INTEGRATION GUIDE

RNI 4.14 MultiSpeak v3.0 Gateway

SECTION 2: METER READING

Copyright

Sensus
637 Davis Drive
Morrisville, NC 27560
1-800-638-3748
www.sensus.com

Document: MultiSpeak v3.0 Gateway Section 2: Meter Reading

Contents

2 Interface 2A (Meter Reading)

2.1 MRCB

2.1.1 Overview

This is a web service provided by the MR function on the RNI. It implements a subset of the methods from the MRCB service defined in the MultiSpeak v3.0 build J specification.

2.1.1.1 References

The MultiSpeak v3.0j MRCB server methods are described at
https://apps.cooperative.com/content/public/multispeak/30j/2A_MR_CB.asmx

2.1.1.2 Permissions

For a web service method to be executed in the gateway, the UserID in the request must have appropriate permissions in the user store (LDAP/AD). The following table lists the permissions needed for each method.

	Method	Permission
General	PingURL	[none]
	GetMethods	[none]
Meter Management[1]	GetAMRSupportedMeters	Get Meter
	MeterAddNotification	Add Meter, Update Meter
	MeterChangedNotification	Update Meter
	ServiceLocationChangedNotification	Update Meter
	MeterRemoveNotification	Update Meter
	IsAMRMeter	Get Meter
	InitiateDisconnectedStatus	Add Usage Monitoring
	CancelDisconnectedStatus	Delete Usage Monitoring
	InitiateUsageMonitoring	Add Usage Monitoring
	CancelUsageMonitoring	Delete Usage Monitoring
Meter Readings[1]	GetLatestReadingByMeterNo	Get Meter Read
	InitiateMeterReadByMeterNumber	Billing Ping

[1] Sensors/Snapshot readings are not supported by the Sensus MultiSpeak v3.0 gateway

2.1.1.3 Meter logical states

The **logical** state property complements the **lifecycle** state property and hence should be considered in relation to the lifecycle state. The currently supported logical states include Active, Inactive, and Disconnected, and are only relevant for an installed meter (i.e., a meter in the Install lifecycle state).

Figure 1 shows the valid transitions between logical states. In the MultiSpeak domain, the RNI is notified of a *move-out* with `InitiateUsageMonitoring`, a *move-in* with `CancelUsageMonitoring`, a supply side disconnect with an `InitiateDisconnectedStatus`, and a supply side connect with a `CancelDisconnectedStatus`.

At present, the logical states are only used for reporting and do not affect any other meter-related operations in the RNI.

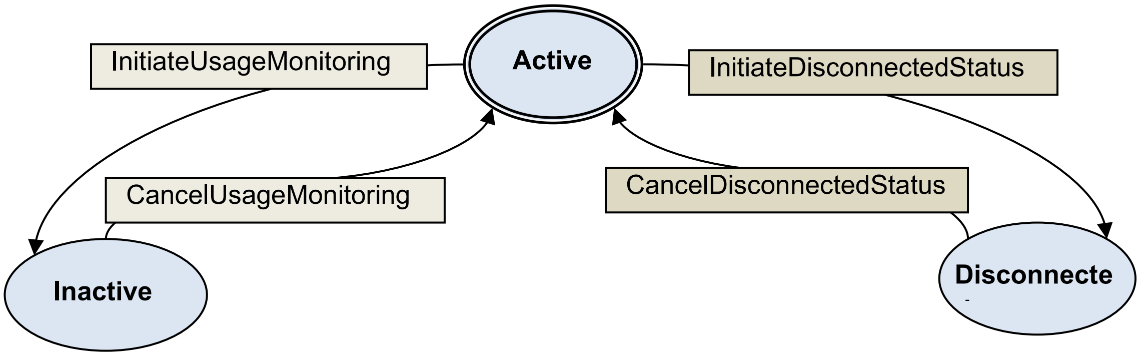

Figure 1: State diagram for meter logical state

The MultiSpeak gateway defaults to strict checking of state transitions and will reject requests that do not adhere to this state diagram (e.g., `InitiateUsageMonitoring` for a meter in a `Disconnected` logical state). However, this checking can be relaxed via system configuration.

2.1.2 PingURL

As described in *Section 1 – Overview*.

2.1.3 GetMethods

As described in *Section 1 – Overview*.

2.1.1 GetAMRSupportedMeters

This method returns the **installed** meters from the RNI DB. The number of meters to be returned in the response is configured on the RNI. This is a paged interface and uses the `LastSent` and `lastReceived` properties for paging.

Special considerations for a shared RNI
Since a single RNI can serve multiple customers (utilities), this request may specify a CustomerID attribute. Otherwise, the service uses a statically configured DefaultCustomerID property in the RNI. If neither the CustomerID attribute nor the DefaultCustomerID property is specified, then it is interpreted as a request to get AMR meters for all utilities permitted for the user. Even though the request does not specify a utility code, it only returns meters that the user is allowed to access based on the customer IDs associated with the user in LDAP.

2.1.1.1 Sample request

```
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:ver="http://www.multispeak.org/Version_3.0">
  <soapenv:Header>
    <ver:MultiSpeakMsgHeader
        Version="3.0"
        UserID="testuser" Pwd="testpass"
        AppName="myapp" AppVersion="1.5"
        Company="myco"/>
  </soapenv:Header>
  <soapenv:Body>
    <ver:GetAMRSupportedMeters>
      <ver:lastReceived>2999M</ver:lastReceived>
    </ver:GetAMRSupportedMeters>
  </soapenv:Body>
</soapenv:Envelope>
```

2.1.1.2 Sample response

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:ns2="http://www.multispeak.org/Version_3.0">
   <soapenv:Header>
       <ns2:MultiSpeakMsgHeader AppName="RNIGateway" AppVersion="1.0"
                      Company="Sensus" CSUnits="feet" SessionID="123"
                      LastSent="3501M"” ObjectsRemaining="0" />
   </soapenv:Header>
   <soapenv:Body>
      <ns2:GetAMRSupportedMetersResponse >
         <ns2:GetAMRSupportedMetersResult>
            <ns2:meter objectID="3000M">
               <ns2:deviceClass>01</ns2:deviceClass>
               <ns2:meterNo>3000M</ns2:meterNo>
               <ns2:nameplate>
                  <ns2:dials>0</ns2:dials>
                  <ns2:form>12S</ns2:form>
                  <ns2:transponderID>3000</ns2:transponderID>
               </ns2:nameplate>
            </ns2:meter>
            <ns2:meter objectID="3001M">
               <ns2:meterNo>3001M</ns2:meterNo>
               <ns2:nameplate>
                  <ns2:dials>2</ns2:dials>
                  <ns2:form>12S</ns2:form>
                  <ns2:transponderID>3001</ns2:transponderID>
               </ns2:nameplate>
            </ns2:meter>
            ................
          </ns2:GetAMRSupportedMetersResult>
      </ns2:GetAMRSupportedMetersResponse>
   </soapenv:Body>
</soapenv:Envelope>
```

2.1.1.3 Request parameters

	Attribute/Element	Usage
HEADER	Common request header attributes	
BODY	lastReceived	A tag indicating the end of the batch of a previous response. This should be the value of the LastSent header attribute value from the previous response. However, using the meterID of the last element from the previous response (legacy behavior) is also supported.

2.1.1.4 Response parameters

	Attribute or Element name	Usage
HEADER	Common response header attributes	
	ObjectsRemaining*	The number of meters remaining in the RNI at that point in time. This will be zero if this response contains the last available MeterID. Negative 1 is returned if there may be meters remaining in the RNI but the count cannot be determined.
	LastSent	A tag indicating the end of the batch in this response. The remote system should use this as the lastReceived value in the subsequent request if the ObjectsRemaining is not zero.
BODY	meter	Multiple elements, one per AMR meter that the requesting user is able to view.
	meter.objectID	The object ID for the meter element is the same as the meterNo.
	meter/meterNo*	The identifier for the meter assigned by the utility i.e., the MeterID. This is unique within the meter population managed by the RNI for a specific utility.
	utilityInfo/servLoc	The identifier for the service location with which the meter is associated.
	meter/deviceClass	The billing rate code for the meter i.e., the RateCode.
	meter/nameplate/dials	The dials for the meter.
	meter/nameplate/form	The form factor for the meter.
	meter/AMRType	Fixed value of SENS.
	meter/inServiceDate	Available for meter in “installed” state.
	meter/nameplate/transponderID*	The identifier for the radio endpoint on the meter (i.e., the FlexnetID). This is globally unique.

2.1.1.5 Error messages

	Error	Reasons
SYSTEM ERRORS	Common security errors	
BUSINESS ERRORS	Device not found	Indicates that the lastReceived value could not be correlated to a meter.
	Duplicate device exception	Indicates that the meter ID in the lastReceived maps to multiple meters. This is a rare condition which would only be seen if the LastReceivedIsMeterNo configuration is set and the requested meter number is used by two different meters in the RNI (a possibility for multi-service customers).

2.1.1.6 Configuration properties

	Property name	Usage
DB	Multispeak.MRCB.GetAMRSupportedMeters.MaxResults	The batch size for results returned by GetAMRSupportedMeters. This must be a positive number.
	DefaultCustomerID	The CustomerID for the utility that owns the meters in the RNI.
	Lifecycle.IncludeUnknownLifecycle (System configuration)	Boolean property to determine whether to include the meters with an Unknown lifecycle state in the response. Default is True.
FILE	LastReceivedIsMeterNo	Boolean property used to configure how the lastReceived provided in the GetAMRSupportedMeters request should be interpreted. If True, the last received will be interpreted as a Meter number. If False, the lastReceived is expected to be the LastSent value from the previous batch. Default is True.

2.1.1 IsAMRMeter

This method queries the RNI for whether a meter is known to be **installed**.

2.1.1.1 Sample request

```
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:ver="http://www.multispeak.org/Version_3.0">
  <soapenv:Header>
    <ver:MultiSpeakMsgHeader UserID="guestAdmin" Pwd="guest123" />
  </soapenv:Header>
  <soapenv:Body>
    <ver:IsAMRMeter>
      <ver:meterNo>1001M</ver:meterNo>
    </ver:IsAMRMeter>
  </soapenv:Body>
</soapenv:Envelope>
```

2.1.1.2 *Sample response*

This is the sample response for a meter that is installed in the RNI.

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
   <SOAP-ENV:Header>
      <MultiSpeakMsgHeader AppName="RNIGateway" AppVersion="3.1" CSUnits="feet" Company="Sensus"
xmlns="http://www.multispeak.org/Version_3.0"/>
   </SOAP-ENV:Header>
   <SOAP-ENV:Body>
      <ns2:IsAMRMeterResponse xmlns:ns2="http://www.multispeak.org/Version_3.0">
         <ns2:IsAMRMeterResult>true</ns2:IsAMRMeterResult>
      </ns2:IsAMRMeterResponse>
   </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

2.1.1.3 *Request parameters*

	Attribute/Element	Usage
HEADER	Common request header attributes	
BODY	meterNo*	The identifier for the installed meter assigned by the utility i.e., the MeterID.

2.1.1.4 *Response parameters*

	Attribute/Element	Usage
HEADER	Common response header attributes	
BODY	IsAMRMeterResult*	Boolean value true if the meter is installed, false otherwise.

2.1.1.5 Error messages

This response message definition does not have `ErrorObjects`. Hence any error is returned as a SOAP Fault. The messages below are in the `faultString` of the SOAP fault.

	Error	Reasons
BUSINESS ERRORS	Device not found	The RNI does not contain the specified meter ID.
	Problem accessing database	General database error.
SYSTEM ERRORS	Common security errors	

2.1.1.6 Configuration properties

	Property name	Usage
DB	Lifecycle.IncludeUnknownLifecycle (System configuration)	Boolean property to determine whether to include the meters with an Unknown lifecycle state in the response. Default is True.

2.1.2 MeterAddNotification

This method notifies the RNI of meters that have been added into the utility back-office systems. The resulting action is that **a meter record is created** in the RNI, **or** an inventoried **meter is marked as Installed**.

Any supported meter properties provided in the `MeterAddNotification` are saved to the RNI DB.

2.1.2.1 Sample request

```
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:ver="http://www.multispeak.org/Version_3.0">
  <soapenv:Header>
    <ver:MultiSpeakMsgHeader
        Version="3.0" UserID="testuser" Pwd="testpass"
        AppName="myapp" AppVersion="1.5" Company="myco"
        SessionID="123" />
  </soapenv:Header>
  <soapenv:Body>
    <ver:MeterAddNotification>
      <ver:addedMeters>
        <!--Zero or more repetitions:-->
        <ver:meter>
          <ver:deviceClass>1Z</ver:deviceClass>
          <ver:meterNo>3000M</ver:meterNo>
          <ver:nameplate>
            <ver:dials>4</ver:dials>
            <ver:form>2S</ver:form>
            <ver:transponderID>3000</ver:transponderID>
          </ver:nameplate>
          <ver:utilityInfo>
            <ver:servLoc>3000S</ver:servLoc>
          </ver:utilityInfo>
        </ver:meter>
        <ver:meter>
          <ver:meterNo>3001M</ver:meterNo>
          ...........
        </ver:meter>
        ...........
      </ver:addedMeters>
    </ver:MeterAddNotification>
  </soapenv:Body>
</soapenv:Envelope>
```

2.1.2.2 Sample response

This sample response shows that, of the two meters in the request above, one succeeded and the other resulted in an error because it is already an installed meter in the RNI.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:ns2="http://www.multispeak.org/Version_3.0">
   <soapenv:Header>
       <ns2:MultiSpeakMsgHeader AppName="RNIGateway" AppVersion="1.0"
                       Company="Sensus" CSUnits="feet"  SessionID="123"/>
   </soapenv:Header>
   <soapenv:Body>
      <ns2:MeterAddNotificationResponse >
         <ns2:MeterAddNotificationResult>
            <ns2:errorObject errorString="Invalid lifecycle state. Meter with flexNetId [ 3001 ]
meterId [ 3001M ] customerId [ ACME ] was expected to be in the [ Inventory ] meter lifecycle
state but was in the [ Install ] meter lifecycle state."
                              eventTime="2009-11-12T17:23:26.601-05:00"
                              nounType="Meter"
                              objectID="3001M"/>
         </ns2:MeterAddNotificationResult>
      </ns2:MeterAddNotificationResponse>
   </soapenv:Body>
</soapenv:Envelope>
```

2.1.2.3 Request parameters

	Attribute/Element	Usage
HEADER	Common request header attributes	
BODY	addedMeters	List of newly installed meters.
	meter/meterNo*	The identifier for the meter assigned by the utility (i.e., the MeterID).
	meter@utility	The utility to which the meter belongs. If present, this takes priority over the custom CustomerID attribute in the MultiSpeakMsgHeader.
	meter/nameplate/dials	The dials for the meter.
	meter/nameplate/form	The form factor for the meter.
	meter/nameplate/transponderID	This is the FlexNet ID for that meter. This is required only if the meter does not already exist in the RNI. If present this should be the same as the FlexNet ID for that meter.
	meter/utilityInfo/servLoc	Service location identifier associated with the meter.
	meter/deviceClass	The billing rate code for the meter (i.e., the RateCode).

2.1.2.4 Response parameters

	Attribute or Element name	Usage
HEADER	Common response header attributes	
BODY	errorObject	Multiple elements, one per failure to execute the add notification.
	errorObject.objectID*	The MeterID for the meter for which the add notification failed.
	errorObject.eventTime*	Timestamp of when the failure occurred on the server.
	errorObject.errorString*	Detailed (user friendly) error message.
	errorObject.nounType*	The type of the object indicated by objectID (e.g., Meter).

2.1.2.5 Error messages

	Error	Reasons
BUSINESS ERRORS	Invalid data for meter	One or more of the provided meter properties are not valid according to the validation rules in the RNI. This is also used to report an attempt to set a data value that overflows the field length constraints in the DB. This case may result in a partial update of the meter data in the DB. e.g., The value of the dials property in the request is constrained by the Gateway.MaxDials RNI property.
	Invalid lifecycle state	The meter is already installed in the RNI.
	Missing required property meterNo	The meterNo is needed to identify the meter being installed.
	Invalid rate code	The value of the deviceClass property does not match a rate code configured in the RNI.
	Missing FlexNetRadioID	For a new meter to be added to the RNI, the FlexNet ID must be provided as transponderId. This is required for communication with the meter.
	Database access failure	General database error.
	Service location ID [] exceeds maximum length of 25	Provided service location is too long.
	Spaces are not allowed in meter serial number.	Provided meterNo contains spaces and the RNI is configured not to allow spaces.
SYSTEM ERRORS	Common security errors	

2.1.2.6 *Configuration properties*

	Property name	Usage
DB	LifeCycle.AllowBypassInventoryState (System configuration)	Allow a meter to be installed directly in the Installed state. If set to False, then this method can only be used to transition a meter in from Inventory state to Install state. Default setting is True.
	Gateway.MaxDials	The maximum value for the dials property for a meter.
	Gateway.MaxRepId	The maximum possible value for the Transponder ID of a meter.

2.1.3 MeterChangedNotification

This method notifies the RNI of changes to the meter properties for an existing meter. The supported properties are updated in the RNI DB.

> The lifecycle state of a meter cannot be changed through this method. Please use MeterAddNotification or MeterRemoveNotification to change the state.

2.1.3.1 *Sample request*

Here is an example request to set meter properties. In this case the meter was previously added to the RNI, either through `MeterAddNotification` or some other means.

```
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:ver="http://www.multispeak.org/Version_3.0">
  <soapenv:Header>
    <ver:MultiSpeakMsgHeader
        Version="3.0" UserID="testuser" Pwd="testpass"
        AppName="myapp" AppVersion="1.5" Company="myco"
        SessionID="123" />
  </soapenv:Header>
  <soapenv:Body>
    <ver:MeterChangedNotification>
      <ver:changedMeters>
        <ver:meter>
          <ver:deviceClass>01</ver:deviceClass>
          <ver:meterNo>1002M</ver:meterNo>
          <ver:nameplate>
            <ver:dials>0</ver:dials>
            <ver:form>2S</ver:form>
          </ver:nameplate>
          <ver:utilityInfo>
            <ver:servLoc>ACME.Keystone12.W6</ver:servLoc>
          </ver:utilityInfo>
        </ver:meter>
        <ver:meter>
          <ver:meterNo>3001M</ver:meterNo>
          ...........
        </ver:meter>
        ...........
      </ver:changedMeters>
    </ver:MeterChangedNotification>
  </soapenv:Body>
</soapenv:Envelope>
```

2.1.3.2 Sample response

This sample response shows that, of the two meters in the request above, one succeeded and the other resulted in an error because it used a rate code that does not exist in the RNI.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:ns2="http://www.multispeak.org/Version_3.0">
   <soapenv:Header>
       <ns2:MultiSpeakMsgHeader AppName="RNIGateway" AppVersion="1.0"
                       Company="Sensus" CSUnits="feet"  SessionID="123"/>
   </soapenv:Header>
   <soapenv:Body>
      <ns2:MeterChangedNotificationResponse >
         <ns2:MeterChangedNotificationResult>
            <ns2:errorObject errorString=" Invalid Rate Code [ 01 ] for meter]. Please see log
file for more details”
                              eventTime="2009-11-14T17:13:22.501-05:00"
                              nounType="Meter”
                              objectID="3001M"/>
         </ns2:MeterChangedNotificationResult >
      </ns2:MeterChangedNotificationResponse >
   </soapenv:Body>
</soapenv:Envelope>
```

2.1.3.3 Request parameters

Same as `MeterAddNotification`.

2.1.3.4 Response parameters

Same as `MeterAddNotification`.

2.1.3.5 Error messages

Same as `MeterAddNotification`.

2.1.3.6 Configuration properties

Same as `MeterAddNotification`.

2.1.4 MeterRemoveNotification

This method notifies the RNI of a meter that has been uninstalled. The lifecycle state of the meter in the RNI will be changed from *Install* to *Inventory* to indicate that the meter is no longer installed.

The MultiSpeak standard allows meter properties to be specified in this request and the RNI gateway will update supported meter properties along with the lifecycle state change. However, it is unusual for properties to be updated during an uninstall operation. When the meter is removed, the service location id is reset to null so that old service locations are not retained when added back into service even if servLoc is specified in the MeterRemoveNotification.

2.1.4.1 Sample request

```
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:ver="http://www.multispeak.org/Version_3.0">
  <soapenv:Header>
    <ver:MultiSpeakMsgHeader
```

```
        Version="3.0" UserID="testuser" Pwd="testpass"
        AppName="myapp" AppVersion="1.5" Company="myco"
        SessionID="123" />
  </soapenv:Header>
  <soapenv:Body>
    <ver:MeterRemoveNotification>
      <ver:removedMeters>
        <ver:meter objectID="3000M">
          <ver:meterNo>3000M</ver:meterNo>
        </ver:meter>
        <ver:meter objectID="3001M">
          <ver:meterNo>3001M</ver:meterNo>
        </ver:meter>
        ...........
      </ver:removedMeters >
    </ver:MeterRemoveNotification >
  </soapenv:Body>
</soapenv:Envelope>
```

2.1.4.2 Sample response

This sample response shows that of the two meters in the request above, one succeeded and the other resulted in an error because it is not installed.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:ns2="http://www.multispeak.org/Version_3.0">
   <soapenv:Header>
       <ns2:MultiSpeakMsgHeader AppName="RNIGateway" AppVersion="1.0"
                       Company="Sensus" CSUnits="feet" SessionID="123"/>
   </soapenv:Header>
   <soapenv:Body>
      <ns2:MeterRemoveNotificationResponse >
         <ns2:MeterRemoveNotificationResult>
            <ns2:errorObject errorString=" Invalid lifecycle state. Meter with flexNetId [ 3001 ]
meterId [ 3001M ] customerId [ ACME ] was expected to be in the [ Install ] meter lifecycle state
but was actually in the [ Inventory ] meter lifecycle state. Please see log file for more
details.”
                              eventTime="2009-11-14T17:13:22.501-05:00"
                              nounType="Meter"
                              objectID="3001M"/>
         </ns2:MeterRemoveNotificationResult>
      </ns2:MeterRemoveNotificationResponse>
   </soapenv:Body>
</soapenv:Envelope>
```

2.1.4.3 Request parameters

Same as `MeterAddNotification`.

2.1.4.4 Response parameters

Same as `MeterAddNotification`.

2.1.4.5 Error messages

Same as `MeterAddNotification`, except that the *Invalid lifecycle state* error is reported if the meter is not in Install state.

2.1.4.6 Configuration properties

Same as `MeterAddNotification`.

2.1.5 ServiceLocationChangedNotification

This method notifies the RNI of a change in a service location (e.g., address changes).

> Multiple meter records may be associated with the same service location. In this case a change to the properties of service location would affect multiple meters.

This notification leads to the creation of a new service location if one does not exist.

2.1.5.1 Sample request

```
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:ver="http://www.multispeak.org/Version_3.0">
  <soapenv:Header>
    <ver:MultiSpeakMsgHeader
        Version="3.0" UserID="testuser" Pwd="testpass"
        AppName="myapp" AppVersion="1.5" Company="myco"
        SessionID="123"/>
  </soapenv:Header>
  <soapenv:Body>
    <ver:ServiceLocationChangedNotification>
      <ver:changedServiceLocations>
        <ver:serviceLocation objectID="3000S">
          <ver:facilityName>FacilityName</ver:facilityName>
          <ver:servAddr1>400 Perimeter Park</ver:servAddr1>
          <ver:servAddr2>Suite K</ver:servAddr2>
          <ver:servCity>Morrisville</ver:servCity>
          <ver:servState>NC</ver:servState>
          <ver:servZip>27560</ver:servZip>
          <ver:billingCycle>B1</ver:billingCycle>
        </ver:serviceLocation>
      </ver:changedServiceLocations>
    </ver:ServiceLocationChangedNotification>
  </soapenv:Body>
</soapenv:Envelope>
```

2.1.5.2 Sample response

This sample response shows an update that failed because the provided service location ID did not exist in the RNI.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:ns2="http://www.multispeak.org/Version_3.0">
   <soapenv:Header>
       <ns2:MultiSpeakMsgHeader AppName="RNIGateway" AppVersion="1.0"
                      Company="Sensus" CSUnits="feet"  SessionID="123"/>
   </soapenv:Header>
   <soapenv:Body>
      <ns2:ServiceLocationChangedNotificationResponse>
         <ns2:ServiceLocationChangedNotificationResult>
               <ns2:errorObject errorString="Service Location with servLocId [
       NO_SUCH_SERVICELOCATION ] from customerId [ ACME ] does not exist in the database."
       eventTime="2009-11-30T09:46:58.843-05:00" nounType="ServiceLocation" objectID="3000S"/>
         </ns2:ServiceLocationChangedNotificationResult>
      </ns2:ServiceLocationChangedNotificationResponse>
   </soapenv:Body>
</soapenv:Envelope>
```

2.1.5.3 Request parameters

	Attribute/Element	Usage
HEADER	Common request header attributes	
BODY	serviceLocation/objectID*	The service location ID associated with a meter
	serviceLocation/servAddr1	Address line 1
	serviceLocation/servAddr2	Address line 2
	serviceLocation/servCity	City
	serviceLocation/servState	State
	serviceLocation/servZip	Zip
	serviceLocation/facilityName	Facility name
	serviceLocation/billingCycle	The billing cycle for meter(s) at this service location

2.1.5.4 Response parameters

	Attribute or Element name	Usage
HEADER	Common response header attributes	
BODY	errorObject	Multiple elements, one per failure to execute the change notification
	errorObject.objectID*	The service location ID for which the update failed
	errorObject.eventTime*	Timestamp of when the failure occurred on the server
	errorObject.errorString*	Detailed (user friendly) error message
	errorObject.nounType*	ServiceLocation

2.1.5.5 Error messages

	Error	Reasons
BUSINESS ERRORS	Missing objectID attribute	The objectID is needed to identify the service location being installed.
	Data validation failure	One or more of the provided service location properties are not valid according to the validation rules in the RNI. This is also used to report an attempt to set a data value that overflows the field length constraints in the DB. This case may result in a partial update of the meter data in the DB. For example, the value of the state property is constrained to be a two-character code, so an attempt to pass in a full state name will cause an invalid data error to be reported. Note: The validation of the two-letter state abbreviation is not supported.
	Problem accessing database	General database error.
	Service location ID [] exceeds maximum length of 25	Provided service location is too long.

	Error	Reasons
	Spaces are not allowed in meter serial number.	Provided meterNo contains spaces and the RNI is configured not to allow spaces.
SYSTEM ERRORS	Common security errors	

2.1.5.6 *Configuration properties*

	Property name	Usage
DB	Gateway.MaxCycle (System configuration)	The maximum value for the billing cycle.

2.1.6 InitiateMeterReadByMeterNumber

This method requests the RNI to get updated meter readings for a meter. The RNI sends out an on-air message to the meter and responds with no errors if the message was transmitted. When the meter responds to the RNI with a meter reading, a `ReadingChangedNotification` is published.

The RNI supports two different modes for this operation.

(a) In **on-the-glass mode**, the meter is pinged for its instantaneous reading. This is the preferred mode and the default out-of-the-box setting. The behavior is consistent across different meter types and configurations. Gas and water devices are supported only when using on-the-glass reading mode. The instantaneous readings are not time-aligned or written to the DB. Therefore, they cannot be retrieved by any of the GetXXX methods for meter reads.

(b) In **supervisory reading mode**, the RNI will ping the meter for its time-aligned billing read. This behavior varies across meter types and configurations and hence is not recommended for use as the default behavior. These reads are stored in the DB and can be retrieved through any of the GetXXX methods for meter reads.

The mode for this operation can be specified per-request by using the `Profile` attribute.

> There are additional differences between the modes that may be of interest:
>
> Gas and water devices are supported only in on-the-glass reading mode.
>
> TOU reads are retrieved only in supervisory reading mode. This feature will be added to on-the-glass mode in a future release.

2.1.6.1 *Sample request*

Here is a sample request to initiate a meter read for four meters.

```
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:ver="http://www.multispeak.org/Version_3.0"
    xmlns:sensus="http://www.sensus.com/multispeak3">
  <soapenv:Header>
    <ver:MultiSpeakMsgHeader
        Version="3.0" UserID="testuser" Pwd="testpass"
        AppName="myapp" AppVersion="1.5" Company="myco"
        SessionID="123" sensus:Profile="onTheGlass"/>
  </soapenv:Header>
  <soapenv:Body>
    <ver:InitiateMeterReadByMeterNumber>
      <ver:meterNos>
        <!-- IconA METER ID-->
        <ver:string>3001M</ver:string>
        <!-- IconA METER ID-->
        <ver:string>3002M</ver:string>
        <!-- Elster A3 METER ID-->
        <ver:string>4001M</ver:string>
        <!-- Icon METER ID-->
        <ver:string>2001M</ver:string>
      </ver:meterNos>
    </ver:InitiateMeterReadByMeterNumber>
  </soapenv:Body>
</soapenv:Envelope>
```

2.1.6.2 *Sample response*

Here is the response for the sample request made in *2.1.6.1 Sample Request* showing that the request succeeded for three meters and failed for one.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:ns2="http://www.multispeak.org/Version_3.0">
   <soapenv:Header>
       <ns2:MultiSpeakMsgHeader AppName="RNIGateway" AppVersion="1.0"
                      Company="Sensus" CSUnits="feet" SessionID="123"/>
   </soapenv:Header>
   <soapenv:Body>
      <ns2:InitiateMeterReadByMeterNumberResponse>
         <ns2:InitiateMeterReadByMeterNumberResult>
        <ns2:errorObject errorString="DEVICE_NOT_ACTIVE"
               eventTime="2009-11-01T16:33:32.245-05:00"
               nounType="Meter" objectID="3001M"/>
         </ns2:InitiateMeterReadByMeterNumberResult>
      </ns2:InitiateMeterReadByMeterNumberResponse>
   </soapenv:Body>
</soapenv:Envelope>
```

2.1.6.3 *Request parameters*

	Attribute/Element	Usage
HEADER	Common request header attributes	
	Profile	The meter reading profile. This should be a user defined profile. This attribute should be specified in the Sensus namespace.
BODY	meterNos	List of meter numbers (i.e., Meter IDs) to be polled for up-to-date readings.

2.1.6.4 Response parameters

	Attribute/Element	Usage
HEADER	Common response header attributes	
BODY	errorObject	Multiple elements, one per failed meter
	errorObject.objectID*	The MeterID for the meter for which the operation failed
	errorObject.eventTime*	Timestamp of when the failure occurred on the server
	errorObject.errorString*	Detailed (user friendly) error message
	errorObject.nounType*	Meter

2.1.6.5 Error messages

	Error	Reasons
BUSINESS ERRORS	Device not found	The specified meterID was not found in the RNI.
	Unsupported device type	A device type is not configured or inferred for the meter. This could be because the meter was added to the RNI without a device type and the meter has not reported in yet.
	Unsupported message type	The meter does not support the billing ping operation or support for it is not (yet) in the RNI.
	Database access failure	General database error.
	Unable to find specified profile (<profilename>) in the profile map.	The meter reading profile specified in the request is not valid. This is most likely seen when there is a typo in the profile name either in the request or in the RNI profile configuration.
	Could not retrieve meter read information from the meter response. The meter may be experiencing a power fail.	An on-the-glass mode request was sent while the meter is powering down so meter readings could not be retrieved.
	No billing data found [Billing Cycle or Rate Code]	No billing cycle or rate code was found for the meter. Correct the problem by adding a service location or rate code to the meter.
SYSTEM ERRORS	Common security errors	

2.1.6.6 Configuration properties

	Property name	Usage
DB	Multispeak.MRCB.ForceTOU (Customer configuration)	Boolean property to determine whether to force including the tier reads for a meter or let the rate code determine if they are returned. If True, tier reads are included regardless of the rate code (assuming the meter is reporting tier reads). If False, tier reads are only included if an appropriate TOU rate code has been assigned in the RNI for this meter. Default is True.
FILE	toudata.includeTouData (System configuration)	System wide switch for TOU data reporting through the MultiSpeak gateway. It is unlikely that this will ever be false in a production setting. Default is true.
	ms3.mr.initiateReading.DefaultProfile	Profile name to use for InitiateMeterReadingsByMeterNumber to specify reading mode, ping priority and total time allowed if one is not specified in the header. Default is onTheGlass. Note: As of RNI 4.0 and later, this property has been renamed to ms3.mr.initiateReading.defaultProfile from mrDefaultProfile. Changes to this property must be made in /opt/flexnet/conf/flexnet.local.properties on the web host.

2.1.7 GetLatestReadingByMeterNo

This is a request to get the **most recent** meter reading that is available for a meter. The RNI returns the most recent readings from its database.

This method can get the register as well as TOU readings if the meter is set up to report TOU read data to the RNI.

2.1.7.1 Sample request

```
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:ver="http://www.multispeak.org/Version_3.0">
  <soapenv:Header>
    <ver:MultiSpeakMsgHeader
        Version="3.0" UserID="testuser" Pwd="testpass"
        AppName="myapp" AppVersion="1.5" Company="myco"
        SessionID="123"/>
  </soapenv:Header>
  <soapenv:Body>
    <ver:GetLatestReadingByMeterNo>
      <ver:meterNo>3001M</ver:meterNo>
    </ver:GetLatestReadingByMeterNo>
  </soapenv:Body>
</soapenv:Envelope>
```

2.1.7.2 Sample response

2.1.7.2.1 For a non-TOU electric meter

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:ns2="http://www.multispeak.org/Version_3.0">
   <soapenv:Header>
       <ns2:MultiSpeakMsgHeader AppName="RNIGateway" AppVersion="1.0"
                       Company="Sensus" CSUnits="feet" SessionID="123"/>
   </soapenv:Header>
   <soapenv:Body>
         <ns2:GetLatestReadingByMeterNoResult>
            <ns2:meterNo>15144M</ns2:meterNo>
            <ns2:deviceID>15144</ns2:deviceID>
            <ns2:readingDate>2013-04-01T17:15:00.000Z</ns2:readingDate>
            <ns2:posKWh>55962</ns2:posKWh>
            <ns2:negKWh>5630</ns2:negKWh>
            <ns2:kW>4.96</ns2:kW>
            <ns2:readingValues>
               <ns2:readingValue>
                  <ns2:units>kW</ns2:units>
                  <ns2:value>4.96</ns2:value>
                  <ns2:name>Demand</ns2:name>
               </ns2:readingValue>
               <ns2:readingValue>
                  <ns2:units>Wh</ns2:units>
                  <ns2:value>5630000</ns2:value>
                  <ns2:name>Reverse</ns2:name>
               </ns2:readingValue>
               <ns2:readingValue>
                  <ns2:units>Wh</ns2:units>
                  <ns2:value>55962000</ns2:value>
                  <ns2:name>Consumption</ns2:name>
               </ns2:readingValue>
            </ns2:readingValues>
         </ns2:GetLatestReadingByMeterNoResult>
      </ns2:GetLatestReadingByMeterNoResponse>
   </soapenv:Body>
</soapenv:Envelope>
```

2.1.7.2.2 For a gas meter

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:ns2="http://www.multispeak.org/Version_3.0">
   <soapenv:Header>
       <ns2:MultiSpeakMsgHeader AppName="RNIGateway" AppVersion="1.0"
                       Company="Sensus" CSUnits="feet" SessionID="123"/>
   </soapenv:Header>
   <soapenv:Body>
       <ns2:GetLatestReadingByMeterNoResponse>
         <ns2:GetLatestReadingByMeterNoResult>
            <ns2:meterNo>RepId:22000051</ns2:meterNo>
            <ns2:deviceID>22000051</ns2:deviceID>
            <ns2:readingDate>2011-09-22T10:15:00.000-04:00</ns2:readingDate>
            <ns2:readingValues>
               <ns2:readingValue>
                  <ns2:units>m3</ns2:units>
                  <ns2:value>159.0</ns2:value>
                  <ns2:name>UncorrGasVol</ns2:name>
               </ns2:readingValue>
            </ns2:readingValues>
         </ns2:GetLatestReadingByMeterNoResult>
      </ns2:GetLatestReadingByMeterNoResponse>
   </soapenv:Body>
</soapenv:Envelope>
```

2.1.7.2.3 For an electric meter with TOU

```xml
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:ns2="http://www.multispeak.org/Version_3.0">
   <soapenv:Header>
       <ns2:MultiSpeakMsgHeader AppName="RNIGateway" AppVersion="1.0"
                      Company="Sensus" CSUnits="feet" SessionID="123"/>
   </soapenv:Header>
   <soapenv:Body>
      <ns2:GetLatestReadingByMeterNoResponse>
         <ns2:GetLatestReadingByMeterNoResult>
            <ns2:meterNo>TEG010079468</ns2:meterNo>
            <ns2:deviceID>2452683</ns2:deviceID>
               <ns2:readingDate>2009-12-29T14:15:00.000-05:00</ns2:readingDate>
            <ns2:TOUReadings>
               <ns2:TOUReading>
                  <ns2:extensionsList>
                     <ns2:extensionsItem>
                        <ns2:extName>kWh F</ns2:extName>
                        <ns2:extValue>0.0</ns2:extValue>
                        <ns2:extType>double</ns2:extType>
                     </ns2:extensionsItem>
                     <ns2:extensionsItem>
                        <ns2:extName>kW F</ns2:extName>
                        <ns2:extValue>0.0010</ns2:extValue>
                        <ns2:extType>double</ns2:extType>
                     </ns2:extensionsItem>
                     <ns2:extensionsItem>
                        <ns2:extName>DemandTime F</ns2:extName>
                        <ns2:extValue>2009-12-03T01:59:00.000-05:00</ns2:extValue>
                        <ns2:extType>dateTime</ns2:extType>
                     </ns2:extensionsItem>
                  </ns2:extensionsList>
                  <ns2:ratePeriod>1</ns2:ratePeriod>
                  <ns2:kWh>0</ns2:kWh>
                  <ns2:kW>0.0010</ns2:kW>
                  <ns2:kWDateTime>2009-12-03T01:59:00.000-05:00</ns2:kWDateTime>
               </ns2:TOUReading>
               <ns2:TOUReading>
                  <ns2:extensionsList>
                     <ns2:extensionsItem>
                        <ns2:extName>kWh F</ns2:extName>
                        <ns2:extValue>0.0040</ns2:extValue>
                        <ns2:extType>double</ns2:extType>
                     </ns2:extensionsItem>
                     <ns2:extensionsItem>
                        <ns2:extName>kW F</ns2:extName>
                        <ns2:extValue>0.0010</ns2:extValue>
                        <ns2:extType>double</ns2:extType>
                     </ns2:extensionsItem>
                     <ns2:extensionsItem>
                        <ns2:extName>DemandTime F</ns2:extName>
                        <ns2:extValue>2009-12-03T01:59:00.000-05:00</ns2:extValue>
                        <ns2:extType>dateTime</ns2:extType>
                     </ns2:extensionsItem>
                  </ns2:extensionsList>
                  <ns2:ratePeriod>0</ns2:ratePeriod>
                  <ns2:kWh>0</ns2:kWh>
                  <ns2:kW>0.0010</ns2:kW>
                  <ns2:kWDateTime>2009-12-03T01:59:00.000-05:00</ns2:kWDateTime>
               </ns2:TOUReading>
               <ns2:TOUReading>
                  <ns2:extensionsList>
                     <ns2:extensionsItem>
                        <ns2:extName>kWh F</ns2:extName>
                        <ns2:extValue>0.0010</ns2:extValue>
                        <ns2:extType>double</ns2:extType>
                     </ns2:extensionsItem>
                     <ns2:extensionsItem>
                        <ns2:extName>kW F</ns2:extName>
                        <ns2:extValue>0.0010</ns2:extValue>
                        <ns2:extType>double</ns2:extType>
                     </ns2:extensionsItem>
                     <ns2:extensionsItem>
                        <ns2:extName>DemandTime F</ns2:extName>
                        <ns2:extValue>2009-11-25T04:29:00.000-05:00</ns2:extValue>
                        <ns2:extType>dateTime</ns2:extType>
                     </ns2:extensionsItem>
                  </ns2:extensionsList>
                  <ns2:ratePeriod>3</ns2:ratePeriod>
                  <ns2:kWh>0</ns2:kWh>
                  <ns2:kW>0.0010</ns2:kW>
                  <ns2:kWDateTime>2009-11-25T04:29:00.000-05:00</ns2:kWDateTime>
               </ns2:TOUReading>
            </ns2:TOUReadings>
            <ns2:readingValues/>
         </ns2:GetLatestReadingByMeterNoResult>
      </ns2:GetLatestReadingByMeterNoResponse>
   </soapenv:Body>
</soapenv:Envelope>
```

2.1.7.3 Request parameters

	Attribute/Element	Usage
HEADER	Common request header attributes	
BODY	meterNo	The meter number (i.e., Meter ID).

2.1.7.4 Response parameters

	Attribute or Element name	Usage
HEADER	Common response header attributes	
BODY	meterNo*	Meter number (i.e., Meter ID) of the meter whose reading is being returned.
	deviceId	The FlexNet ID (i.e., Radio endpoint ID).
	readingDate	The sample point for this reading.
	kW	The peak demand since the last demand reset. This is only populated for a non-TOU configured meter.
	posKWh	The cumulative energy consumption. This may be F, F+R or F-R depending on the meter setup. This is only populated for a non-TOU configured meter.
	negKWh	The reverse energy consumption if the meter is configured to report it separately. This is only populated for a non-TOU configured meter. Note, this does not indicate the value is in units Kwh; Units information is conveyed in the readingValue.

	Attribute or Element name	Usage
	readingValues	Collection of reading values. This is only populated for a non-TOU configured meter.
	readingValue.name*	Type of reading will be one of {Consumption, Reverse, Demand,totalKWh,netKWh}.
	readingValue.units*	Units of measure (UOM) for this reading e.g., kW, kWh, kVA, kVAr, Wh, etc.
	readingValue.value*	Reading value: For Consumption, this is the cumulative usage. If the corresponding units are kWh, then this value is the same as the posKWh element. For Reverse, this is the reverse energy usage. If the corresponding units are kWh, then this value is the same as the negKWh element. For Demand, this is the Peak Demand since the last demand reset. If the corresponding units are kW, then this value is the same as the kW element.
	TOUReadings	Collection of TOU readings, one per tier. This is only populated for a TOU-configured meter.
	TouReading.ratePeriod*	Numeric value of the TOU tier being reported. The value 0 indicates the totals across all the other tiers.
	TouReading.kWh	The Consumption in kWh. This is only available if the meter reports consumption in kWh.
	TouReading.kW	The Peak Demand in kW since the last demand reset. This is only reported if the meter reports peak demand in kW.
	TOUReading.kWDateTime	The date-time of when the peak demand occurred. This is present only if TOUReading.kW is present.
	TOUReading.extensionsList	Sensus-specific extensions that allow TOU readings to be reported in units other than kW and kWh.
	TOUReading.extensionsItem	One of Consumption, Demand, and Demand Time: If the consumption is in kWh, then this has the same value as TouReading.kWh. If the peak demand is in kW, then this has the same value as TouReading.kW. If the demand is in kW, then DemandTime has the same value as TouReading.kWDateTime.
	TOUReading.extName	The UOM for a consumption (e.g., kWh, kVah) or demand (e.g., kW, kVAR) or the constant DemandTime suffixed with the net flow type e.g., F (Forward), R (Reverse).
	TOUReading.extValue	Value corresponding to the extName.
	TOUReading.extType	The data type of the extValue. This is double if the extName is a UOM and dateTime for DemandTime.

2.1.7.5 Error messages

This response message definition does not contain error objects. Therefore, errors are returned using the `errorString` attribute of the `GetLatestReadingByMeterResult`.

	Error	Reasons
BUSINESS ERRORS	Duplicate entries found in database with same meter identity but different FlexNet Radio IDs.	The RNI has two meters discovered with the same meter identity, but different FlexNet ids. This makes any request based on the meter identity ambiguous. Correct this problem by removing the duplicate entry.
	Unsupported serviceType	The serviceType value is not supported by the RNI. The currently supported service types are Electric, Gas and Water.
	Unable to determine service type	The service type of the given meter in the database in invalid. The service type of the meter needs to be corrected before it can be used for this operation.
SYSTEM ERRORS	Common security errors	

2.1.7.6 Configuration properties

	Property name	Usage
DB	DefaultCustomerID	The CustomerID for the utility that owns the meters in the RNI.
	Multispeak.MRCB.ForceTOU (Customer configuration)	Boolean property to determine whether to force including the tier reads for a meter or let the rate code determine if they are returned. If True, tier reads are included regardless of the rate code (assuming the meter is reporting tier reads). If False, tier reads are only included if an appropriate TOU rate code has been assigned in the RNI for this meter. Default is True.
FILE	toudata.includeTouData (System configuration)	System-wide switch for TOU data reporting through the MultiSpeak gateway. It is unlikely that this will ever be false in a production setting. Default is True.

2.1.8 InitiateUsageMonitoring

This method notifies the RNI of meters where **zero usage** is expected (i.e., **move out**). The RNI will set the *logical* state for the meter record from Active to Inactive. This does not disable any of the monitoring for this meter in the RNI.

2.1.8.1 Sample request

```
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:ver="http://www.multispeak.org/Version_3.0">
  <soapenv:Header>
    <ver:MultiSpeakMsgHeader
        Version="3.0" UserID="testuser" Pwd="testpass"
        AppName="myapp" AppVersion="1.5" Company="myco"
        SessionID="123"/>
  </soapenv:Header>
  <soapenv:Body>
    <ver:InitiateUsageMonitoring>
      <ver:meterNos>
```

```
        <ver:string>3000M</ver:string>
        <ver:string>3001M</ver:string>
        <ver:string>3002M</ver:string>
        <ver:string>3003M</ver:string>
      </ver:meterNos>
    </ver:InitiateUsageMonitoring>
  </soapenv:Body>
</soapenv:Envelope>
```

2.1.8.2 *Sample response*

Here is the response corresponding to the sample request above. This shows that of the 4 meters in the request, the operation succeeded for 2 and failed for 2, one because it was not in the Install lifecycle state and the other because it was not in the Active logical state.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:ns2="http://www.multispeak.org/Version_3.0">
   <soapenv:Header>
       <ns2:MultiSpeakMsgHeader AppName="RNIGateway" AppVersion="1.0"
                       Company="Sensus" CSUnits="feet"  SessionID="123"/>
   </soapenv:Header>
   <soapenv:Body>
      <ns2:InitiateUsageMonitoringResponse>
         <ns2:InitiateUsageMonitoringResult>
            <ns2:errorObject errorString="Invalid lifecycle state" eventTime="2012-12-
06T23:01:59.616-05:00" nounType="Meter" objectID="3000M"/>
            <ns2:errorObject errorString="Invalid logical state" eventTime="2012-12-
06T23:01:59.850-05:00" nounType="Meter" objectID="3001M"/>
         </ns2:InitiateUsageMonitoringResult>
      </ns2:InitiateUsageMonitoringResponse>
   </soapenv:Body>
</soapenv:Envelope>
```

2.1.8.3 *Request parameters*

	Attribute/Element	Usage
HEADER	Common request header attributes	
BODY	meterNos	List of meter numbers (i.e., Meter IDs) to be put in zero-usage monitoring.

2.1.8.4 *Response parameters*

	Attribute or Element name	Usage
HEADER	Common response header attributes	
BODY	errorObject	Multiple elements, one per failed meter
	errorObject.objectID*	The MeterID for the meter for which the operation failed
	errorObject.eventTime*	Timestamp of when the failure occurred on the server

	Attribute or Element name	Usage
	errorObject.errorString*	Detailed (user friendly) error message
	errorObject.nounType*	Meter

2.1.8.5 *Error messages*

	Error	Reasons
BUSINESS ERRORS	Invalid lifecycle state	The meter is not yet in the “Install” state. The MeterAddNotification can be used to notify the gateway that the meter has been installed.
	Invalid logical state	The meter has a logical state for which this operation is not supported. The most common reasons for this are InitiateUsageMonitoring was previously invoked for this meter but never cancelled so the meter is already in the “Inactive” logical state. InitiateDisconnectedStatus was previously invoked for this meter but never cancelled so the meter is in the “Disconnected” logical state.
	Device not found	The specified meterID was not found in the RNI.
SYSTEM ERRORS	Common security errors	

2.1.8.6 *Configuration properties*

	Property name	Usage
FILE	usagemonitoring.statechange.enforce	A flag to enforce the logical state transitions as described in section 2.1.1.3 Meter logical states. Default is true.

2.1.9 CancelUsageMonitoring

This method notifies the RNI of cancellation of zero usage monitoring (i.e., **move in**). It is the complement of `InitiateUsageMonitoring` and causes the logical state for the meter to be changed from *Inactive* to *Active*.

2.1.9.1 *Sample request*

```
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:ver="http://www.multispeak.org/Version_3.0">
  <soapenv:Header>
    <ver:MultiSpeakMsgHeader
        Version="3.0" UserID="testuser" Pwd="testpass"
        AppName="myapp" AppVersion="1.5" Company="myco"
        SessionID="123"/>
  </soapenv:Header>
  <soapenv:Body>
    <ver:CancelUsageMonitoring>
      <ver:meterNos>
        <ver:string>3000M</ver:string>
        <ver:string>3001M</ver:string>
        <ver:string>3002M</ver:string>
    <ver:string>3003M</ver:string>
   </ver:meterNos>
  </ver:CancelUsageMonitoring>
 </soapenv:Body>
</soapenv:Envelope>
```

2.1.9.2 Sample response

This shows a sample response with failure for two out of the four meters in the request.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:ns2="http://www.multispeak.org/Version_3.0">
   <soapenv:Header>
       <ns2:MultiSpeakMsgHeader AppName="RNIGateway" AppVersion="1.0"
                    Company="Sensus" CSUnits="feet"  SessionID="123"/>
   </soapenv:Header>
   <soapenv:Body>
      <ns2:CancelUsageMonitoringResponse>
         <ns2:CancelUsageMonitoringResult>
            <ns2:errorObject errorString="Invalid lifecycle state" eventTime="2012-12-07T09:31:35.533-05:00" nounType="Meter" objectID="3000M"/>
            <ns2:errorObject errorString="Invalid logical state" eventTime="2012-12-07T09:31:35.596-05:00" nounType="Meter" objectID="3001M"/>
         </ns2:CancelUsageMonitoringResult>
      </ns2:CancelUsageMonitoringResponse>
   </soapenv:Body>
</soapenv:Envelope>
```

2.1.9.3 Request parameters

Same as `InitiateUsageMonitoring`.

2.1.9.4 Response parameters

Same as `InitiateUsageMonitoring`.

2.1.9.5 Error messages

Same error conditions as for `InitiateUsageMonitoring`, with the following difference.

The most common reason for the Invalid logical state is that `InitiateUsageMonitoring` was not previously invoked for this meter, so this meter is not in an Inactive logical state.

2.1.9.6 Configuration properties

Same as `InitiateUsageMonitoring`.

2.1.10 InitiateDisconnectedStatus

This method *notifies* the RNI of meters that have been disconnected and no AMR reading is expected. The RNI changes the logical state of the meter to *Disconnected* in its data store. This method does not trigger a remote disconnect nor disable meter read collection or outage detection.

2.1.10.1 Sample request

```
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:ver="http://www.multispeak.org/Version_3.0">
```

```
  <soapenv:Header>
    <ver:MultiSpeakMsgHeader
        Version="3.0" UserID="testuser" Pwd="testpass"
        AppName="myapp" AppVersion="1.5" Company="myco"
        SessionID="123"/>
  </soapenv:Header>
  <soapenv:Body>
    <ver:InitiateDisconnectedStatus>
      <ver:meterNos>
        <ver:string>3000M</ver:string>
        <ver:string>3001M</ver:string>
        <ver:string>3002M</ver:string>
        <ver:string>3003M</ver:string>
      </ver:meterNos>
    </ver:InitiateDisconnectedStatus>
  </soapenv:Body>
</soapenv:Envelope>
```

2.1.10.2 Sample response

This shows a sample response with failures for two out of the four meters in the request.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:ns2="http://www.multispeak.org/Version_3.0">
   <soapenv:Header>
       <ns2:MultiSpeakMsgHeader AppName="RNIGateway" AppVersion="1.0"
                       Company="Sensus" CSUnits="feet"  SessionID="123"/>
   </soapenv:Header>
   <soapenv:Body>
      <ns2:InitiateDisconnectedStatusResponse>
         <ns2:InitiateDisconnectedStatusResult>
            <ns2:errorObject errorString="Invalid lifecycle state" eventTime="2012-12-
07T09:34:47.644-05:00" nounType="Meter" objectID="3000M"/>
            <ns2:errorObject errorString="Invalid logical state" eventTime="2012-12-
07T09:34:47.644-05:00" nounType="Meter" objectID="3001M"/>
         </ns2:InitiateDisconnectedStatusResult>
      </ns2:InitiateDisconnectedStatusResponse>
   </soapenv:Body>
</soapenv:Envelope>
```

2.1.10.3 Request parameters

Same as `InitiateUsageMonitoring`.

2.1.10.4 Response parameters

Same as `InitiateUsageMonitoring`.

2.1.10.5 Error messages

Same error conditions as for `InitiateUsageMonitoring`, with the following difference.

The most common reasons for the Invalid logical state error are:

a) `InitiateDisconnectedstatus` was previously invoked for this meter but never cancelled so the meter is already in the Disconnected logical state.

b) `InitiateUsageMonitoring` was previously invoked for this meter but never cancelled so the meter is in the Inactive logical state.

2.1.10.6 Configuration properties

Same as `InitiateUsageMonitoring`.

2.1.11 CancelDisconnectedStatus

This method notifies the RNI of meters that should be removed from disconnected status. This is the complement of `InitiateDisconnectedStatus` and causes the logical state of the meter to be changed from *Disconnected* to *Active*.

2.1.11.1 Sample request

```
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:ver="http://www.multispeak.org/Version_3.0">
  <soapenv:Header>
    <ver:MultiSpeakMsgHeader
        Version="3.0" UserID="testuser" Pwd="testpass"
        AppName="myapp" AppVersion="1.5" Company="myco"
        SessionID="123"/>
  </soapenv:Header>
  <soapenv:Body>
    <ver:CancelDisconnectedStatus>
      <ver:meterNos>
        <ver:string>3000M</ver:string>
        <ver:string>3001M</ver:string>
        <ver:string>3002M</ver:string>
        <ver:string>3003M</ver:string>
      </ver:meterNos>
    </ver:CancelDisconnectedStatus>
  </soapenv:Body>
</soapenv:Envelope>
```

2.1.11.2 Sample response

This shows a sample response with a failure for one out of the four meters in the request.

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:ns2="http://www.multispeak.org/Version_3.0">
   <soapenv:Header>
       <ns2:MultiSpeakMsgHeader AppName="RNIGateway" AppVersion="1.0"
                      Company="Sensus" CSUnits="feet"  SessionID="123"/>
   </soapenv:Header>
   <soapenv:Body>
             <ns2:CancelDisconnectedStatusResponse>
         <ns2:CancelDisconnectedStatusResult>
            <ns2:errorObject errorString="Invalid lifecycle state" eventTime="2012-12-
07T09:37:36.140-05:00" nounType="Meter" objectID="3000M"/>
         </ns2:CancelDisconnectedStatusResult>
      </ns2:CancelDisconnectedStatusResponse>
   </soapenv:Body>
</soapenv:Envelope>
```

2.1.11.3 Request parameters

Same as `InitiateUsageMonitoring`.

2.1.11.4 Response parameters

Same as `InitiateUsageMonitoring`.

2.1.11.5 Error messages

Same error conditions as for `InitiateUsageMonitoring`, with the following difference:

The most common reason for the Invalid logical state error is that `InitiateDisconnectedStatus` was not previously invoked for this meter, so this meter is not in a Disconnected logical state.

2.1.11.6 Configuration properties

Same as `InitiateUsageMonitoring`.

2.2 MR notification client

2.2.1 Overview

The RNI includes a web service client that publishes meter readings as notifications to the integrating system. These are typically consumed by the MDM or CB systems.

This notification client uses the CBMR WSDL.

2.2.1.1 References

MultiSpeak v3.0.j reference for the CBMR Server abstract function:
https://apps.cooperative.com/content/public/multispeak/30j/2A_CB_MR.asmx

2.2.1.2 Batching (flow control)

To manage the flow of notifications emitted, a batching mechanism is implemented by the MR notification client. Batching behavior may be controlled by batch size and batch time. The batch size determines how many events must be pending for a particular destination before they will be published in one SOAP payload. The batch time determines the maximum time the client will buffer an event before publishing it.

A batch (containing events) that reaches either limit triggers sending of the batch of events in a SOAP notification.

To manage the flow of notifications emitted, a batching mechanism is provided by the MR system. Batching behavior may be used to control batch size and batch time. Refer to Section 2.2.2 for details regarding configuration for the specific notification (ReadingChangedNotification).

2.2.1.3 Reliable delivery

If the remote system is not available to receive notification messages, then these are saved in a disk persistent cache for later delivery. The number of retries is configurable. This provides resilience to short network outages, maintenance events etc. Note, delivery is *not* guaranteed, and resource limitations (disk space/memory) may force notifications to be discarded.

2.2.2 ReadingChangedNotification

This notification is used to report cumulative register reads and tier/TOU reads or onTheGlass readings depending on profile configuration for `InitiateMeterReadByMeterNumber`.

At present this is **only** a solicited notification (i.e., a `ReadingChangedNotification` is only published by the RNI in response to a prior `InitiateMeterReadByMeterNumber` request).

2.2.2.1 *Sample notification*

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                      xmlns:ns2="http://www.multispeak.org/Version_3.0">
   <soapenv:Header>
       <ns2:MultiSpeakMsgHeader UserID="cbmr user" Pwd="cbmr pw"
               AppName="RNI Gateway (MR)" AppVersion="3.1" Company="Sensus"/>
   </soapenv:Header>
   <soapenv:Body>
      <ns2:ReadingChangedNotification >
         <ns2:changedMeterReads>
            <ns2:meterRead>
               <ns2:meterNo>1001M</ns2:meterNo>
               <ns2:deviceID>1001</ns2:deviceID>
               <ns2:readingDate>2012-08-13T14:13:30.000Z</ns2:readingDate>
               <ns2:posKWh>60532</ns2:posKWh>
               <ns2:readingValues>
                  <ns2:readingValue>
                     <ns2:units>kWh</ns2:units>
                     <ns2:value>60532</ns2:value>
                     <ns2:name>Consumption</ns2:name>
                  </ns2:readingValue>
               </ns2:readingValues>
            </ns2:meterRead>
         </ns2:changedMeterReads>
      </ns2:ReadingChangedNotification>
   </soapenv:Body>
</soapenv:Envelope>
```

2.2.2.2 *Notification parameters*

	Attribute or Element name	Usage
HEADER	Common response header attributes	
BODY	meterNo*	Meter number (i.e., Meter ID) of the meter whose reading is being returned.
	deviceId	The FlexNet ID (i.e., Radio endpoint ID).
	readingDate	The sample point for this reading.
	kW	The peak demand since the last demand reset. This is only populated for a non-TOU configured meter.
	posKWh	The cumulative energy consumption. This may be F, F+R or F-R depending on the meter setup. This is only populated for a non-TOU configured meter.
	negKWh	The reverse energy consumption if the meter is configured to report it separately. This is only populated for a non-TOU configured meter. Note, this does not indicate the value is in units Kwh; Units information is conveyed in the readingValue.
	readingValues	Collection of reading values. This is only populated for a non-TOU configured meter.
	readingValue.name*	Type of reading will be one of Consumption, Reverse or Demand, netKWh, totalKWh.

	Attribute or Element name	Usage
	readingValue.units*	Units of measure (UOM) for this reading e.g., kW, kWh, kVA, kVAr, Wh, etc.
	readingValue.value*	Reading value: For Consumption, this is the cumulative usage. If the corresponding units are kWh, this value is the same as the posKWh element. For Reverse, this is the reverse energy usage. If the corresponding units are kWh, this value is the same as the negKWh element. For Demand, this is the Peak Demand since the last demand reset. If the corresponding units are kW, then this value is the same as the kW element.
	TOUReadings	Collection of TOU readings, one per tier. This is only populated for a TOU configured meter.
	TouReading.ratePeriod*	Numeric value of the TOU tier being reported. The value 0 indicates the TOTALs across all the other tiers.
	TouReading.kWh	The Consumption in kWh. This will only be available if the meter reports consumption in kWh.
	TouReading.kW	The Peak Demand in kW since the last demand reset. This is only reported if the meter reports peak demand in kW.
	TOUReading.kWDateTime	The date-time of when the peak demand occurred. This is present only if TOUReading.kW is present.
	TOUReading.extensionsList	Sensus-specific extensions to allow TOU readings to be reported in units other than kW and kWh.
	TOUReading.extensionsItem	One of Consumption, Demand, and Demand Time: If the consumption is in kWh, this has the same value as TouReading.kWh. If the peak demand is in kW, this has the same value as TouReading.kW. If the demand is in kW, then DemandTime has the same value as TouReading.kWDateTime.
	TOUReading.extName	The Unit of Measurement (UOM) for a consumption (e.g., kWh, kVah) or demand (e.g., kW, kVAR) or the constant DemandTime.
	TOUReading.extValue	Value corresponding to the extName.
	TOUReading.extType	The data type of the extValue. This is double if the extName is a UOM and dateTime for DemandTime.

2.2.2.3 *Expected response*

Responses are *mandatory* when a `ReadingChangedNotification` request is published. The expected response is a `ReadingChangedNotificationResponse` or a SOAP Fault. Either of these responses will be interpreted as a successful delivery (regardless of errors returned in the response). If neither of these responses is provided, redelivery may be attempted depending on reliable delivery configuration.

2.2.2.4 Configuration properties

	Property name	Usage
DB	Multispeak.CBMR.DestinationEnabled (Customer configuration)	An on/off switch for the MR notification client in the RNI.
	Multispeak.CBMR.DestinationURL (Customer configuration)	The URL of the CBMR web service (e.g., https://www.myhost.com/multispeak/cbmr).
	Multispeak.CBMR.DestinationUserId (Customer configuration)	A user ID for the CBMR service. This is the value used for the UserID header attribute of the request.
	Multispeak.CBMR.DestinationPassword (Customer configuration)	The password corresponding to the user account on the CBMR system (see above). This is the value used for the Pwd header attribute of the request.
	Multispeak.CBMR.ReadingChangedNotification.BatchSize (Customer configuration)	The number of readings the MR system will batch per SOAP message (e.g., a value of 100 means that MR will wait until it receives 100 readings for that destination and then forward them on in one ReadingChangedNotification message). If batch size and batch time are specified, then either condition may trigger the SOAP message to be sent. Changes to this parameter take effect after gateway restart. Default is 5.
	Multispeak.CBMR.ReadingChangedNotification.BatchTime (Customer configuration)	The time in milliseconds that the MR system holds readings to batch into a SOAP message. For example, a value of 5000 means that MR will buffer events for 5 seconds before sending them on in one ReadingChangedNotification message. Changes to this parameter take effect after gateway restart. NOTE: Even if the batching is primarily being controlled with the batchSize property, a non-zero value should always be specified for the batch time property so that readings do not get buffered in the MR system indefinitely. Default is 100.
	Multispeak.CBMR.ReadingChangedNotification.MaxCacheCount (Customer configuration)	Maximum number of caches that the notification client should keep in play. A higher number of caches provides a better tolerance to slow consumers and event floods because the notification client has a bigger sandbox for processing events. This value should be bounded depending on the available memory on the server. Recommend that the value in a production system be 2 or higher. Changes to this parameter take effect after gateway restart. Default is 100.
	Multispeak.CBMR.ReadingChangedNotification.RetryCount	Number of retry attempts to send a SOAP message to the subscriber.

	Property name	Usage
	(Customer configuration)	Default is 5.
	Multispeak.CBMR.ReadingChangedNotification.MaxAutoRecoveries	If all configured retry attempts have completed without success, the gateway will enter an auto recovery state in which additional rounds of retries will be scheduled. This value controls the maximum number of auto recovery cycles before undeliverable data is purged. (-1 is unlimited i.e., data is never purged.) Default is 32.
	Multispeak.CBMR.ReadingChangedNotification.AutoRecoveryInterval	Time in milliseconds between auto recovery attempts. Default is 900000 (i.e., 15 minutes).

2.2.2.5 *Error messages*

	Error	Reasons
BUSINESS ERRORS	No response from meter	The device did not respond to the reading request. One possible reason may be that the device is not powered on. This may be a transient condition so the client should wait and retry.
	No routes to endpoint	The on-air message could not be sent to the meter because the RNI does not have any routes to it.
	Request was preempted	The InitiateMeterReadByMeterNumber request was not sent because it was preempted by a higher priority request.
	Message throttled by communication module	The RNI's traffic management policy has temporarily blocked transmissions to the meter. The most common reason is the RNI already has a message in the queue or in transit for the meter. This is a transient condition so the client should wait and retry.
	Could not retrieve meter read information from the meter response. The meter may be experiencing a power fail.	When the InitiateMeterReadByMeterNumber is sent in on-the-glass mode, a meter experiencing a power fail may result in this message being produced.

2.3 CBMR

2.3.1 Overview

The RNI MultiSpeak v3 gateway includes a web service client for synchronizing a meter with a remote CB system by invoking methods on the CBMR service.

The MultiSpeak gateway deployment includes both a command line script and a web interface to run each of the methods in this section.

The web interface URL is [protocol]://[rni_web_server]/multispeakv3-cb-client.

2.3.1.1 References

MultiSpeak v3.0.j reference for the CBMR Server abstract function:
https://apps.cooperative.com/content/public/multispeak/30j/2A_CB_MR.asmx

2.3.2 GetAllMeters

This operation gets meter data for all meters in the CB and *adds* meters in the RNI or *updates* the corresponding entries in the RNI, if present. This is used to synchronize the relevant meter data in the RNI with the CB (system of record).

2.3.2.1 Sample request

```
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:ns2="http://www.multispeak.org/Version_3.0">
  <soapenv:Header>
    <ns2:MultiSpeakMsgHeader
        Pwd="cbmrpass" UserID="cbmruser"
        AppName="RNIGateway" AppVersion="1.0" Company="Sensus"
        CSUnits="feet"/>
  </soapenv:Header>
  <soapenv:Body>
    <ns2:GetAllMeters>
      <ns2:lastReceived>M3000</ns2:lastReceived>
    </ns2:GetAllMeters>
  </soapenv:Body>
</soapenv:Envelope>
```

2.3.2.2 Sample response

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
                      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                      xmlns:ns1="http://www.multispeak.org/Version_3.0">
   <soapenv:Header>
      <ns1:MultiSpeakMsgHeader
             CSUnits="feet" AppName="iVUE" Company="NISC"
               LastSent="11425854" ObjectsRemaining="-1"
               SessionID="1258123773638" Version="3.0"/>
   </soapenv:Header>
   <soapenv:Body>
      <ns1:GetAllMetersResponse>
         <ns1:GetAllMetersResult>
            <ns1:meter objectID="10027052">
               <ns1:facilityID>83100212</ns1:facilityID>
               <ns1:meterNo>10027052</ns1:meterNo>
               <ns1:serialNumber/>
               <ns1:meterType>0</ns1:meterType>
               <ns1:manufacturer>SI</ns1:manufacturer>
               <ns1:AMRType>SENS</ns1:AMRType>
               <ns1:nameplate>
                  <ns1:dials>5</ns1:dials>
                  <ns1:form>2S</ns1:form>
                  <ns1:multiplier>1</ns1:multiplier>
               </ns1:nameplate>
               <ns1:utilityInfo>
                  <ns1:district>4</ns1:district>
                  <ns1:servLoc>116122</ns1:servLoc>
                  <ns1:accountNumber>2297583000</ns1:accountNumber>
                  <ns1:custID>2297583</ns1:custID>
               </ns1:utilityInfo>
            </ns1:meter>
         </ns1:GetAllMetersResult>
      </ns1:GetAllMetersResponse>
   </soapenv:Body>
</soapenv:Envelope>
```

2.3.2.3 Request parameters

	Attribute/Element	Usage
HEADER	UserID*	User ID on the CB assigned to the MultiSpeak gateway.
	Pwd*	Clear text password corresponding to the UserID.
	AppName*	MultiSpeak gateway client name e.g., RNIGateway.
	AppVersion*	MultiSpeak gateway client version e.g., 1.0.
	Company*	MultiSpeak gateway client provider e.g., Sensus.
BODY	lastReceived	The last object ID received in a previous batch. The returned meters will commence with the first meter ID after the lastReceived ID. Leave this attribute empty to get the meters starting from the beginning.

2.3.2.4 Response parameters

	Attribute/Element	Usage
HEADER	AppName	Web service client application name. This is only used for logging.
	AppVersion	Web service client application version. This is only used for logging.
	Company	Web service client application vendor. This is only used for logging.
	CustomerID	The identifier for the utility in the RNI. Usually 4 letter alphanumeric code.
	ObjectsRemaining	The number of meters remaining in the CB at that point in time. A value of zero indicates that there are no more meters remaining. A negative value is interpreted as an unknown number of meters remaining. Note that if the attribute is not provided it is assumed to be zero.
BODY	meter/meterNo*	The identifier for the meter assigned by the utility i.e., the MeterID.
	meter/AMRType*	The Vendor ID for the AMR system that supports this meter e.g., SENS. Meters that do not belong to the Sensus system will be ignored.
	meter/nameplate/dials	The dials for the meter.
	meter/nameplate/form	The form factor for the meter.
	meter/nameplate/transponderID	This is the FlexnetID for that meter. This is required only if the meter does not already exist in the RNI. If present this should be the same as the FlexnetID for that meter.
	meter/utilityInfo/servLoc	Service location identifier associated with the meter.
	meter/deviceClass	The billing rate code for the meter i.e., the RateCode.

2.3.2.5 Error messages

	Error	Reasons
SYSTEM ERRORS	WebServiceIOException MalformedURLException	The value of Multispeak.CBMR.DestinationURL is not a valid URL.
	WebServiceIOException Connection Refused	The destination web service is not accepting connections. Ensure that the Multispeak.CBMR.DestinationURL in the DB is correct and that the remote service is running.

	Error	Reasons
	WebServiceTransportException Internal Server Error [500]	The remote CBMR service has responded with a non-SOAP 1.1 message. Some common causes are: The service does not support SOAP 1.1. The service expects a soapAction in the request. The service is responding with an HTTP error code.
BUSINESS ERRORS	Meter does not exist	The CB returned meter ID that does not belong to any meter in the RNI. This is an expected occurrence if the CB contains more meters than the RNI.

2.3.2.6 *Configuration properties*

	Property name	Usage
DB	Multispeak.CBMR.DestinationEnabled (Customer configuration)	An on/off switch for the CBMR client in the RNI. This should be set to True.
	Multispeak.CBMR.DestinationURL (Customer configuration)	The URL of the CBMR web service e.g., https://www.myhost.com/multispeak/cbmr
	Multispeak.CBMR.DestinationUserId (Customer configuration)	The user ID on the CB system used by the MultiSpeak gateway. This is the value used for the UserID header attribute of the request.
	Multispeak.CBMR.DestinationPassword (Customer configuration)	The password corresponding to the user account on the CB system (see above). This is the value used for the Pwd header attribute of the request.
FILE	ws.filter.amrVendorId	Expected value of the AMRVendorID for all meters managed by this system.
	rniMeterBatchSize	Large GetAllMetersResponse will be sub-divided into smaller batches during update into the RNI based on this value. Default is 100. (Modification NOT recommended.)

2.3.3 GetAllServiceLocations

This operation gets the service location data for all meters in the CB and updates the corresponding entries in the RNI, if present. This synchronizes the relevant service location data in the RNI with the CB (system of record).

2.3.3.1 *Sample request*

```
<soapenv:Envelope
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:ns2="http://www.multispeak.org/Version_3.0">
  <soapenv:Header>
    <ns2:MultiSpeakMsgHeader
        Pwd="cbmrpass" UserID="cbmruser"
        AppName="RNIGateway" AppVersion="1.0" Company="Sensus"
        CSUnits="feet"/>
  </soapenv:Header>
  <soapenv:Body>
    <ns2:GetAllServiceLocations>
```

```
      <ns2:lastReceived>223232</ns2:lastReceived>
    </ns2:GetAllServiceLocations>
  </soapenv:Body>
</soapenv:Envelope>
```

2.3.3.2 Sample response

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                      xmlns="http://www.multispeak.org/Version_3.0">
   <soapenv:Header>
      <MultiSpeakMsgHeader
             CSUnits="feet" AppName="iVUE" Company="NISC"
               LastSent="11425854" ObjectsRemaining="-1"
               SessionID="1258123773638" Version="3.0"/>
   </soapenv:Header>
   <soapenv:Body>
      <GetAllServiceLocationsResponse>
         <GetAllServiceLocationsResult>
            <serviceLocation objectID="1">
               <comments>HOUSE</comments>
               <mapLocation>
                  <coord>
                     <X>2337672.92900</X>
                     <Y>1633896.31200</Y>
                  </coord>
               </mapLocation>
               <gridLocation>01030001</gridLocation>
               <custID>2000708</custID>
               <accountNumber>2000708000</accountNumber>
               <servAddr1>2641 BEN HIGGINS RD</servAddr1>
               <servAddr2/>
               <servCity>DAHLONEGA</servCity>
               <servState>GA</servState>
               <servZip>305335224</servZip>
               <servType/>
               <revenueClass>40</revenueClass>
               <servStatus>1</servStatus>
               <billingCycle>12</billingCycle>
               <route>10006</route>
               <specialNeeds/>
               <loadMgmt/>
               <connectDate>2004-06-24T04:00:00.262Z</connectDate>
               <network>
                  <boardDist>4</boardDist>
                  <schoolDist>6</schoolDist>
                  <district>2</district>
                  <county>6</county>
                  <substationCode>EE</substationCode>
                  <feeder>2</feeder>
                  <eaLoc name="Unknown"/>
                  <linkedTransformer>
                     <bankID>1</bankID>
                     <unitList>
                        <unitID>1</unitID>
                     </unitList>
                  </linkedTransformer>
                  <linemanServiceArea>N</linemanServiceArea>
               </network>
            </serviceLocation>
               </GetAllServiceLocationsResult>
               </GetAllServiceLocationsResponse>
          </soapenv:Body>
</soapenv:Envelope>
```

2.3.3.3 Request parameters

	Attribute/Element	Usage
HEADER	UserID*	User ID on the CB assigned to the MultiSpeak gateway.
	Pwd*	Clear text password corresponding to the UserID.
	AppName*	MultiSpeak gateway client name e.g., RNIGateway.
	AppVersion*	MultiSpeak gateway client version e.g., 1.0.
	Company*	MultiSpeak gateway client provider e.g., Sensus.
BODY	lastReceived	The last object ID received in a previous batch. The returned meters commence with the first meter ID after the lastReceived ID.

2.3.3.4 Response parameters

	Attribute/Element	Usage
HEADER	AppName	Web service client application name. This is only used for logging.
	AppVersion	Web service client application version. This is only used for logging.
	Company	Web service client application vendor. This is only used for logging.
	CustomerID	The identifier for the utility in the RNI. Usually 4 letter alphanumeric code.
	ObjectsRemaining	The number of meters remaining in the CB at that point in time. A value of zero indicates that there are no more meters remaining. A negative value is interpreted as an unknown number of meters remaining. Note that if the attribute is not provided it is assumed to be zero.
BODY	serviceLocation/objectID*	The service location ID associated with a meter.
	serviceLocation/servAddr1	Address line 1.
	serviceLocation/servAddr2	Address line 2.
	serviceLocation/servCity	City.
	serviceLocation/servState	State.
	serviceLocation/servZip	Zip.
	serviceLocation/facilityName	Facility name.
	serviceLocation/billingCycle	The billing cycle for meter(s) at this service location.
	serviceLocation/revenueClass	The rate code for the meter(s) at this service location.

2.3.3.5 Error Messages

	Error	Reasons
SYSTEM ERRORS	WebServiceIOException MalformedURLException	The value of Multispeak.CBMR.DestinationURL is not a valid URL.
	WebServiceIOException Connection Refused	The destination web service is not accepting connections. Ensure that the Multispeak.CBMR.DestinationURL in the DB is correct and that the remote service is running.
	WebServiceTransportException Internal Server Error [500]	The remote CBMR service has responded with a non-SOAP 1.1 message. Some common causes are: The service does not support SOAP 1.1 The service expects a soapAction in the request. The service is responding with an HTTP error code.
BUSINESS ERRORS	Service Location does not exist	The CB returned a service location ID that does not belong to any meter in the RNI. This is an expected occurrence if the CB contains more meters than the RNI.

2.3.3.6 Configuration properties

	Property name	Usage
DB	Multispeak.CBMR.DestinationEnabled (Customer configuration)	An on/off switch for the CBMR client in the RNI. This should be set to True.
	Multispeak.CBMR.DestinationURL (Customer configuration)	The URL of the CBMR web service (e.g., https://www.myhost.com/multispeak/cbmr).
	Multispeak.CBMR.DestinationUserId (Customer configuration)	The user ID on the CB system used by the MultiSpeak gateway. This is the value used for the UserID header attribute of the request.
	Multispeak.CBMR.DestinationPassword (Customer configuration)	The password corresponding to the user account on the CB system (see above). This is the value used for the Pwd header attribute of the request.
FILE	rniServiceLocationBatchSize	Large GetAllServiceLocationsResponse will be sub-divided into smaller batches during update into the RNI based on this value. Default is 50. (Modification NOT recommended.)
	meterlifecycle.serviceLocation.strictUpdateDisable	This configuration controls whether missing service locations will be added to the RNI. If this value is true, then missing service locations will be added to the RNI. If this value is false, missing service locations will not be added. Default is False.

	Property name	Usage
	meterlifecycle.ignoreMissingServiceLocation	The GetAllServiceLocationsResponse for some customers contains service locations that are not intended to be added to the RNI. If the meterlifecycle.serviceLocation.strictUpdateDisable configuration setting is False, then no new service locations are added and an error is returned. This property allows this condition to be treated as a no-op instead of an error, so that overall process can succeed. Default is True.

2.3.4 CBMR script

The MultiSpeak installation includes a command line utility to trigger data synchronization using the `GetAllMeters` and `GetAllServiceLocations` CBMR web methods. This utility uses the `Multispeak.CBMR.*` configuration values in the RNI to identify the remote web service endpoint.

The script is `/opt/flexnet-gateway/webapps/multispeakv3-cb-client/WEB-INF/bin/cb.sh`.

Log messages are written to `cb.log`.

2.3.4.1 Usage

Change directory to is `/opt/flexnet-gateway/webapps/multispeakv3-cb-client/WEB-INF/bin/` prior to running the script.

The usage is as follows. All flags are optional because the script will prompt for missing information as needed.

```
      cb.sh [-q|-n|-v]
                   [-c <Customer ID>]
                   [-lm <last received object ID>]
                   [-ls <last received service location object ID>]
                   [<operation>]
```

2.3.4.2 Options

Option	Description
-h	Displays the help message.
-q|-n|-v	Supported verbosity levels for the script output are quiet, normal, and verbose. If none are specified, then this defaults to normal verbosity.
-c <Customer ID>	The customerID identifies the company whose CB service should be invoked. This must be provided for all operations.
-lm <Last received object ID>	Object ID that will be used as the lastReceived element in CB requests.
-ls <Last received service location object ID>	Service location object ID that is used as the lastReceived element in CB requests.

Option		Description
Operations	PingURL	Pings the remote CB server.
	GetMethods	Retrieves the supported methods from the remote CB service. Use this to verify that the remote service supports the GetAllMeters and GetAllServiceLocations methods.
	StartSyncAll	Starts the data synchronization. This includes meter and location synchronization.
	StartSyncMeters	Starts meter synchronization. This option is provided in case the GetAllMeters needs to be run separately. If both meter and location data need to be synchronized, use StartSyncAll.
	StartSyncServiceLocations	Starts service location synchronization. This option is provided in case the GetAllServiceLocations needs to be run separately. Service locations can only be updated with this operation, i.e., no new service locations are created. If both meter and location data need to be synchronized, use StartSyncAll.
	StopSyncAll	Stops the data synchronization. This is the complement of StartSyncAll.
	StopSyncMeters	Stop the meter synchronization. This is the complement of StartSyncMeters.
	StopSyncServiceLocations	Stop the service location synchronization. This is the complement of StartSyncServiceLocations.
	GetSyncMeterStatus	Display the status of the meter synchronization.
	GetSyncServiceLocationStatus	Display the status of the service location synchronization.

2.3.4.3 Examples

To verify connectivity with the remote CB service:

```
cb.sh –v –c ACME PingURL
```

To start a data synchronization:

```
cb.sh –v –c ACME StartSyncAll
```

To view the status of a data synchronization:

```
cb.sh –v –c ACME GetSyncMeterStatus

cb.sh –v –c ACME GetSyncServiceLocationStatus
```

To stop a data synchronization:

```
cb.sh –v –c ACME StopSyncAll
```

To continue a data synchronization:

```
cb.sh –v –c ACME –lm <meterID> –ls <serviceLocationID> StartSyncAll
```

Xylem |ˈzīləm|

1) The tissue in plants that brings water upward from the roots;

2) a leading global water technology company.

We’re a global team unified in a common purpose: creating advanced technology solutions to the world’s water challenges. Developing new technologies that will improve the way water is used, conserved, and re-used in the future is central to our work. Our products and services move, treat, analyze, monitor and return water to the environment, in public utility, industrial, residential and commercial building services settings. Xylem also provides a leading portfolio of smart metering, network technologies and advanced analytics solutions for water, electric and gas utilities. In more than 150 countries, we have strong, long-standing relationships with customers who know us for our powerful combination of leading product brands and applications expertise with a strong focus on developing comprehensive, sustainable solutions.

Sensus
637 Davis Drive
Morrisville, NC 27560
Tel +1.800.638.3748
www.sensus.com