CRC®
Cooperative Response Center, Inc.

MILSOFT
Utility Solutions

MultiSpeak Version 4.1.6 Interoperability Assertion

Vendor: Milsoft Utility Solution's DisSPatch Outage Management System and CRC's CRCLink Contact Center Application

Interfaces: Outage Analysis (OA), Notification Server (NOT)

Vendor(s)	Product	Product Version	Role	Web Client Interfaces	Web Server Interfaces
Cooperative Response Center (CRC)	CRCLink	2015.1	OA	OA→OA NOT→NOT	NOT→NOT
Milsoft Utility Solutions, Inc.	Milsoft MultiSpeak Service	8.2	OA NOT	NOT→NOT	OA→OA NOT→NOT

Summary:

Web Service interfaces using MultiSpeak standards were developed in order to provide CRC and Milsoft DisSPatch customers a way to keep the system databases synchronized without having to update and maintain data in both systems through manual processes. The integration also turns on two way real time requests for current outage status, outage detection, outage alerts and other real time events. Real time data can be displayed from within CRC's CRCLink and Milsoft DisSPatch outage management screens.

Prerequisites:

Account Numbers, Meter Numbers, and Service Locations or Service Location Map identifiers in CRCLink must match Milsoft DisSPatch circuit model element names. There must be a working network connection between the MultiSpeak web services of CRC and Milsoft. The interface can be enabled by exchanging the URL's, user names and passwords of each vendors MultiSpeak web service along with any other required configuration specific to the software of each vendor. The CRCLink NOT Client will register with the Milsoft DisSPatch Server to begin receiving notifications from the outage management system.

Specific Vendor Assertions:

1) CRCLink will register with the Milsoft MultiSpeak Notification Server and subscribe for receiving notification messages from the Milsoft DisSPatch system.

Importance to user: This provides CRCLink with the ability to subscribe for receiving updates from the Milsoft DisSPatch as they occur in real-time. CRCLink can also unsubscribe to receive updates when the software is stopped.

How Achieved: During initialization, CRCLink uses the RequestRegistrationID method to request a unique ID from the Milsoft MultiSpeak Service that can be used to make a RegisterForService request and subscribe to receive notification messages for the methods ODEventNotification, OutageEventChangedNotification, CHEventNotification and CustomersAffectedByOutageNotification. Once CRCLink has received a registration ID, it can re-use the ID to check if a registration still exists by making a GetRegistrationInfoByID request to the Milsoft MultiSpeak Service and verify that a previous subscription is still valid. This is used in the case where the CRCLink software may have been restarted and a subsequent registration is unnecessary. Should the CRCLink software be stopped, due to uninstalling software or discontinuing services, an UnregisterForService request is issued to the Milsoft MultiSpeak Service by CRCLink to unregister a previous subscription. Unregistering will cause the Milsoft MultiSpeak Service to stop sending notification messages to CRCLink, thus keeping it from attempting to establish connections to CRCLink unnecessarily.

2) CRCLink will pull the updated problem code list from Milsoft DisSPatch

Importance to user: This will provide CRCLink with an updated list of call problem codes from Milsoft DisSPatch.

How Achieved: CRCLink pulls the list of problem codes from Milsoft DisSPatch daily through the GetDomainMembers method with a domain name of “Call Problem Codes” to update the CRCLink problem code list.

3) CRCLink will pull the updated outage reason code list from Milsoft DisSPatch

Importance to user: This will provide CRCLink with an updated list of reason codes from Milsoft DisSPatch. This can include cause codes, equipment codes, weather codes, voltage codes and other custom codes.

How Achieved: CRCLink pulls the list of outage reason codes from Milsoft DisSPatch daily through the GetOutageReasonCodes method to update the CRCLink outage reason code list.

4) CRCLink will synchronize the list of configured crews from Milsoft DisSPatch

Importance to user: This will provide CRCLink with an updated list of active crews from Milsoft DisSPatch.

How Achieved: CRCLink pulls the list of active crews from Milsoft DisSPatch daily through the GetAllCrews method to update the CRCLink crew list.

5) CRCLink will synchronize all currently active outage calls from Milsoft DisSPatch

Importance to user: This will provide CRCLink with a complete list of outage calls that exist in Milsoft DisSPatch on initialization.

How Achieved: On initialization or to resynchronize the two systems when the connection is lost, CRCLink will send a GetAllActiveCalls request to the Milsoft MultiSpeak Service to get the current list of active outage calls.

6) CRCLink will synchronize all currently active outage events from Milsoft DisSPatch

Importance to user: This will provide CRCLink with a complete list of outage events that exist in Milsoft DisSPatch on initialization.

How Achieved: On initialization or to resynchronize the two systems when the connection is lost, CRCLink will send a GetAllActiveOutageEvents request to the Milsoft MultiSpeak Service to get the current list of active outage events.

7) CRCLink will notify Milsoft DisSPatch of new outage calls

Importance to user: This will provide Milsoft DisSPatch with new outage calls from CRCLink.

How Achieved: CRCLink sends an ODEventNotification request to the Milsoft MultiSpeak Service with a verb of New on outageDetectionEvent objects to update Milsoft DisSPatch when a new outage call is entered in CRCLink.

8) Milsoft DisSPatch will notify CRCLink of new outage calls

Importance to user: This will provide CRCLink with new outage calls from Milsoft DisSPatch.

How Achieved: The Milsoft MultiSpeak Service sends an ODEventNotification request with a verb of New on outageDetectionEvent objects to update CRCLink when an outage call enters the DisSPatch system.

9) CRCLink will notify Milsoft DisSPatch of updated outage calls

Importance to user: This function will provide Milsoft DisSPatch with updates for outage calls from CRCLink.

How Achieved: CRCLink sends an ODEventNotification request to the Milsoft MultiSpeak Service with a verb of Change on outageDetectionEvent objects to update Milsoft DisSPatch when an outage call is updated for a consumer in CRCLink.

10) Milsoft DisSPatch will notify CRCLink of updated outage calls

Importance to user: This will provide CRCLink with updates to outage calls from Milsoft DisSPatch.

How Achieved: The Milsoft MultiSpeak Service sends an ODEventNotification request with a verb of Change to update CRCLink when an outage call is updated in Milsoft DisSPatch.

11) CRCLink will notify Milsoft DisSPatch of new non-outage calls

Importance to user: This will provide Milsoft DisSPatch with new non-outage calls from CRCLink.

How Achieved: CRCLink sends a CHEventNotification request to the Milsoft MultiSpeak Service with a verb of New to update Milsoft DisSPatch when a non-outage call is entered in CRCLink.

12) Milsoft DisSPatch will notify CRCLink of new non-outage calls

Importance to user: This will provide CRCLink with new non-outage calls from Milsoft DisSPatch.

How Achieved: The Milsoft MultiSpeak Service sends a CHEventNotification request with a verb of New to update CRCLink when a non-outage call enters the DisSPatch system.

13) CRCLink will notify Milsoft DisSPatch of updated non-outage calls

Importance to user: This will provide Milsoft DisSPatch with updates for non-outage calls from CRCLink.

How Achieved: CRCLink sends a CHEventNotification request to the Milsoft MultiSpeak Service with a verb of Change on customerCall objects to update Milsoft DisSPatch when a non-outage call is modified in CRCLink.

14) Milsoft DisSPatch will notify CRCLink of updated non-outage calls

Importance to user: This will provide CRCLink with updates for non-outage calls from Milsoft DisSPatch.

How Achieved: The Milsoft MultiSpeak Service sends a CHEventNotification request with a verb of Change to update CRCLink when a non-outage call is modified in DisSPatch.

15) CRCLink will notify Milsoft DisSPatch when a crew is assigned to an outage

Importance to user: This will provide Milsoft DisSPatch with new crew assignments for outages from CRCLink.

How Achieved: CRCLink sends an AssignCrewsToOutage request to the Milsoft MultiSpeak Service to update Milsoft DisSPatch when a crew assignment is entered for an outage in CRCLink.

16) CRCLink will notify Milsoft DisSPatch when and outage crew is unassigned from an outage

Importance to user: This will provide Milsoft DisSPatch with updates when crews are unassigned from outages from CRCLink.

How Achieved: CRCLink sends a UnassignCrewsFromOutage request to the Milsoft MultiSpeak Service to update Milsoft DisSPatch when a crew is unassigned from an outage in CRCLink.

17) Milsoft DisSPatch will notify CRCLink of outage crew changes

Importance to user: This will provide CRCLink with updates when crews are assigned to an outage or unassigned from an outage when crews are assigned to an outage or unassigned from an outage that were performed from Milsoft DisSPatch.

How Achieved: Milsoft DisSPatch sends an OutageEventChangedNotification with verb Change on the outageEvent object and an updated crews dispatched list to update CRCLink when crews are assigned to an outage or unassigned from an outage in Milsoft DisSPatch.

18) Milsoft DisSPatch will send new outages to CRCLink

Importance to user: This will provide CRCLink with new outages from Milsoft DisSPatch. This includes the predicted device, outage status, crew assignments, dispatch time and restoral information for every outage event.

How Achieved: The Milsoft MultiSpeak Service sends an OutageEventChangedNotification request with a verb of New on outageEvent objects to CRCLink when an outage event is created in Milsoft DisSPatch. All outage events will originate from the outage management system.

19) Milsoft DisSPatch will send updated outage information to CRCLink

Importance to user: This will provide CRCLink with updated outage information from Milsoft DisSPatch. This will include the predicted device, outage status, crew assignments, first dispatch time, estimated time to restoration(ETOR) and restoral information(restoral time, outage reason codes and comments) for every outage event.

How Achieved: The Milsoft MultiSpeak Service sends an OutageEventChangedNotification request with a verb of Change on outageEvent objects to CRCLink when an outage event is updated in Milsoft DisSPatch.

20) CRCLink will notify Milsoft DisSPatch of outage restorations

Importance to user: This will provide Milsoft DisSPatch with updated outage restorations for outages from CRCLink. This will include RUS codes and restoral comments.

How Achieved: CRCLink sends a RestoreOutage request to the Milsoft MultiSpeak Service to update Milsoft DisSPatch when an outage is restored in CRCLink. Upon receiving the RestoreOutage request, the outage restoration operation will be performed for the corresponding outage event in Milsoft DisSPatch along with updating any comments and RUS codes for the outage event.

21) Milsoft DisSPatch will notify CRCLink of outage restorations

Importance to user: This will provide CRCLink with updated outage restorations for outages from Milsoft DisSPatch. This will include any RUS codes and restoral comments that were set for the outage.

How Achieved: Milsoft DisSPatch sends an OutageEventChangedNotification request with a verb of Change on the outageEvent object and an outageStatus of Restored to update CRCLink when an outage is restored in Milsoft DisSPatch.

22) CRCLink will notify Milsoft DisSPatch of closed non-outage calls

Importance to user: This will close non-outage calls in Milsoft DisSPatch when they are closed in CRCLink.

How Achieved: CRCLink sends a CloseCalls request to the Milsoft MultiSpeak Service to update Milsoft DisSPatch when a non-outage call is closed.

23) Milsoft DisSPatch will notify CRCLink of closed non-outage calls

Importance to user: This will close non-outage calls in CRCLink when they are discarded in Milsoft DisSPatch.

How Achieved: Milsoft DisSPatch sends a CHEventNotification with a verb of Delete on customerCall objects to update CRCLink when a non-outage call is discarded.

24) CRCLink will notify Milsoft DisSPatch when an outage is discarded

Importance to user: This will update Milsoft DisSPatch when an outage is discarded in CRCLink. In CRCLink this is achieved by pressing the "Not an Outage" button.

How Achieved: CRCLink sends a DiscardOutage request to the Milsoft MultiSpeak Service to update Milsoft DisSPatch when an outage is discarded in CRCLink.

25) Milsoft DisSPatch will notify CRCLink when an outage is discarded

Importance to user: This will update CRCLink when an outage is discarded in Milsoft DisSPatch.

How Achieved: Milsoft DisSPatch sends an OutageEventChangedNotification request with a verb of Delete on outageEvent objects to update CRCLink when an outage is discarded in Milsoft DisSPatch.

26) CRCLink will notify Milsoft DisSPatch when an outage call is discarded

Importance to user: This will update Milsoft DisSPatch when an outage call is discarded in CRCLink.

How Achieved: CRCLink sends a CloseCalls request to the Milsoft MultiSpeak Service to update Milsoft DisSPatch when an outage call is discarded in CRCLink.

27) Milsoft DisSPatch will notify CRCLink when an outage call is discarded

Importance to user: This will update CRCLink when an outage call is discarded in Milsoft DisSPatch.

How Achieved: Milsoft DisSPatch sends an ODEventNotification request with a verb of Delete on outageEvent objects to update CRCLink when an outage is discarded in DisSPatch.

28) CRCLink can pull the list of up line devices from Milsoft DisSPatch for a predicted outage

Importance to user: During the process of confirming that a device is out of power from CRCLink, it may be that a device up line from the predicted device is the correct device causing the power outage. This function will provide a list of up line devices for a user in CRCLink to choose from when confirming which device that is out of power.

How Achieved: CRCLink pulls the list of up line devices by sending a GetUplineCircuitElements request to the Milsoft MultiSpeak Service. CRCLink will use the circuit element name of the device that is currently predicted out of power on an outage to send in the GetUplineCircuitElements request. The Milsoft MultiSpeak Service will reply with a list of circuit elements that are up line from the circuit element specified in the request.

29) CRCLink can confirm or change the device that is predicted to be out of power in Milsoft DisSPatch

Importance to user: This will allow CRCLink to confirm or change the device that is predicted to be out of power for an outage.

How Achieved: CRCLink sends a SetOutageElementStatus request to the Milsoft MultiSpeak Service with the name of a circuit element and one of 3 values for status in order to confirm or change the prediction in Milsoft DisSPatch. A status of VerifiedOpen is sent to confirm that a device is out of power. A status of VerifiedClosedNoPower is sent to tell Milsoft DisSPatch to re-predict upstream from the currently predicted device. A status VerifiedClosedWithPower is sent to tell Milsoft DisSPatch to re-predict downstream from the currently predicted device. This MultiSpeak method performs the same operations as using the “Set Verify Status” dialog on a circuit element in DisSPatch. This interface does not use this MultiSpeak method to operate on individual phases for multi-phase devices and will set all existing phases of the device to the same status.

30) CRCLink can set or update the estimated time to restoration for an outage in Milsoft DisSPatch

Importance to user: This will allow a CRCLink user to set or update the estimated time of restoration for an outage in DisSPatch.

How Achieved: CRCLink sends an UpdateOutageETOR request to the Milsoft MultiSpeak Service to update the estimated time to restoration (ETOR) of an outage event in Milsoft when the ETOR for the outage is updated in CRCLink.

31) Milsoft DisSPatch can provide CRCLink with a list of service locations that are affected by an outage event.

Importance to user: This will provide CRCLink with the complete list of service locations that are currently affected by outage events as recorded in Milsoft DisSPatch. Users in CRCLink will see whether the service location is in an existing outage or not before entering a new outage call.

How Achieved: The Milsoft MultiSpeak Service sends a CustomersAffectedByOutageNotification request to CRCLink when Milsoft DisSPatch records that service locations are being affected by an outage event. As the number of service locations affected by outage events grows or shrinks, subsequent requests are sent to CRCLink to provide the most up to date list of service locations affected by active outage events.

Products: CRC CRCLink and Milsoft MultiSpeak Service

Summary of Interoperability Test Results
OA Server (Milsoft MultiSpeak Service) → OA Client (CRCLink)

Table 1
Required MultiSpeak Methods

Method Name	Importance to User	Supported by Server[1] (OA)	Supported by Client[2] (OA)	Verified Inter-operable[3]
GetMethods	Client requests a list of methods supported by the server.	X	X	X
PingURL	Client verifies that the server is running and reachable.	X	X	X

Table 2
Non-Required MultiSpeak Methods

Method Name	Importance to User	Supported by Server[1] (OA)	Supported by Client[2] (OA)	Verified Inter-operable[3]
GetDomainMembers	Requests the members of a given domain (type of fixed information, such as all of the Problem Codes in the database).	X	X	X
GetDomainNames	Requests the domains (lists of fixed information, such as the Problem Codes, or acceptable status codes).	X	X	X
GetAllActiveCalls	Returns all active calls that have been processed by the outage management system.	X	X	X
GetAllActiveOutageEvents	Returns all active outage event information.	X	X	X
AssignCrewsToOutage	Assign crews to an outage event.	X	X	X
UnassignCrewsFromOutage	Remove crews from an outage event.	X	X	X
GetUplineCircuitElements	Returns all information for circuit elements in the shortest route to source from the given line section or node (eaLoc).	X	X	X
SetOutageElementStatus	Assigns an outage status to a circuit element.	X	X	X
RestoreOutage	Restores an outage event.	X	X	X
GetOutageReasonCodes	Requests the list of outage reason codes (list of RUS or other standard codes).	X	X	X
UpdateOutageETOR	Updates the estimated time to restoration for an outage event.	X	X	X
GetAllCrews	Requests the list of crews.	X	X	X
CloseCalls	Publisher notifies OA of a list of customer calls to close out.	X	X	X
DiscardOutage	Discards an outage event.	X	X	X
ODEventNotification	Publisher notifies OA of a change in an outage detection device by sending an array of changed OutageDetectionDevice objects.	X	X	X
CHEventNotification	Publisher notifies OA of non-outage events by sending an array of customerCall objects.	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.

Products: CRC CRCLink and Milsoft MultiSpeak Service

Summary of Interoperability Test Results
NOT Server (Milsoft MultiSpeak Service) → NOT Client (CRCLink)

Table 3
Required MultiSpeak Methods

Method Name	Importance to User	Supported by Server[1] (NOT)	Supported by Client[2] (NOT)	Verified Inter-operable[3]
GetMethods	Client requests a list of methods supported by the server.	X	X	X
PingURL	Client verifies that the server is running and reachable.	X	X	X
RequestRegistrationID	Subscriber requests of the publisher a unique registration ID that would subsequently be used to refer unambiguously to that specific subscription.	X	X	X
GetPublishMethods	Subscriber requests list of methods to which this server can publish information.	X	X	X
RegisterForService	Establishes a subscription using a previously requested registration ID.	X	X	X
GetRegistrationInfoByID	Requests the return of existing registration information for a specific registration ID.	X	X	X
UnregisterForService	Deletes a previously established subscription.	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.

Products: CRC CRCLink and Milsoft MultiSpeak Service

Summary of Interoperability Test Results
NOT Server (CRCLink) → NOT Client (Milsoft MultiSpeak Service)

Table 4
Required MultiSpeak Methods

Method Name	Importance to User	Supported by Server[1] (NOT)	Supported by Client[2] (NOT)	Verified Inter-operable[3]
GetMethods	Client requests a list of methods supported by the server.	X	X	X
PingURL	Client verifies that the server is running and reachable.	X	X	X

Table 5
Non-Required MultiSpeak Methods

Method Name	Importance to User	Supported by Server[1] (NOT)	Supported by Client[2] (NOT)	Verified Inter-operable[3]
ODEventNotification	Publisher notifies subscriber of a change in OutageDetectionEvents by sending an array of changed OutageDetectionEvent objects.	X	X	X
CHEventNotification	Publisher notifies subscriber of non-outage events by sending an array of customerCall objects.	X	X	X
OutageEventChangedNotification	Publisher notifies subscriber of a change in outage events by sending an array of changed OutageEvent objects.	X	X	X
CustomersAffectedByOutageNotification	Publisher Notifies subscriber of changes to the customers affected by an outage event.	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.

Certified by:

For Cooperative Response Center:

Name: Paul Thompson

Title: VP of Administration / CFO

Date: 6/3/2015

For Milsoft Utility Solutions, Inc.:

Name: Luis R. Malavé

Title: Board Director, Vice President, Milsoft

Date: 6/3/2015

Assertions Verified by:

Name: Hannu Huhdanpaa

Title: MultiSpeak Testing Agent, UISOL

UISOL Inc.
Testing Agent
Date: 6/3/2015

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.