Ping By Endpoints Response
D8-ReadDisconStateByMtr	Itron - Read Disconnect State
D8-ReadDisconStateMtrRslt	Itron - Read Disconnect State Result
D8-ReadInterval	Itron - Contingency Read (Interval)
D8-ReadResult	Itron - Contingency Read Result
D8-ReadScalar	Itron - Contingency Read (Scalar)
D8-ReconnectMeter	Itron - Reconnect Meter (Remote Connect)
D8-ReconnectMeterResult	Itron - Reconnect Meter Result

Itron OpenWay Event Data Mapping

The Itron OpenWay event file format maps as follows into the business object, D1-DeviceEventMappingLookup:

Itron OpenWay Flat File Field	Device Event Seeder BO Element	Comments
Transaction ID (from Header record)	External Source Identifier	This is the file name.
Device Identifier	External Device Identifier	
Event Name	External Event Name	
Event Creation Date/Time	Event Date/Time	
Device Type	External Device Type	This element has no real bearing on the device type within MDM/SGG. Its valid values include (although the element itself is free-form): Meter Collector Router
Service Location ID	External Service Location ID	
Communication Module Serial Number	External Communication Module Identifier	
Event Category ID	External Event Category	
Event Severity	External Event Severity	Valid values include (although the element itself is free-form): Alert Information
Status Value	External Status Value	This represents additional information that relates to the event itself.
Status Date/Time	External Status Date/Time	The date & time at which the additional information referenced above had occurred.

External System

You must create an External System for each external system to which Oracle Utilities Smart Grid Gateway will send messages. Each external system defines a set of outbound message types that will be sent to that system. Each external system outbound message type also specifies the following:

- The processing method used to send the message (Batch or Real-time)
- The corresponding message senders
- Batch Control (if Processing Method is set to Batch)
- Message XSL, W3C Schema, and Response XSL (as applicable)

Outbound Message Types

Acknowledgment and response messages are sent and received validating that commands have been transmitted.

Outbound Message Type	Description
D8ITRECO	D8 Itron Remote Connect

Inbound / Outbound Service Configuration

The inbound/outbound message utility allows you to configure your system to receive information from and to send information to external applications using XML. The Itron OpenWay adapter for Smart Grid Gateway uses one inbound web service to map device events. This is the same inbound web service used by the D1 application.

inbound Web Services

Inbound web services define the details of how messages are received from an external system, including the inbound communication business object (or business service or service script) to be invoked when the response message is received. As in the case of inbound communication business objects, the set of inbound web services you need to create is based on the types of messages the system is designed to send.

The Oracle Utilities Smart Grid Gateway adapter for Itron OpenWay includes the following inbound web services:

Inbound Web Service	Description
D1-BulkRequestHeader	Bulk Request Header
D1-BulkRequestUpdate	Bulk Request Update
D1-BulkResponse	Bulk Response
D1-DeviceEventSeeder	Used for upload of device events. The Device Event Seeder business object serves as a means of adding device events both from outside the application and from online. Its pre-processing algorithms determine the device event type - which in turn defines the device event BO that should be used to create the device event. If a device event type can't be determined, the device event is created using this BO. Such a device event can then be re-processed - and if successful, a new device event is created.
D1-DeviceStatusCheck	Device Status Check

Inbound Web Service	Description
	This service is invoked by the integration layer to instantiate a Device Status Check command.
D1-InitialLoadIMD	Used by OSB to instantiate an IMD The IMDSeeder business object is used to determine the type of initial measurement business object to instantiate when receiving usage readings from a head-end system.
D1-RemoteConnect	Remote Connect This service is invoked by the integration layer to instantiate a Remote Connect command.
D1-RemoteDisconnect	Remote Disconnect This service is invoked by the integration layer to instantiate a Remote Disconnect command.
D8-DetLoadSideVoltageMtrRslt	Itron - Detect Load Side Voltage by Meter Result
D8-DisconnectMeterResult	Itron - Disconnect Meter Result Service
D8-InterrogateByGroupResult	Itron - Interrogate By Group Result XIA
D8-PingByEndpointsMDResponse	Itron - Ping By Endpoints Multi-Device Response XAI Inbound
D8-PingByEndpointsResponse	Itron - Ping By End Response inbound web service
D8-ReadDisconStateMtrRslt	Itron - Read Disconnect State by Meter Result Service
D8-ReadResult	Itron - Contingency Read Result
D8-ReconnectMeterResult	Itron - Reconnect Meter Result Service

Message Senders

Message senders define the details of how messages are sent to an external system. As in the case of outbound communication business objects and outbound message types, the set of message senders you need to create is based on the types of messages the system is designed to accept.

The Oracle Utilities Smart Grid Gateway adapter for Itron OpenWay includes the following message senders:

Message Sender	Description
D8-RemoteCon	D8 Remote Connect

BPEL Processes

These processes are responsible for performing the conversion from Oracle Utilities format to Itron OpenWay format, invoking process callouts and invoking the remote endpoint to trigger the device events. BPEL modules are divided into “Composites,” or collections of business logic.

OnDemandRead Composite Process — The OnDemandRead composite invokes a sequence of web methods that call the head-end system to retrieve meter reading data and send it back to the OUAF layer. In the case of Itron OpenWay, this is accomplished using the ContingencyReadByEndpoints/GetContingencyReadByEndpointsResult services defined in the Data service WSDL.

ConnectDisconnect Composite Process — This composite is responsible for triggering the Connect and Disconnect events on the head-end system. A second, asynchronous reply sends the results back into the OUAF layer when the head-end system signals a change in the status of the device. The Itron OpenWay Adapter uses the ReconnectMeter/

GetReconnectMeterResult web service pair to connect, and the DisconnectMeter/GetDisconnectMeterResult web service pair to disconnect. Each of these services is defined in the Control service WSDL.

CommissionDecommission Composite Process — Commissioning and Decommissioning of devices, sometimes referred to as “Provisioning,” is handled by the CommissionDecommission composite. Commissioning invokes the AddMeterDefinitions web service and Decommissioning uses the DeregisterMeters web service. Both are defined in the Provisioning service.

DeviceStatusCheck Composite — This composite uses the PingByEndpoints and GetPingByEndpointsResult web services to check the health of a device. These Itron OpenWay services are defined in the Control.Diagnostic service.

Common Composite — The Common composite contains three main classes of operations: Proxies, ProcessCallouts, and utility functions. Proxies are usually simple mediators that forward a web service call to a preset endpoint. In this Adapter they have two additional roles. First, they interrogate a composite property and determine whether the supported version of head-end software is 3.70 or 3.90. Some of the proxies will also append the callback URL for the StatusChanged service. Proxies are convenient because they allow head-end URLs and security to be set in a single composite. In this case, they also offer a common location to make these checks. ProcessCallouts are points of customization which allow users to modify data and/or initiate some external business process. Utility functions serve as a central location for business logic needed by multiple composites.

Web Services

The following web services are all defined in the Itron OpenWay head-end system:

- **CommissionDecommissionService**
 - **BPEL Process:** CommissionDecommission
 - **Operation:** AddMeterDefinitions
 - **Endpoint URL:** http://<EM_SERVER>:<EM_SERVER_PORT>/soa-infra/services/<EM_PARTITION>/CommissionDecommission/CommissionDecommissionService
- **CommissionDecommissionService**
 - **BPEL Process:** CommissionDecommission
 - **Operation:** DeregisterMeters
 - **Endpoint URL:** http://<EM_SERVER>:<EM_SERVER_PORT>/soa-infra/services/<EM_PARTITION>/CommissionDecommission/CommissionDecommissionService
- **ConnectDisconnectService**
 - **BPEL Process:** ConnectDisconnect
 - **Operation:** ReconnectMeter
 - **Endpoint URL:** http://<EM_SERVER>:<EM_SERVER_PORT>/soa-infra/services/<EM_PARTITION>/ConnectDisconnect/ConnectDisconnectService
- **ConnectDisconnectService**
 - **BPEL Process:** ConnectDisconnect
 - **Operation:** DisconnectMeter
 - **Endpoint URL:** http://<EM_SERVER>:<EM_SERVER_PORT>/soa-infra/services/<EM_PARTITION>/ConnectDisconnect/ConnectDisconnectService
- **DeviceStatusCheckService**
 - **BPEL Process:** DeviceStatusCheck
 - **Operation:** DeviceStatusCheck

- **Endpoint URL:** http://<EM_SERVER>:<EM_SERVER_PORT>/soa-infra/services/<EM_PARTITION>/DeviceStatusCheck/DeviceStatusCheckService
- **OnDemandReadService**
 - **BPEL Process:** OnDemandRead
 - **Operation:** ContingencyReadByEndpoints
 - **Endpoint URL:** http://<EM_SERVER>:<EM_SERVER_PORT>/soa-infra/services/<EM_PARTITION>/OnDemandRead/OnDemandReadService

Itron OpenWay Web Services

The following table describes Itron OpenWay web services and operations used for the Oracle Utilities Smart Grid Gateway command messaging:

Smart Grid Gateway Command	AMI Adapter Business Objects	Itron OpenWay Web Services	Itron OpenWay Operations
Device Commissioning	D8-AddMeterDefinitions	ProvisioningService	AddMeterDefinitions
Device Decommissioning	D8-DeregisterMeter	ProvisioningService	DeregisterMeters
Remote Connect	D8-ReconnectMeter	ControlService	ReconnectMeter
	D8-DetectLoadSideVoltageByMtr	DiagnosticService	DetectLoadSideVoltageByMeter
	D8-ReadDisconStateByMtr		ReadDisconnectStateByMeters
Remote Disconnect	D8-DisconnectMeter	ControlService	DisconnectMeter
Device Status Check	D8-PingByEndpoints	ControlService	PingByEndpoints
			GetPingByEndpointsResult
On-Demand Read	D8-ReadInterval	DataService	ContingencyReadByEndpoints
	D8-ReadScalar		GetContingencyReadByEndpointsResult
	D1-InitialLoadIMD		InterrogateByGroup
	D1-DeviceEventSeeder	GetInterrogateByGroupResult	
		DataSubscriberService (Async Response)	DataArrived
Event Loading	D1-DeviceEventSeeder	ExceptionSubscriberService	ExceptionsArrived

Chapter 3

Configuring an Itron OpenWay Head-End System

This section outlines the configuration required for the Oracle Utilities Smart Grid Gateway Adapter for Itron OpenWay to communicate with the Itron OpenWay application.

Master Configurations

Master Configurations are sources of global parameter records used by a system implementation. This section describes the master configuration that is specific to Oracle Utilities Smart Grid Gateway Adapter for Itron OpenWay. Refer to the Oracle Utilities Meter Data Management / Smart Grid Gateway documentation for more information about other master configurations used by Oracle Utilities Smart Grid Gateway.

Itron Version Master Configuration

This master configuration specifies the version of the Itron head-end system with which the system is communicating. The configuration options are Itron OpenWay 3.70, Itron OpenWay 3.90, and Itron OpenWay 6.10. Only one version can be specified at a time.

Inbound Web Services

Inbound web services define the details of how messages are received from an external system. This includes incoming usage and device events, as well as messages sent from the Itron OpenWay application in response to a command request.

The following inbound web services must be configured in your system. If these are not present in your configuration, add them. Refer to the Oracle Utilities Application Framework documentation for more information about creating inbound web services.

Inbound Web Service Name	Description
D1-BulkRequestHeader	Bulk Request Header
D1-BulkRequestUpdate	Bulk Request Update

Inbound Web Service Name	Description
D1-BulkResponse	Bulk Response
D1-DeviceEventSeeder	Device Event Seeder
D1-DeviceStatusCheck	Device Status Check
D1-InitialLoadIMD	Used by OSB to instantiate an IMD
D1-PayloadErrorNotif	Payload Error Notification
D1-PayloadStatistics	Payload Statistics
D1-PayloadSummary	Payload Summary
D8-DetLoadSideVoltageMtrRslt	Itron - Detect Load Side Voltage by Meter Result
D8-DisconnectMeterResult	Disconnect Meter Result Service
D8-InterrogateByGroupResult	Itron - Interrogate By Group Result XIA
D8-PingByEndpointsMDResponse	Itron - Ping By Endpoints Multi-Device Response XAI Inbound
D8-PingByEndpointsResponse	Itron - Ping By Endpoints Response inbound web service
D8-ReadDisconStateMtrRslt	Itron - Read Disconnect State by Meter Result Service
D8-ReadResult	Itron - Contingency Read Result
D8-ReconnectMeterResult	Itron - Reconnect Meter Result Service

Note: The following apply to all of the above inbound web services:

Message Options

- **Trace:** No
- **Debug:** No
- **Active:** Yes

Operations

- **Operation Name:** Same as web service name
- **Schema Type:** Business Object
- **Schema Name:** Applicable business object code
- **Transaction Type:** Add

Message Senders

Message senders define the details of how messages are sent to an external system, such as messages containing device command requests. An message sender should be configured for each command.

The following message senders must be configured in your system. If these are not present in your configuration, add them. Refer to the Oracle Utilities Application Framework documentation for more information about creating message senders.

Message Sender	Description
D8-ITRON_IBG	Itron Interrogate By Group
D8-COMM	Itron Commission
D8-DCOMM	Itron Decommission
D8-DLSV	Itron Detect Load Side Voltage
D8-IPBE	Itron Ping By Endpoints
D8-RCONN	Itron Remote Connect
D8-RDCONN	Itron Remote Disconnect
D8-RDSS	Itron Read Disconnect State

Note: The following apply to all of the above message senders:

Main Tab:

- **Invocation Type:** Real-time
- **Message Class:** RTHTTPSND (Sender routes message via HTTP real-time)
- **MSG Encoding:** UTF-8 message encoding

Context Tab:

- **HTTP Header:** SOAPAction:http://xmlns.oracle.com/ouaf/Itron/<OPERATION>
- **HTTP Login User:** <USER_ID>
- **HTTP Login Password:** <PASSWORD>
- **HTTP Method:** POST
- **HTTP URL 1:** http://<EM_SERVER>:<EM_SERVER_PORT>/soa-infra/services/Itron/<SERVICE>
- where:
 - <OPERATION>: the operation performed by the message sender (see Operation column in the table above)
 - <USER_ID>: the user ID used to log into WebLogic Enterprise Manager
 - <PASSWORD>: the password used to log into WebLogic Enterprise Manager
 - <EM_SERVER>: the machine name or IP address of server where the WebLogic Enterprise Manager is installed
 - <EM_SERVER_PORT>: the port where the WebLogic Enterprise Manager is installed
 - <SERVICE>: the service invoked by the message sender (see Service column in the table above)

How to Use Enterprise Manager to Locate the URL for the Message Sender

Follow this procedure to find the correct URL for the command associated with an message sender:

1. Open Enterprise Manager and use the navigation pane to open the dashboard of the service used by the message sender:
2. The top bar of the dashboard contains several buttons and icons. One of these is a “world” icon with a puzzle piece over it. Click this icon to display a list of the WSDLs and endpoint URIs for the service:
3. Click the service’s WSDL URL link to see the WSDL in the browser, or right click and save it to your machine.

Outbound Message Types

Outbound message types define specific types of messages sent to an external system, such as messages containing device command requests.

The following outbound message types must be configured in your system. If these are not present in your configuration, add them. Refer to the Oracle Utilities Application Framework documentation for more information about creating outbound message types.

Outbound Message Type	Description
D8-CONREAD	Itron Contingency Read By Endpoints
D8-INTRGRP	Itron Interrogate By Group
D8-COMM	Itron Commission
D8-DCOMM	Itron Decommission
D8-DSC	Itron Device Status Check
D8-DSCMD	Itron Device Status Check Multi Device
D8-DLSV	Itron Detect Load Side Voltage
D8-RCONN	Itron Remote Connect

Outbound Message Type	Description
D8-RDCONN	Itron Remote Disconnect
D8-RDSS	Itron Read Disconnect State

Note: The following apply to all of the above outbound message types:

- **Business Object:** D1-OutboundMessage (Outbound Message)
- **Priority:** Priority 50

External System

External systems represent external applications with which the Smart Grid Gateway will exchange messages or data. In the case of the Smart Grid Gateway adapters, external systems represent the head-end systems with which the adapters communicate.

An external system that represents the Itron OpenWay head end must be present in your system. If this is not present in your configuration, add it, along with the following Outbound Message Types. Refer to the Oracle Utilities Application Framework documentation for more information about creating external systems.

External System - Itron OpenWay:

- **External System:** Itron OpenWay
- **Description:** Itron OpenWay
- **Outbound Message Types:**

Outbound Message Type	Message Sender
D8-INTRGRP	D8-ITRON_IBG
D8-COMM	D8-COMM
D8-DCOMM	D8-DCOMM
D8-DSC	Message sender associated with the Device Status Check Outbound Message Type
D8-DSCMD	Message sender associated with the Device Status Check Multi Device Outbound Message Type
D8-DLSV	D8-DLSV
D8-RCONN	D8-RCONN
D8-RDCONN	D8-RDCONN
D8-RDSS	D8-RDSS

Note: The following apply to all of the above outbound message types:

- **Processing Method:** Real-time
- **Message XSL:** D8-Request.xsl
- **Response XSL:** D8-Response.xsl

Service Provider

Service providers represent external entities that serve various roles relative to the application, including head-end systems, billing systems to which the application sends bill determinant data, market participants in a deregulated environment, outage management systems that receive meter event data from the application, or other parties that require or provide information to the system. The head-end systems that collect and send measurement data and meter events to the application are defined as service providers.

A service provider that represents the Itron OpenWay head end must be present in your system. If this are not present in your configuration, add it. Refer to the Oracle Utilities Service and Measurement Data Foundation documentation for more information about creating service providers.

Service Provider - Itron OpenWay:

- **Service Provider:** Itron OpenWay
- **Description:** Itron OpenWay
- **External Reference ID:** Itron OpenWay
- **External System:** Itron OpenWay
- **Our Name/ID in Their System:**
- **AMI Device ID Type:** Internal Meter Number
- **Pre-Commissioning Device ID Type:**
- **AMI Measuring Component ID Type:** Channel ID

Processing Methods

Processing methods define the format or means by which a service provider receives and/or sends data from and/or to the application, including bill determinants, usage data, or device events. Processing methods are also used to define how to create information internal to the application such as initial measurement data and device events. Processing methods can also be used to define how command requests are sent to Itron OpenWay.

The following types of processing methods must be configured for the Itron OpenWay service provider. Refer to the Oracle Utilities Service and Measurement Data Foundation documentation for more information about configuring processing methods.

Initial Measurement Creation

Initial measurement creation processing methods define the business objects used to create initial measurements. The IMD Seeder inbound web service uses this processing method to determine which type of initial measurement business object to instantiate when receiving usage from the Itron OpenWay application.

Device Event Mapping

Device event mapping processing methods define how head-end-specific device events are mapped to standard device event names. The Device Event Seeder inbound web service uses this processing method to determine which type of device event business object to instantiate when receiving device events from the Itron OpenWay application.

UOM Translation

UOM translation processing methods define how head-end-specific unit of measure (UOM) codes are mapped to standard UOM codes. This processing method is used to determine how to map Itron UOM codes to standard UOM codes when receiving usage from the Itron OpenWay system.

Commands

Command processing methods define how command requests are sent to a head-end system. More specifically, they define the type of outbound communication business object to create for each type of command, and the outbound message type to send to the head-end system.

The following types of command processing methods can be configured for the Itron OpenWay service provider, based on the requirements of each implementation.

The following types of command processing methods can be configured for the Itron OpenWay service provider, based on the requirements of each implementation using the “How to Create OB COMM/Send OB Message” processing method business object (D1–HowToCreateActivityOBComm).

Command	Processing Role	Default Business Object	Default Outbound Message Type
Device Commission	Device Registration	D8-AddMeterDefinitions	Commission
Device Decommission	Device Removal	D8-DeregisterMeter	Decommission
Device Status Check	Device Status Check	D8-PingByEndpoints	Device Status Check
Load Check*	Load Check	D8-DetectLoadSideVoltageByMtr	Detect Load Side Voltage by Meter
Multi-Devie Status Check	Multi-Devie Status Check	D8-PingByEndpointsMultiDevice	Read Disconnect State By Meters
On-Demand Read (Scalar)	On-Demand Read (Scalar)	D8-ReadScalar	On Demand Read — Scalar
Remote Connect	Remote Connect	D8-ReconnectMeter	Connect
Remote Disconnect	Remote Disconnect	D8-DisconnectMeter	Disconnect

* The Load Check processing method can be used to ensure that it is safe to connect a meter. The load check processing method is executed during the Connection Ready state of the Remote Connect activity and can be configured to detect the possibility of a load side voltage at the meter, or to check the disconnect meter switch on the meter.

If the Read Disconnect State business object is used as the processing method, you should configure the disconnect switch state values that indicate the state of the switch. These values are defined on the D8-DisconnectSwitchStateLookup extendable lookup, and are then added in the Disconnect Switch State Error Values section of the Itron - Read Disconnect State Result Inbound communication type. See [Itron OpenWay Disconnect Switch State Lookup](#) for more information.

Configuring Endpoint URIs

Part of configuring your Itron OpenWay adapter is configuring your BPEL composites to work with your head-end system by defining the appropriate Endpoint URIs for each of the commands.

The default approach to defining Endpoints URIs is redeployment or reinstallation of the BPEL composites. For example, by default, changing an adapter from using the test harness to a production environment using the actual head end system requires editing the appropriate installation menu options and redeploying the BPEL composites. The Endpoints URIs defined during installation and deployment for each adapter are listed in the **Smart Grid Gateway Installation and Configuration Worksheets** section of the *Oracle Utilities Smart Grid Gateway Installation Guide*.

You can also use an “Endpoint Override” Domain Value Map (DVM) to override Endpoints URIs defined during deployment and installation. This DVM allows defining specific keys that provide an alternate URL that will override the original installed value. Endpoint Override DVMs can be edited using the Oracle SOA Composer.

The Itron OpenWay endpoint override DVM (D8–EndpointOverrides.dvm) uses a specific set of keys, each used for one or more commands. The table below lists the DVM keys available for the Itron OpenWay adapter and the command used with each:

DVM Key	Commands
DataService390	Scheduled Read (Scalar) Scheduled Read (Interval) On-Demand Read (Scalar) On-Demand Read (Interval)
ProvisioningService370	Device Commissioning
ProvisioningService390	Device Decommissioning

DVM Key	Commands
ControlService370	Device Status Check (if using v3.70) Remote Connect (if using v3.70) Remote Disconnect (if using v3.70)
ControlService390	Remote Connect (if using v3.90) Remote Disconnect (if using v3.90)
DiagnosticService390	Device Status Check (if using v3.90)
ProcessCallout	User Exit Functions

NOTE:

The numbers in the keys above (370 and 390) designate the version of the Itron OpenWay head-end system. Only a single version of each key should be defined, based on the version of the Itron OpenWay head-end system

To define an override Endpoint URI for the Itron OpenWay adapter, use the following procedure:

1. Open the SOA Composer for your BPEL configuration.

The URL for the SOA Composer is <http://server:port/soa/composer>.

2. Select the D8-EndpointOverrides.dvm in the left panel.

- In Fusion Middleware v12.2.1, this is located under **Shared** in the **Deployment View**, or under **Domain Value Maps** in the **Types View**.
- In Fusion Middleware v12.2.2, this is located under **Metadata** in the **Deployment View**.

The DVM will open in the right panel. The **Description** field lists the available keys for the DVM (only the first key is shown, but you can scroll through the contents to view the list. The panel also displays a list of previously defined keys.

3. Click the **Create Session** button (above the left panel) to begin an editing session.
4. Click the **Add Domain Values** icon (“+”) to add a new key.

The **Add Domain Values** dialog opens.

5. Enter the appropriate values in the **Add Domain Values** dialog as follows:

- **key:** The DVM key for the Endpoint URI you wish to define (see the table above).
- **EndpointURI:** The override Endpoint URI.

6. Click **OK**.

The new DVM value will appear in the list of keys.

7. Click the **Save** icon to save the DVM values.

8. Click the **Publish** button to activate all the changes in the editing session.

Enter an optional note for the session in the **Publish Session** dialog.

Changes take effect immediately upon publishing the session.

Click **Discard** to discard your changes.

Click **Exit** to exit your current session. Note that your session will still be open if you exit. Use **Discard** to end your session without making changes.

Chapter 4

Configuring Itron OpenWay Extendable Lookups

This section outlines some of the extendable lookups that must be configured for use with the Itron OpenWay adapter. Refer to the Oracle Utilities Application Framework documentation for more information about working with extendable lookups.

Itron Application Group Lookup

The Itron OpenWay head-end system allows users to define addressable groups of meters based on criteria such as a geographic identifier or zip code. These application groups can be represented in Oracle Utilities Smart Grid Gateway by using the Itron OpenWay Application Group Lookup. The groups defined for this lookup are used with Scheduled Read commands.

Each value defined for the Itron OpenWay Application Group extendable lookup should include the following:

- **Application Group:** The application group name as defined in the Itron OpenWay head-end system.
- **Description:** A description of the application group.

Itron Device Event Mapping

The Itron OpenWay Device Event Mapping extendable lookup is used to determine which type of device event business object to instantiate when receiving device events from the Itron OpenWay application.

Each value defined for the Itron OpenWay Device Event Mapping extendable lookup should include the following:

- **Head-End System Event Name:** The event name used by the Itron OpenWay application
- **Description:** A description of the device event
- **Status:** The status of the lookup value (can be Active or Inactive)
- **Standard Event Name:** The standard event name for device events of this type, from the “Standard Event Name” extendable lookup.

Itron Disconnect Switch State Lookup

Some utilities may require verification of the load on a meter to ensure that it is safe to connect the meter. One method of doing this is to check the disconnect switch on the meter to see if it is connected or armed. To use this method, the Itron

Openway Disconnect Switch State extendable lookup should be configured to list the possible states of the meter disconnect switch.

Each value defined for the Itron OpenWay Interval Disconnect Switch State extendable lookup should include the following:

- **Switch State:** The Itron OpenWay disconnect switch state code
- **Description:** A description of the disconnect switch state code.

Itron UOM Code to Standard UOM Mapping

Usage received from Itron OpenWay may use utility-specific unit of measures (UOMs). These custom UOMs must be mapped to standard UOM codes. The Itron OpenWay UOM Code to Standard UOM Mapping extendable lookup is used to determine how to map Itron OpenWay UOM codes to standard UOM codes when receiving usage from the Itron OpenWay application.

Each value defined for the Itron OpenWay UOM Code to Standard UOM Mapping extendable lookup should include the following:

- **Head-End Unit of Measure:** The unit of measure code used by the Itron OpenWay application
- **Description:** A description of the unit of measure code.
- **Status:** The status of the lookup value (can be Active or Inactive)
- **Unit of Measure:** The unit of measure defined in the system.
- **Time of Use:** An optional time of use period, defined in the system, used to further distinguish the unit of measure.
- **Service Quantity Identifier:** An optional service quantity identifier, defined in the system, used to further distinguish the unit of measure.
- **Unit of Measure Magnitude Conversion:**
 - **Magnitude Multiplier:** An optional; multiplier used to convert raw values received from the head-end system to values appropriate for use with the system. For example, if a reading or interval data is received in Wh, a multiplier of “.001” would convert Wh to kWh. If not provided or left blank, no conversion is performed.

Itron Interval Status Code to Condition Mapping

Interval usage received from the Itron OpenWay application can include Itron OpenWay interval status codes that indicate the status or condition of the interval value. These interval status codes must be mapped to standard condition codes in the system. The Itron OpenWay Interval Status Code to Condition Mapping extendable lookup is used to determine how to map Itron OpenWay interval status codes to standard status codes when receiving usage from the Itron OpenWay application.

Each value defined for the Itron OpenWay Interval Status Code to Condition Mapping extendable lookup should include the following:

- **Interval Status:** The Itron OpenWay interval status code
- **Description:** A description of the interval status code.
- **Status:** The status of the lookup value (can be Active or Inactive)
- **Condition:** The condition code to which the interval status code is to be mapped, from the Measurement Condition extendable lookup.

Other Extendable Lookups

Business Object Name	Description
D8-DvcStatusMappingLookup	Itron - Device Status Lookup
D8-FailureReasonLookup	Itron - Failure Reason Lookup

Business Object Name	Description
D8-ItronVersions	Itron - Versions Lookup
D8-JobStatusLookup	Itron - Job Status
D8-ResultLookup	Itron - Result Lookup
D8-RptSelfReadOptionLookup	Itron - Report Self Read Option Lookup

Chapter 5

Extending the Itron OpenWay Adapter

The Oracle Utilities Smart Grid Gateway Adapter for Itron OpenWay supports a number of commands, including:

- Commission Device
- Decommission Device
- Device Status Check
- Multi-Device Status Check
- On-Demand Read
- Remote Connect
- Remote Disconnect

The Adapter for Itron OpenWay can be extended to support additional commands provided by the Itron OpenWay application.

Chapter 6

The Itron OpenWay Test Harness

Oracle Utilities Smart Grid Gateway Adapter for Itron OpenWay includes a test harness that can be configured to simulate a general head-end system for testing the two-way commands. The test harness includes a BPEL composite, web services for standard meter functions, and an XML file that can be used to contain information for one or more meters. This chapter describes the test harness and its components.

Test Harness Design

The Itron Harness is divided into two main layers. The “front end” set of services implements the Itron-specified interfaces. They receive requests corresponding to the following:

- [www.itron.com.ami.2008.10.control](http://www.itron.com/ami/2008.10/control)
 - ReconnectMeter
 - GetReconnectMeterResult
 - DisconnectMeter
 - GetDisconnectMeterResult
- [www.itron.com.ami.2008.10.data](http://www.itron.com/ami/2008.10/data)
 - ContingencyReadByEndpoint
 - GetContingencyReadByEndpointResult
 - InterrogateByGroup
 - GetInterrogateByGroupResult
- [www.itron.com.ami.2008.10.control](http://www.itron.com/ami/2008.10/control)
 - PingByEndpoints
 - ReconnectMeter
 - GetReconnectMeterResult
 - DisconnectMeter

- GetDisconnectMeterResult
- www.itron.com.ami.2012.03.control.diagnostic
 - PingByEndpoints
 - GetPingByEndpointsResult
- www.itron.com.ami.2009.08.provisioning
 - AddMeterDefinitions
 - DeregisterMeters

Each of these services calls into the “back end” layer, which defines meters and sets their attributes. These meters are stored in a file within the test harness called meterdb.xml. This file can be modified pre-deployment. Post-deployment changes to the file are not supported. However, the Test Harness retains an in-memory “database” of the meters in the file. The in-memory representation can be modified using the Utility web services. Note that any changes to the in-memory structure will be lost when the server is restarted or the Test Harness composite is redeployed.

Locating the WSDL for the Test Harness

Follow these procedures to locate the Itron OpenWay test harness WSDL:

How to Use Enterprise Manager to Locate the WSDL

1. Open Enterprise Manager and use the navigation pane to open the dashboard of the test harness composite:
2. The top bar of the dashboard contains several buttons and icons. One of these is a “world” icon with a puzzle piece over it. Click this icon to display a list of the WSDLs and endpoint URIs for the composite:
3. Click the UtilService WSDL URL link to see the WSDL in the browser, or right click and save it to your machine

Depending on your requirements, it may be necessary to download the associated schema found in the wsdl:types section. The URL can be pasted into a browser tab and downloaded in the same manner as the WSDL. The main schema has imported schemas that may also be required.

How to Use a Direct URL to locate the WSDL

The WSDL can be accessed without Enterprise Manager by understanding the paths used on the SOA server. In general, they have the following form:

```
http://{server name}:{port number}/soa-infra/services/{partition}/{Composite}/{Web Service}?WSDL
```

So by default, the test harness WSDL can be found at

```
http://{server name}:{port number}/soa-infra/services/Itron_Test/ItronTestHarness/UtilService?WSDL
```

Web Services

This section describes the web services included in the Itron OpenWay test harness BPEL composite.

General Services

This section describes the general services of the Itron OpenWay test harness composite.

LoadMeterIndex

This web service loads the data store from the internal file. By default, if the store is already in memory, it will NOT reload. This behavior can be overridden with the forceReload parameter.

Input: LoadMeterIndexInput

Part: payload

Element: LoadMeterIndexRequest

Parameter	Description
forceReload	A switch telling the system whether to reload the meter index from the configuration file. Default is false.

Output: LoadMeterIndexOutput

Part: payload

Element: LoadMeterIndexResult

Parameter	Description
loaded	A boolean value for whether or not the index was reloaded from the configuration file

Fault: UtilityFault (see [UtilityFault](#) for more details).

ViewAuditTrail

This web service returns the audit log for the entire session.

Input: ViewAuditTrailInput

Part: payload

Element: ViewAuditTrailRequest

Output: ViewAuditTrailOutput

Part: payload

Element: ViewAuditTrailResult

An Entry consisting of a timestamp and an Operation. Each entry may have an associated meter object showing the latest update.

Fault: UtilityFault (see [UtilityFault](#) for more details).

UtilityFault

Fault with similar mapping to SGG/OUAF faults:

Typically, the faultCode, faultString, faultActor, and detail/text elements will be populated.

Locate Meter Services

This section describes the locate meter web services of the Itron OpenWay test harness composite.

FindMeters

This web service queries the data store for one or more meters. The difference between GetMeter and FindMeters is GetMeter can return at most one meter and it must match the provided ID exactly. GetMeter will throw an error if the ID is not found. FindMeters can return more than one meter (when using the regex) and will not throw an error when the ID does not match any of the meters in the index.

Input: FindMetersInput

Part: payload

Element: FindMetersRequest

Parameter	Description
id	The meter ID for which to search
isRegex	The provided id can be a regex value when this parameter is true. Hint: to search for all meters in the system, use ".*" for the ID.

Output: FindMetersOutput

Part: payload

Element: FindMetersResult

Zero or more meter objects can be returned from the search

Fault: See [UtilityFault](#). Unlike other methods, FindMeters does not throw an exception if the meter is not found. As such, it can be used to test for the existence of a Meter prior to querying for it.

IsMeterDefined

This web service queries whether a particular meter is defined in the data store.

Input: IsMeterDefinedInput

Part: payload

Element: IsMeterDefinedRequest

Parameter	Description
id	The meter ID for which to search

Output: IsMeterDefinedOutput

Part: payload

Element: IsMeterDefinedResult

Whether or not the provided ID is part of the index.

Fault: See [UtilityFault](#). Thrown when meter id is not found.

GetMeter

This web service returns all the attributes of a single meter from the in-memory data store. The difference between GetMeter and FindMeters is GetMeter can return at most one meter and it must match the provided ID exactly. GetMeter will throw an error if the ID is not found. FindMeters can return more than one meter (when using the regex) and will not throw an error when the ID does not match any of the meters in the index.

Input: GetMeterInput

Part: payload

Element: GetMeterRequest

Parameter	Description
id	The meter ID for which to search

Output: GetMeterOutput

Part: payload

Element: GetMeterResult

The meter object requested by the ID.

Fault: See [UtilityFault](#). Thrown when meter id is not found.

GetGroupMeters

This web service retrieves the set of meters with the specified group name

Input: GetGroupMetersInput

Part: payload

Element: GetGroupMetersRequest

Parameter	Description
id	The group name common to the meters to retrieved

Output: GetGroupMetersOutput

Part: payload

Element: GetGroupMetersResult

The set of meters with the provided group name.

Fault

Fault: See [UtilityFault](#). Thrown only in unusual circumstances.

Meter Administration Services

This section describes the meter administration services of the Itron OpenWay test harness composite.

AddMeters

This web service adds a set of meters to the in-memory data store. This will not permanently add it to the control file.

Input: AddMetersInput

Part: payload

Element: AddMetersRequest

Parameter	Description
id	The identification code for the meter.
utility	An informational string.
serviceType	One of the valid ServiceType values (see schema). "Electric" is the only option at this time.
isCommissioned	Whether or not the meter is in a commissioned state.

Parameter	Description
loadActionCode	One of the possible LoadActionCode values used in Connect and Disconnect (see schema).
outageEventType	One of the possible OutageEventType values used in Device Status Check (see schema).
executionStatus	One of the possible ExecutionStates (see schema). These values control how the meter will respond to commands.
groupName	The name linking multiple meters together into a set.
updateIfExisting	Whether or not to update the meter with the provided values if it already exists in the index.
Comment	An informational string describing the purpose of the meter.
Channels	A listing of unit of measures supported by this meter.
uomCode	A code describing the unit of measure for the channel.
uomName	A short string containing the name of the unit of measure.
decimals	The number of digits to the right of the decimal that should be generated when reading the meter.
description	A longer description of the unit of measure.

Output: AddMetersOutput

Part: payload

Element: AddMetersResult

Whether or not each meter was added to the index.

Fault: See [UtilityFault](#).

RemoveMeter

This web service removes a meter from the in-memory data store. This will not permanently remove it from the control file.

Input: RemoveMeterInput

Part: payload

Element: RemoveMeterRequest

Parameter	Description
id	The ID for the meter to be removed

Output: RemoveMeterOutput

Part: payload

SElement: RemoveMeterResult

Whether or not the meter was removed from the index.

Fault: See [UtilityFault](#). Thrown when meter id is not found.

AddMeterChannel

This web service adds a new channel to a single meter.

Input: AddMeterChannelInput

Part: payload

Element: AddMeterChannelRequest

Parameter	Description
id	The identification code for the meter.
uomCode	A code describing the unit of measure for the channel.
uomName	A short string containing the name of the unit of measure.
decimals	The number of digits to the right of the decimal that should be generated when reading the meter.
description	A longer description of the unit of measure.

Output: AddMeterChannelOutput

Part: payload

Element: AddMeterChannelResult

Whether or not the channel was added to the index.

Fault: See [UtilityFault](#). Thrown when meter id is not found.

RemoveMeterChannel

This web service removes a Channel from a meter.

Input: RemoveMeterChannelInput

Part: payload

SElement: RemoveMeterChannelRequest

Parameter	Description
id	The ID for the meter to be removed.
uomCode	A code describing the unit of measure for the channel.
uomName	A short string containing the name of the unit of measure.

These three parameters are combined to locate a unique channel

Output: RemoveMeterChannelOutput

Part: payload

Element: RemoveMeterChannelResult

Whether or not the channel was removed from the meter.

Fault: See [UtilityFault](#). Thrown when meter id is not found.

ReadScalarMeter

This web service generates a scalar reading for each channel of a given meter.

Input: ReadScalarMeterInput

Part: payload

Element: ReadScalarMeterRequest

Parameter	Description
id	The ID for the meter to be read

Output: ReadScalarMeterOutput

Part: payload

Element: ReadScalarMeterResult

Zero or more scalar readings for the given meter.

Parameter	Description
uomCode	A code describing the unit of measure for the channel.
uomName	A short string containing the name of the unit of measure.
decimals	The number of digits to the right of the decimal that should be generated when reading the meter.
description	A longer description of the unit of measure.
value	A random number representing the scalar reading.

Meter Attribute Administration Services

This section describes the meter administration services of the Itron OpenWay test harness composite.

GetDeviceStatus

This web service queries the device status of a meter. The `OutageEventType` is used by `DeviceStatusCheck`. The possible values of a `deviceStatusType` are:

Status	Description
Instantaneous	The meter responds immediately to the status check
SlowResponse	An <code>EndpointFailure</code> with a <code>FailureReason</code> of "ErrorResponse" will be returned from the Test Harness
NoResponse	An <code>EndpointFailure</code> with a <code>FailureReason</code> of "NoResponse" will be returned from the Test Harness
Unknown	An <code>EndpointFailure</code> with a <code>FailureReason</code> of "Unspecified" will be returned from the Test Harness

Input: GetDeviceStatusInput

Part: payload

Element: GetDeviceStatusRequest

Parameter	Description
id	The ID for the meter for which the status should be retrieved

Output: GetDeviceStatusOutput

Part: payload

Element: GetDeviceStatusResult

The value of the device status for the requested meter.

Fault: See [UtilityFault](#). Thrown when meter id is not found.

SetDeviceStatus

This web service updates the device status for a given meter. The possible values of a deviceStatusType are:

Status	Description
Instantaneous	The meter responds immediately to the status check
SlowResponse	An EndpointFailure with a FailureReason of "ErrorResponse" will be returned from the Test Harness
NoResponse	An EndpointFailure with a FailureReason of "NoResponse" will be returned from the Test Harness
Unknown	An EndpointFailure with a FailureReason of "Unspecified" will be returned from the Test Harness

Input: SetDeviceStatusInput

Part: payload

Element: SetDeviceStatusRequest

Parameter	Description
id	The ID for the meter for which the device status should be set
value	the new device status to set on the meter

Output: SetDeviceStatusOutput

Part: payload

Element: SetDeviceStatusResult

The boolean response indicates the success or failure of the update (not the current field status).

Fault: See [UtilityFault](#). Thrown when meter id is not found.

GetConnectionStatus

This web service queries whether the given meter is connected or disconnected. This method is used by the Connect/Disconnect service. The values for connectionStatusType are:

- Connected
- Disconnected
- Unknown

Input: GetConnectionStatusInput

Part: payload

Element: GetConnectionStatusRequest

Parameter	Description
id	The ID for the meter for which the connection status should be retrieved

Output: GetConnectionStatusOutput

Part: payload

Element: GetConnectionStatusResult

The connection status of the requested meter.

Fault: See [UtilityFault](#). Thrown when meter id is not found.

SetConnectionStatus

This web service updates the load action code for a given meter. This method is used by the Connect/Disconnect service. The values for connectionStatusType are:

- Connected
- Disconnected
- Unknown

Input: SetConnectionStatusInput

Part: payload

Element: SetConnectionStatusRequest

Parameter	Description
id	The ID for the meter for which the connection status should be set.
value	The new value of LoadActionCode to set on the meter.

Output: SetConnectionStatusOutput

Part: payload

Element: SetConnectionStatusResult

The boolean response indicates the success or failure of the update (not the current field status).

Fault: See [UtilityFault](#). Thrown when meter id is not found.

IsCommissioned

This web service queries the commissioning status for a given meter. This service is used by the Commission/Decommission process. The commissioning attribute can be true or false.

Input: IsCommissionedInput

Part: payload

Element: IsCommissionedRequest

Parameter	Description
id	The ID for the meter for which the Commissioned status should be retrieved

Output: IsCommissionedOutput

Part: payload

Element: IsCommissionedResult

The value of the Commissioned status attribute for the requested meter.

Fault: See [UtilityFault](#). Thrown when meter id is not found.

SetCommissioned

This web service updates the commissioning status for a given meter. This service is used by the Commission/Decommission process. The commissioning attribute can be true or false.

Input: SetCommissionedInput

Part: payload

Element: SetCommissionedRequest

Parameter	Description
id	The ID for the meter for which the Commissioned status should be set
value	The new value of Commissioned status to set on the meter

Output: SetCommissionedOutput

Part: payload

Element: SetCommissionedResult

The boolean response indicates the success or failure of the update (not the current field status).

Fault: See [UtilityFault](#). Thrown when meter id is not found.

GetExecutionStatus

This web service queries the status of the property controlling the overall execution of the command. The possible values of execution status are:

Status	Description
Success	The command should complete successfully
ResponseTimeout	The asynchronous response will never arrive
SyncOperationFail	A simulated fault will occur in the during the initial request
AsyncOperationFailure	A simulated fault will occur in the asynchronous response

Input: GetExecutionStatusInput

Part: payload

Element: GetExecutionStatusRequest

Parameter	Description
id	The ID for the meter for which the ExecutionStatus should be retrieved

Output: GetExecutionStatusOutput

Part: payload

Element: GetExecutionStatusResult

The value of the ExecutionStatus attribute for the requested meter.

Fault: See [UtilityFault](#). Thrown when meter id is not found.

SetExecutionStatus

This web service updates the property controlling the overall completion of the command. The possible values of execution status are:

- Success - The command should complete successfully
- ResponseTimeout - The asynchronous response will never arrive
- SyncOperationFail - A simulated fault will occur in the during the initial request
- AsyncOperationFailure - A simulated fault will occur in the asynchronous response

Input: SetExecutionStatusInput

Part: payload

Element: SetExecutionStatusRequest

Parameter	Description
id	The ID for the meter for which the ExecutionStatus should be set
value	The new value of ExecutionStatus to set on the meter

Output: SetExecutionStatusOutput

Part: payload

Element: SetExecutionStatusResult

The boolean response indicates the success or failure of the update (not the current field status).

Fault: See [UtilityFault](#). Thrown when meter id is not found.