MultiSpeak Version 3.0 Interoperability Assertion

Statement of Interoperable Functionality Between:

Vendor(s)	Product	Product Version	Role	Web Client Interfaces	Web Server Interfaces
National Information Solutions Cooperation (NISC)	iVUE CIS	x.18 (and greater)	Customer Billing (CB)	MRCB	CBMR
Sensus	RNI	2.2.4 (and greater)	Meter Reading (MR)	CBMR	MRCB

1 Summary

Web Service interfaces using the MultiSpeak® 3.0 standard were developed in order to provide NISC and Sensus customers a way to keep the meter information synchronized in the NISC iVUE CIS and the Sensus RNI (AMI head end) without having to update and maintain data in both systems through manual processes. The integration enables the exchange of data for meter lifecycle management (install/remove), service location updates and meter reading.

2 Prerequisites:

- HTTP/HTTPS communications must be available between the iVUE CIS and the RNI MultiSpeak v3.0 gateway.
- If using HTTPS then the appropriate SSL certificates must be configured on both systems.
- The RNI MultiSpeak v3.0 gateway does not allow unsecured access, hence a user account with a password *must* be created in the RNI with the appropriate access levels to avail of the AMI capabilities needed for the integration. This user name and password must be saved in iVUE and passed in on *every* MultiSpeak request.
- The destination URL of the CBMR endpoint in iVUE must be configured in the Sensus RNI.
- If the CBMR service is secured then the user name and password must also be configured in the RNI. This will be sent with every MultiSpeak request.
- The meter number (a.k.a. meter ID) for a meter *must be the same* in iVUE and the RNI. This is the shared identifier for a meter in both systems. This must be unique within the operating company.

- The meters that are to be managed through MultiSpeak *must be inventoried* in the RNI. This is done by uploading a proprietary meter manufacturing file into the RNI.

3 Specific Vendor Assertions:

3.1 iVUE will notify the Sensus RNI of newly deployed (added) AMR meters

Importance to user: This function will save the utility time and money during the initial deployment and ongoing maintenance of their Sensus AMI system by removing the need to manually enter or import a list of newly deployed AMR meters in the Sensus RNI.

How Achieved:
An in-stock meter is installed at a meter service location on iVUE CIS

A transaction history record is created for the meter installation (Transaction = 'MTRINSREM', Activity = 'ADD', Meter Status = 1 or 2).

A *MeterAddNotification* with the meter number is automatically sent by iVUE to the RNI. The RNI will move the indicated meter from an "Inventory" to an "Install" state. If the meter number is not known to the RNI then an errorObject is returned to iVUE.

3.2 iVUE will notify the Sensus RNI of removed AMR meters

Importance to user: This function will save the utility time and money during the ongoing maintenance of their systems by removing the need to manually enter a list of removed AMR meters in RNI

How Achieved:
A remove from service transaction is completed in iVUE.

A transaction history record is created for the meter removal (Transaction = 'MTRINSREM', Meter Service Function = 'Remove Meter'), which is picked up and the meter number and status is passed to RNI™.

A *MeterRemoveNotification* with the meter number is automatically sent by iVUE to the RNI. The RNI will move the indicated meter from an "Install" to an "Inventory" state. If the meter number is not known to the RNI or not in an "Install" state then an errorObject is returned to iVUE.

3.3 iVUE will notify the Sensus RNI of exchanged AMR meters

Importance to user: This function will save the utility time and money during the ongoing maintenance of their Sensus AMI system by removing the need to manually enter meter change information in RNI

How Achieved: A meter exchange is performed in the iVUE CIS.

A transaction history record is created for the meter exchange (Transaction = 'MTREXC', Activity = 'ADD'), and the out and in meters and statuses are picked up and passed to RNI. This will generate a *MeterRemoveNotification* for the 'out' meter followed by a *MeterAddNotification* for the 'in' meter.

The RNI handles each of the *MeterRemoveNotification* and *MeterAddNotification* methods independently i.e. it does not associate the two transactions as an "exchange". Hence the RNI side processing is the same as in the two use cases above.

3.4 iVUE will notify the Sensus RNI of changed service location information

Importance to user: This function will save the utility time and money by removing the need to manually enter initial and changed customer information to the Sensus AMI system.

How Achieved:
If a property of a service location such as the customer id, account number, map location, revenue class or billing cycle changes within iVUE then iVUE notifies RNI of a change in any of several service location specific elements by sending a *ServiceLocationChangedNotification* with the changed service location information.

The RNI will use the objectID value to identify the service location record to be updated. If the record exists in the RNI DB then it will be updated with the properties that it supports such as street address and billing cycle.If the indicated service location is not known to the RNI or any of the changed values do not pass the validation rules in the RNI then an *errorObject* will be returned to iVUE.

3.5 iVUE will notify the Sensus RNI of locations where Usage should be monitored.

Importance to user: Provides automatic maintenance of status groups that are used to track locations where electric usage is not expected, such as vacant dwellings. This function is intended to save the utility money by allowing for a "soft disconnect" of a

meter versus sending an employee to “hard disconnect” the meter. The utility can take comfort in knowing they will be notified within a day of usage occurring at a virtually disconnected location.

How Achieved: A disconnect order is processed within the iVUE CIS.

The meter status on meter inventory is made “inactive installed” (status 2) and neither the “Disconnect at Transformer” nor the “Dead Plug” options are checked.

A transaction history record is created for the disconnect order (Transaction = ‘SRVORD’, SO function = ‘DS’, SO Status = ‘C’), which is picked up and the meter number and status is passed to RNI

An *InitiateUsageMonitoring* request is automatically sent by iVUE to the RNI. The RNI will set the logical state of the indicated meter to “Inactive”. A report can be generated from the RNI DB of the meters in usage monitoring along with their last known usage.

3.6 iVUE will notify the Sensus RNI of locations where Usage Monitoring is no longer required.

Importance to user: This feature is intended to benefit the utility by removing the necessity to manually remove meters from the Usage Monitoring status group.

How Achieved: A reconnect order is processed within the iVUE CIS.

The meter status on meter inventory is made “active” (status 1).

A transaction history record is created for the reconnect order (Transaction = ‘SRVORD’, SO function = ‘RS’, ‘RN’, ‘RO’ or ‘NS’, SO Status = ‘C’), which is picked up and the meter number and status is passed to RNI.

A *CancelUsageMonitoring* request is automatically sent by iVUE to the RNI. The RNI will set the logical state of the indicated meter to “Active”.

3.7 iVUE will notify the Sensus RNI of locations where no AMR reading is expected.

Importance to user:
This feature is intended to benefit the utility by removing the necessity to manually add meters to the Disconnected Status group.

How Achieved:

When a meter is totally disconnected and no electricity is available to power the AMR device, a disconnect order is processed within the iVUE CIS.

The meter status on meter inventory is made “inactive installed” (status 2) and the “Disconnect at Transformer” or “Dead Plug” are checked indicating power to the meter is disconnected:

In this scenario a transaction history record is created for the disconnect order (Transaction = ‘SRVORD’, SO function = ‘DS’, SO Status = ‘C’), which is picked up and the meter number and status is passed to RNI.

An *InitiateDisconnectedStatus* request is automatically sent by iVUE to the RNI. The RNI will set the logical state of the indicated meter to “Disconnected”. This will remove the meter from the scheduled reporting process.

3.8 iVUE will notify the Sensus RNI of locations where AMR readings are once again expected.

Importance to user:
This feature will benefit the utility by removing the necessity to manually remove meters from the Disconnected Status group.

How Achieved:
A reconnect order is processed within the iVUE CIS.

The meter status on meter inventory is made “active” (status 1).

A transaction history record is created for the reconnect order (Transaction = ‘SRVORD’, SO function = ‘RS’, ‘RN’, ‘RO’ or ‘NS’, SO Status = ‘C’), which is picked up and the meter number and status is passed to RNI

A *CancelDisconnectedStatus* request is automatically sent by iVUE to the RNI. The RNI will set the logical state of the indicated meter to “Active”. This will re-include the meter in the scheduled reporting process.

3.9 iVUE can initiate a new reading and request latest meter readings by meter number from the Sensus RNI.

Importance to user:
The utility benefits because the Customer Service Representative doesn’t have to log in to a separate system to get the most recent meter reading information.

How Achieved:
This feature is intended to be used by Customer Service to assist in answering customer questions or in completing connect/disconnect orders.

Method 1: The CSR selects the "Read Now" button on the iVUE Reading Interval History panel which initiates a new reading for the specified meter number and when the reading is returned it is posted to the Interval/Daily Reading table. iVUE will request RNI to perform a physical read of the meter and return the current usage to the iVUE system. If a new reading is not received within the timeout period set in iVUE, then iVUE will request the last available reading from RNI and post that reading as a substitute.

Method 2: The CSR selects the reading "binoculars" button on the iVUE Service Order Connect/Disconnect Meter panel which initiates a new reading for the specified meter number and when the reading is returned it is posted to the Service Order reading field. Again, RNI will perform a physical read of the meter and return the current usage to the iVUE system. As with method 1, the latest available reading can be retrieved if no new reading is received.

In both the methods above the RNI receives an *InitiateMeterReadByMeterNumber* request and in turns sends an on-air message to the meter. The reading from the meter is stored in the DB but, at this time, the RNI does not support the *ReadingChangedNotification* . Thus iVUE falls back to getting the last available reading through a *GetLatestReadingByMeterNo* request. This looks up the latest reading from the DB and returns it in the response.

3.10 iVUE can request all AMR supported meters from the Sensus RNI

Importance to user: The user can determine which consumers in the iVUE have meters with Sensus AMR capabilities.

How Achieved: iVUE sends the *GetAMRSupportedMeters* request to the RNI which does a DB lookup for a batch of "Installed" meters. The AMRType and TransponderID for each meter from the response are updated in iVUE.

iVUE will iterate over all the meters in the RNI, one batch at a time by using the *objectsRemaining* and *lastReceived* properties. The size of the batch is a configuration on the RNI.

3.11 RNI can request meter information from the iVUE CIS

Importance to user: This capability allows the Sensus RNI to get the list of installed meters from the CIS so that it can update its internal data store. This will save the utility time and money during the initial deployment and ongoing maintenance of their Sensus SENSUS–NISC Billing Interoperability Document

AMI system by removing the need to manually enter or import the list of installed meters in the RNI. This capability can also be used as a fallback mechanism to catch up with missed meter add and change notifications from the CIS.

How Achieved: Kick off a command line script cbmr.sh on the Sensus RNI to get batches of meters from iVUE.

The RNI sends the *GetAllMeters* request to iVUE which does a DB lookup for a batch of deployed meters. Each meter in the response is processed on the RNI as follows:

- The supported properties (including service location) for each meter from the response are updated in the RNI.
- If a meter is in "Inventory" then it is moved to "Install" as part of the update.
- If a meter is not known to the RNI then it is ignored.

RNI will iterate over all the meters in iVUE CIS, one batch at a time by using the *objectsRemaining* and *lastReceived* properties. The size of the batch is a configuration in iVUE.

Note: The AMRType property is used to identify meters on the Sensus AMI network. Only meters with an AMRType of "SENS" are processed by the RNI.

3.12 RNI can request service location information from the iVUE CIS

Importance to user: This capability allows the Sensus RNI to get the list of service locations from the CIS so that it can update its internal data store. This will save the utility time and money during the initial deployment and ongoing maintenance of their Sensus AMI system by removing the need to manually update the service location properties in the RNI. This capability can also be used as a fallback mechanism to catch up with missed service location change notifications from the CIS.

How Achieved: Kick off a command line script cbmr.sh on the Sensus RNI to get batches of meters from iVUE.

The RNI sends the *GetAllServiceLocations* request to iVUE which does a DB lookup for a batch of deployed meters. Each meter in the response is processed on the RNI as follows:

- The supported properties (street address, billing cycle) for each service location from the response are updated in the RNI.
- If a service location is not known to the RNI then it is ignored.

RNI will iterate over all the meters in iVUE CIS, one batch at a time by using the *objectsRemaining* and *lastReceived* properties. The size of the batch is a configuration in iVUE.

4 Summary of Interoperability Test Results (#2A MR>CB)

PRODUCTS: Sensus RNI and NISC iVUE

Table 3: Recommended MultiSpeak Methods

Method Name	Importance to User	Supported by Server[1] (MR)	Supported by Client[2] (CB)	Verified Inter-operable[3]
CustomerChangedNotification	CB Notifies MR of a change in the Customer object by sending the changed customer object		X	
GetAMRSupportedMeters	Returns all meters that have AMR	X	X	X
GetHistoryLogByMeterNo	Returns History Log Data for a given MeterNo and Date Range			
GetHistoryLogsByDate	Returns History Log Data for a all Meters Given a Date Range			
GetHistoryLogsByDateAndEventCode	Returns History Log Data for a all Meters Given the eventCode and a Date Range			
GetHistoryLogsByMeterNoAndEventCode	Returns History Log Data for a given MeterNo, eventCode and Date Range			
InitiateMeterReadByMeterNumber	CB requests a new meter reading from MR, on meters selected by meter number.	X	X	X
GetReadingsByDate	Returns Reading Data for All Meters Given a Date Range			
GetLatestReadingByMeterNo	Returns Meter Reading Data for a given MeterNo and Date Range	X	X	X
GetReadingsByMeterNo	Returns Meter Reading Data for a given MeterNo and Date Range			
IsAMRMeter	Return true if given meterNo has AMR			
MeterAddNotification	CB Notifies MR to Add the associated Meter(s).	X	X	X
MeterChangedNotification	CB Notifies MR of a change in the Meter object by sending the changed meter object.	X	X	X
MeterRemoveNotification	CB Notifies MR to remove the associated Meter(s).	X	X	X
ServiceLocationChangedNotification	CB Notifies MR of a change in the Service Location	X	X	X
GetMethods	Requests a list of web service methods supported by the Customer Billing program.	X	X	X
PingURL	Queries status of the Customer Billing program.	X	X	X

1) Supported by Server means that the server has demonstrated in some interoperability test (not necessarily with this client) that it can support the method.
2) Supported by Client means that the client has demonstrated in some interoperability test (not necessarily with this server) that it can call the method.
3) Verified Interoperable means that both the client and server have demonstrated in this interoperability test that they can usefully transfer data using this method.

Table 4: Optional MultiSpeak Methods

Method Name	Importance to User	Supported by Server[1] (MR)	Supported by Client[2] (CB)	Verified Inter-operable[3]
CancelDisconnectedStatus	CB Notifies MR of Meters that should be removed from disconnected status.(i.e. made active).	X	X	X
CancelPlannedOutage	Notify MR of Cancellation of Planned Outage Given a List of MeterNumbers			
CancelUsageMonitoring	Notify MR of Cancellation Of Zero Usage Monitoring.(ie Move Ins).	X	X	X
GetDomainMembers	enable systems to exchange information about application-specific or installation-specific lists of information, such as the lists of counties for this installation or the list of serviceStatusCodes used by the server			
GetDomainNames	enable systems to exchange information about application-specific or installation-specific lists of information, such as the lists of counties for this installation or the list of serviceStatusCodes used by the server.			
GetModifiedAMRMeters	Returns all meters that support AMR and that have been modified since the specified sessionID			
GetReadingsByBillingCycle	Returns all required Reading Data for a given BillingCycle and Date Range			
InitiateDisconnectedStatus	CB Notifies MR of Meters that have been disconnected and no AMR reading is expected.	X	X	X
InitiatePlannedOutage	Notify MR of Planned Outage Meters Given a List of MeterNumbers and Start and End Dates of the Outage			
InitiateUsageMonitoring	Notify MR of Meters Where Zero Usage is Expected.(ie Move outs).	X	X	X

1) Supported by Server means that the server has demonstrated in some interoperability test (not necessarily with this client) that it can support the method.
2) Supported by Client means that the client has demonstrated in some interoperability test (not necessarily with this server) that it can call the method.
3) Verified Interoperable means that both the client and server have demonstrated in this interoperability test that they can usefully transfer data using this method.

5 Summary of Interoperability Test Results (#2A CB>MR)

PRODUCTS: Sensus RNI and NISC iVUE

Table 1: Recommended MultiSpeak Methods

Method Name	Importance to User	Supported by Server[1] (CB)	Supported by Client[2] (MR)	Verified Inter-operable[3]
GetMeterByAccountNumber	Returns the requested Meter(s) data given Account Number.	X		
GetMeterByMeterId	Returns the requested Meter data given meterID	X		
GetMeterByServLoc	Returns the requested Meter(s) data given Service Location.	X		
GetMeterByMeterNo	Returns the requested Meter data given Meter Number.	X		
GetAllCustomers	Returns all required customer data for all customers	X		
GetServiceLocationByCustId	Returns the requested Service Location data given Customer ID.	X		
GetServiceLocationByServLoc	Returns the requested Service Location data given Service Location ID	X		
GetCustomerByCustId	Returns the requested Customer if it exists.	X		
GetAllServiceLocations	Returns all required Service Location data for all Service Locations	X	X	
GetMeterByCustID	Returns the requested Meter(s) data given Customer ID	X		
GetServiceLocationByAccountNumber	Returns the requested Service Location data given Account Number	X		
GetAllMeters	Returns all required Meter data for all Meters	X	X	X
GetMethods	Requests a list of web service methods supported by the AMR application	X	X	X
PingURL	Queries status of the AMR application.	X	X	X

1) Supported by Server means that the server has demonstrated in some interoperability test (not necessarily with this client) that it can support the method.
2) Supported by Client means that the client has demonstrated in some interoperability test (not necessarily with this server) that it can call the method.
3) Verified Interoperable means that both the client and server have demonstrated in this interoperability test that they can usefully transfer data using this method.

Table 2: Optional MultiSpeak Methods

Method Name	Importance to User	Supported by Server[1] (CB)	Supported by Client[2] (MR)	Verified Inter-operable[3]
GetCustomerByName	Returns the requested Customer(s) data given First and Last name			
GetServiceLocationByServiceType	Returns the requested Service Location(s) data given the Service Type			
GetModifiedMeters	Returns all required Meter data for all Meters that have been modified since the specified sessionID			
GetDomainNames	Enables systems to exchange information about application-specific or installation-specific lists of information, such as the lists of counties for this installation or the list of serviceStatusCodes used by the server	X		
GetCustomerByMeterNo	Returns the requested Customer data given a Meter Number	X		
GetCustomerByDBAName	Returns the requested Customer given the Doing Business As (DBA) name			
ModifyCBDataForMeter	Allows MR to Modify CB data for a single Meter.			
GetServiceLocationByLoadGroup	Returns the requested Service Location(s) data for a given Load Group			
GetServiceLocationByGridLocation	Returns the requested Service Location(s) data given a single Grid Location	X		
GetServiceLocationByMeterNo	Returns the requested Service Location data given the meter number of a meter served at that location	X		
GetServiceLocationByShutOffDate	Returns the requested Service Location(s) data given the Service ShutOff Date			
HistoryLogChangedNotification	MR Notifies CB of a change in the History Log by sending the changed historyLog object			
ReadingChangedNotification	MR Notifies CB of a change in Meter Reads by sending the changed meterRead objects	X		
GetDomainMembers	Enable systems to exchange information about application-specific or installation-specific lists of information, such as the lists of counties for this installation or the list of serviceStatusCodes used by the server	X		
GetModifiedServiceLocations	Returns all required Service Location data for all Service Locations that have been modified since the specified sessionID			
GetServiceLocationByPhaseCode	Returns the requested Service Location(s) data given the Phase			
GetServiceLocationByServiceStatus	Returns the requested Service Location(s) data given the Service Status			
ModifyCBDataForCustomer	Allow MR to Modify CB data for a specific customer			
GetModifiedCustomers	Returns all required customer data for all customers that have been modified since the specified sessionID			
GetMeterByAMRType	Returns the requested Meter(s) data given AMR Type.			

ModifyCBDataForServiceLocation	Allow MR to Modify CB data for the Service Location			

1) Supported by Server means that the server has demonstrated in some interoperability test (not necessarily with this client) that it can support the method.
2) Supported by Client means that the client has demonstrated in some interoperability test (not necessarily with this server) that it can call the method.
3) Verified Interoperable means that both the client and server have demonstrated in this interoperability test that they can usefully transfer data using this method.

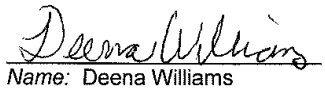
Certified by:

For Sensus:

_____ Sr. Software Development Manager
Name: Jatin Hansoty Title

Date : 12/19/2012

For National Information Systems Cooperative:



Name: Deena Williams

Sr. Software Specialist

_____ _____
Name: Deena Williams Title

12/19/2012
Date : _____

Assertions Verified by:

_____ MultiSpeak Testing Agent
Name: Hannu Huhdanpaa Title

UISOL,Inc.. .
Testing Agent
12/19/2012
Date : _____

Disclaimer:

The assertions made in this document are statements of the vendors offering the two products listed above. The Testing Agent has observed the software performing the tasks described in these vendor assertions.

Neither NRECA, Cornice Engineering, Inc. (MultiSpeak Technical Coordinator), nor UISOL, Inc, acting on behalf of NRECA, makes any warranty or guarantee that the software will perform as described in this assertion when installed at any specific utility. Furthermore, neither NRECA, Cornice Engineering, Inc., nor UISOL, Inc. makes any warranty or guarantee that the software described will be suitable for any specific purpose or need.

As used herein, the word *verify* shall mean an expression of the Testing Agent's professional opinion to the best of its information, knowledge and belief, and does not constitute a warranty or guarantee by NRECA or the Testing Agent.