

---

# MULTISPEAK VERSION 3.0 INTEROPERABILITY ASSERTION

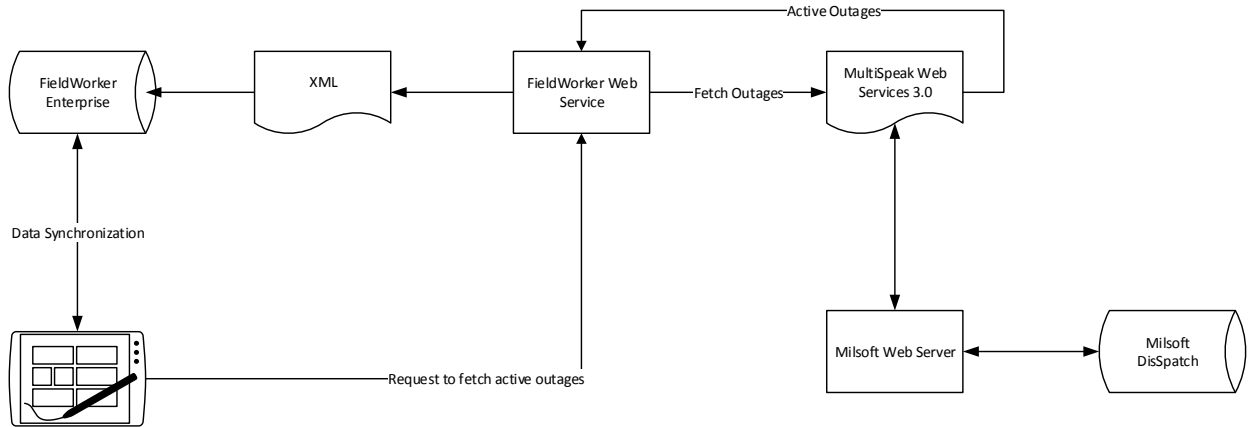
Statement of Interoperable Functionality Between:

Vendor(s)	Product	Product Version	Role	Batch Interface	Web Client Interfaces	Web Server Interfaces
FieldWorker Products Limited	FieldWorker Enterprise	9	DGV	N/A	OA→DGV	DGV→OA
Milsoft	Milsoft Web Server DisSPatch	8.1	OA	N/A	DGV→OA	OA→DGV

## SUMMARY

Milsoft's DisSPatch Outage Management System is able to pass active outage information to FieldWorker Enterprise via the MultiSpeak v3.0 Build ac OA web service through which data is then propagated to FieldWorker's built-in XML services. Completed outages are then sent back via the same conduit in reverse. The cycle starts with the FieldWorker Client calling its own Outage web service that communicates with FieldWorker Enterprise and the Milsoft Web Server. The FieldWorker Outage web service acts as an intermediary between FieldWorker Enterprise and the Milsoft Web Server. The Milsoft Web Server acts as an intermediary between the FieldWorker Outage web service and DisSPatch. There are two ways in which outage information is queried. The default behavior is the following:

1. The FieldWorker client invokes a web service call to the FieldWorker Outage web service.
2. The FieldWorker Outage web service, receiving this call request from the client, itself calls the Milsoft Web Server to query active outages. These active outages that were assigned to the calling user are then pushed to FieldWorker Enterprise by the FieldWorker Outage web service, updating existing outages that were already pushed before. This updating behavior means that fields that once contained data that no longer contain data will also be reflected.
3. FieldWorker client synchronizes after the successful web service call, retrieving the pushed outages.
4. Should an error occur on the server, it passes the error information to the client. The client then displays this information in a dialog and will not synchronize. A similar message is also sent if there are no active outages.



**FIGURE 1: FETCHING ACTIVE OUTAGES**

Once the outage is restored, it can be sent back to Milsoft’s DisPatch Outage Management System in the following manner:

1. The user clicks on a button in FieldWorker Client which will synchronize the data from the device to FieldWorker Enterprise.
2. Upon successful synchronization the client will then invoke outage restoration web service method from FieldWorker Outage web service.
3. FieldWorker Outage web service will retrieve all restored outages for the specified user from FieldWorker Enterprise that have not yet been sent to DisPatch and will call RestoreOutage web service method on Milsoft’s Web Server for each outage that needs to be restored. OutageReasonChangedNotification web service method will also be called for each of the outage’s reason codes.
4. The user will receive a confirmation back as to which outages have been successfully restored. The outage information on user’s device will no longer be editable after it has been successfully sent to DisPatch Outage Management System.

Alternatively, the solution can be configured to behave like the following:

1. Fieldworker web service is configured to continually call the Milsoft Web Server to query active outages at a predefined interval. These active outages are then pushed to FieldWorker Enterprise during each cycle, updating existing ones that were already pushed before. This updating behavior means that fields that once contained data that no longer contain data will also be reflected. A mapping between FieldWorker users and Milsoft crews is established via a configuration file so that FieldWorker web service will correctly push outages to required users.
2. Restoration of outages can be configured in the same manner where the web service will poll at a predefined interval for any restored outages that have not yet been sent to DisPatch. Those outages will be sent to DisPatch Outage Management System and will be marked in FieldWorker Enterprise so that they will not be sent over again.
3. FieldWorker client is configured to auto-synchronize at a predefined interval, thus ensuring that any new outages will be reflected on the client.



## FIELDWORKER WILL SYNCHRONIZE ACTIVE OUTAGE DETAILS

- **Importance to user:** Mobile users will have access to the outage details for each active outage on their mobile devices without the need to consult the back-office personnel.
- **How Achieved:** FieldWorker retrieves the current outage details for all active outages by calling the GetOutageEvent web service method on the Milsoft Web Server during the time that the GetActiveOutages method is called.

## FIELDWORKER WILL SYNCHRONIZE OUTAGE REASON (CAUSE) CODES

- **Importance to user:** The DisSPatch software acts as the main backend system for managing outages. It maintains a user-defined set of Outage Reason codes which are used as meta-data for each outage. During a lifecycle of each outage, an Outage Reason Code will be assigned by the dispatcher using Milsoft's DisSPatch system to correctly identify the outage state. FieldWorker can extract these specific reason codes from DisSPatch so that they are available for display to the mobile users out in the field.
- **How Achieved:** FieldWorker imports the Outage Reason codes by calling GetOutageReasonCodes web service method provided by the MultiSpeak v3.0 interface from the Milsoft Web Server. This is performed automatically when the 'Category' list containing the reason codes does not yet exist in FieldWorker.

## FIELDWORKER WILL RESTORE ACTIVE OUTAGES

- **Importance to user:** FieldWorker allows field users to restore outages out in the field, thus speeding up the workflow of outage management without the need of the field user calling in to the back-office. The user is allowed to restore any active outage only once.
- **How Achieved:** Once the data is synchronized from the mobile device to FieldWorker server, FieldWorker web service will check for any outages that have been restored in FieldWorker but not yet been sent to DisSPatch. These outages will then be restored by calling RestoreOutage web service method on Milsoft's web server. If the web service method was initiated as a result of user action then only restored outages assigned to that specific user will be restored using RestoreOutage web service method. Otherwise if the call to the web service methods was a result of automatic polling interval, then all restored outages in FieldWorker will be restored.

## FIELDWORKER WILL UPDATE OUTAGE REASON (CAUSE) CODES

- **Importance to user:** FieldWorker allows the field user to enter and modify outage reason codes for any outage that has not yet been closed. This allows the user to correct information about the outage if it was entered incorrectly by the backend office staff or enter additional reason codes that were missed when the outage was initially created. Any active outage can only have one reason code from each available category.
- **How Achieved:** FieldWorker web service will check for changes for any outages currently in FieldWorker and if any of the outages contain changes to their outage reason codes then those codes will be updated in DisSPatch using OutageReasonChangedNotification web service method for that specific outage. This action is done when restoring outages (manually via user invocation) and also automatically at a predefined interval via polling.

## SUPPORT FOR GENERIC METHODS

- **Importance to user:** MultiSpeak v3.0 specification states that both service ends must implement GetMethods and PingURL web service methods to allow each side to verify the existence and validity of the other as well as provide with a list of supported MultiSpeak methods that can be called.
- **How achieved:** Both the Milsoft Web Server and the FieldWorker Outage web service implement GetMethods and PingURL web service methods as required by the MultiSpeak v3.0 specification.

**Products: FieldWorker Enterprise and Milsoft Web Server  
Summary of Interoperability Test Results (OA→DGV)**

**Table 1  
MultiSpeak Methods (OA)**

Method Name	OA-DGV	Importance to User	Supported by Server <sup>1</sup> (OA)	Supported by Client <sup>2</sup> (DGV)	Verified Inter-operable <sup>3</sup>
AVLChangedNotification	OPT	Publisher notifies subscriber of new AVL events by sending an AVLMessage object.	<b>X</b>		
AssessmentLocationChangedNotification	OPT	Publisher notifies Subscriber of new assessmentLocation(s).	<b>X</b>		
AddRemarksToOutage	OPT	Allows a system operator to add a remark to an outage event.			
AssignCrewsToAssessment	OPT	Assigns crews to an assessment given the assessmentID (objectID of the assessment object).			
AssignCrewsToOutage	OPT	Assigns crews to an outage given the outage event ID.	<b>X</b>		
DiscardOutage	OPT	This method allows a dispatcher or operator to discard an outage that has been created erroneously or which was generated for training purposes.	<b>X</b>		
GetActiveOutages	OPT	Synchronize current list of currently active outages between the operations center and field workers.	<b>X</b>	<b>X</b>	<b>X</b>
GetAllActiveCalls	OPT	Returns all active calls that have been processed by the outage management system in the form of an outageDetectionLogList.	<b>X</b>		
GetAllActiveOutageEvents	OPT	Returns all of the outageEvent(s) for all active outages.	<b>X</b>		
GetAllCrews	OPT	Returns all active crews that are available for dispatching if the parameter activeOnly is	<b>X</b>		

		set to be true, otherwise all crews are returned.			
GetCallsReceivedOnOutage	OPT	Returns all calls that have been processed by the outage management system in the form of an outageDetectionLogList.			
GetCircuitElementsNearLatLong	OPT	Returns an array of circuitElements that lie within the distance tolerance of the location expressed in latitude and longitude.	<b>X</b>		
GetCircuitElementStatus	OPT	Returns the outage event, if any, associated with a circuitElement given the objectRef of the circuitElement.			
GetCustomersAffectedByOutage	OPT	Returns all customers that are affected by a specific outage of interest, given the outageEventID.	<b>X</b>		
GetCustomerCallHistory	OPT	Returns all calls that have been processed by the outage management system for a given customer account and service location in the form of an outageDetectionLogList.			
GetCustomerCallsOnServiceLocation	OPT	Returns all calls that have been processed by the outage management system for a given service location in the form of an outageDetectionLogList.			
GetOutageByCircuitElement	OPT	Returns the outage event, if any, associated with a circuitElement.			
GetOutageDurationEvents	OPT	Returns all outage duration events that have been processed by the outage management system for a given outage identified by outageEventID, returned in the form of an outageDurationEventList.	<b>X</b>		
GetOutageEvent	OPT	Synchronize outage details between the operations center and field workers.	<b>X</b>	<b>X</b>	<b>X</b>

GetOutageHistoryOnServiceLocation	OPT	Returns all outage duration events that have been processed by the outage management system for a given service location in the form of an outageDurationEventList.	X		
GetCustomerOutageHistory	OPT	Returns all outage duration events that have been processed by the outage management system for a given customer account and service location in the form of an outageDurationEventList.	X		
GetOutageEventsByDate	OPT	Returns outage events within a range of times.			
GetOutageEventStatus	OPT	Returns the current status of an outage event, given the outage event ID.	X		
GetOutageEventStatusByOutageLocation	OPT	Returns the current status of an outage event, given the outage location.	X		
GetOutagedODDevices	OPT	Returns the outageDetectionDevices that are currently experiencing outage conditions.			
GetOutageReasonCodes		Returns the list of outage reason codes used by the OMS implementation.	X	X	X
OutageReasonChangedNotification	OPT	Publisher notifies Subscriber of the causes and other information related to an outage event.	X	X	X
RestoreOutage	OPT	Publisher notifies OA of an outage that should be denoted as being restored, given an outage event ID.	X	X	X
SetOutageElementStatus	OPT	This method allows a dispatcher or operator to verify or restore any circuit element by phase.			
UnassignCrewsFromAssessment	OPT	Unassigns crew(s) from an assessment given the assessmentID (objectID of the assessment object).			
UnassignCrewsFromOutage	OPT	Unassigns crew(s) from an outage given the outage	X		



		event ID.			
UnassignOutagesFromCrew	OPT	Unassigns outages(s) from a crew given the crew ID.	<b>X</b>		
UpdateOutageETOR	OPT	Publisher notifies OA that the estimated time to restoration (ETOR) should be modified.			
Discovery					
DomainMembersChangedNotification	OPT	This method permits a client to have changed information on domain members published to it using a previously arranged subscription, set up using the RegisterForServiceMethod.			
DomainNamesChangedNotification	OPT	This method permits a client to have changed information on domain names published to it using a previously arranged subscription, set up using the RegisterForServiceMethod.			
GetDomainMembers	OPT	The client requests from the server the members of a specific domain of information, identified by the domainName parameter, which are supported by the server.	<b>X</b>		
GetDomainNames	OPT	The client requests from the server a list of names of domains supported by the server.	<b>X</b>		
GetMethods	REQ	Requester requests list of methods supported by OA.	<b>X</b>	<b>X</b>	<b>X</b>
GetPublishMethods	REC	Requester requests list of methods to which this server can publish information.			
Subscription and Network Management					
GetRegistrationInfoByID	REC	This method requests the return of existing registration information (that is to say the details of what is subscribed on			

		this subscription) for a specific registrationID.			
RegisterForService	REC	This method establishes a subscription using a previously requested registrationID.			
RequestRegistrationID	REC	This service requests of the publisher a unique registration ID that would subsequently be used to refer unambiguously to that specific subscription.			
UnregisterForService	REC	This method deletes a previously established subscription (registration for service) that carries the registration identifier listed in the input parameter registrationID.			
PingURL	REQ	Requester pings URL of OA to see if it is alive. Returns errorObject(s) as necessary to communicate application status.	<b>X</b>	<b>X</b>	<b>X</b>

**REQ** – Inclusion is required for this interface.

**REC** – Inclusion is recommended for this interface.

**OPT** – Inclusion is optional for this interface.

<sup>1</sup>Supported by Server means that the server has demonstrated in some interoperability test (not necessarily with this client) that it can support the method.

<sup>2</sup>Supported by Client means that the client has demonstrated in some interoperability test (not necessarily with this server) that it can call the method.

<sup>3</sup>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: FieldWorker Enterprise and DisSPatch  
Summary of Interoperability Test Results (DGV→OA)**

**Table 2  
MultiSpeak Methods (DGV)**

Method Name	DGV -OA	Importance to User	Supporte d by Server <sup>1</sup> (DGV)	Supporte d by Client <sup>2</sup> (OA)	Verified Inter- operable <sup>3</sup>
AssessmentChangedNotification	OPT	Publisher notifies Subscriber of new assessment(s).			
AssessmentLocationChangedNotification	OPT	Publisher notifies Subscriber of new assessmentLocation(s).			
OutageEventChangedNotification	REC	Publisher notifies subscriber of a change in OutageEvent by sending an array of changed OutageEvent objects.			
Discovery					
DomainMembersChangedNotification	OPT	This method permits a client to have changed information on domain members published to it using a previously arranged subscription, set up using the RegisterForServiceMethod.			
<i>DomainNamesChangedNotification</i>	<i>OPT</i>	<i>This method permits a client to have changed information on domain names published to it using a previously arranged subscription, set up using the RegisterForServiceMethod.</i>			
GetDomainMembers	OPT	The client requests from the server the members of a specific domain of information, identified by the domainName parameter, which are supported by the server.			

GetDomainNames	OPT	The client requests from the server a list of names of domains supported by the server.			
GetMethods	REQ	Requester requests list of methods supported by DGV.	X	X	X
GetPublishMethods	REC	Requester requests list of methods to which this server can publish information.			
Subscription and Network Management					
GetRegistrationInfoByID	REC	This method requests the return of existing registration information (that is to say the details of what is subscribed on this subscription) for a specific registrationID.			
RegisterForService	REC	This method establishes a subscription using a previously requested registrationID.			
RequestRegistrationID	REC	This service requests of the publisher a unique registration ID that would subsequently be used to refer unambiguously to that specific subscription.			
UnregisterForService	REC	This method deletes a previously established subscription (registration for service) that carries the registration identifier listed in the input parameter registrationID.			
PingURL	REQ	Requester pings URL of DGV to see if it is alive. Returns errorObject(s) as necessary to communicate application status.	X	X	X

**REQ** – Inclusion is required for this interface.

**REC** – Inclusion is recommended for this interface.

**OPT** – Inclusion is optional for this interface.

<sup>1</sup>Supported by Server means that the server has demonstrated in some interoperability test (not necessarily with this client) that it can support the method.

<sup>2</sup>Supported by Client means that the client has demonstrated in some interoperability test (not necessarily with this server) that it can call the method.

<sup>3</sup>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 FieldWorker:**



**Rostislav Shnaper**

**Sr. Developer**

\_\_\_\_\_  
Name:

\_\_\_\_\_  
Title

December 2<sup>nd</sup>, 2013  
Date: \_\_\_\_\_

**For Milsoft Utility Solutions, Inc.:**



**Becky Paul**

**VP Product Management**

\_\_\_\_\_  
Name:

\_\_\_\_\_  
Title

December 2<sup>nd</sup>, 2013  
Date: \_\_\_\_\_

**Assertions Verified by:**



**Hannu Huhdanpaa**

**MultiSpeak Testing Agent**

\_\_\_\_\_  
Name:

\_\_\_\_\_  
Title

December 09, 2013

12/2/2013

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 Project Coordinator), nor UISOL, 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 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, Cornice Engineering, Inc., or the Testing Agent.