

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
QEI, Inc	TDMS Plus SCADA System	7.0.0	SCADA	-	OA→SCADA	SCADA→OA
Milsoft Utility Solutions	Milsoft Web Server	7.2	OA	-	-	OA→SCADA
Milsoft Utility Solutions	DisSPatch	7.2	OA	-	SCADA→OA	-

Summary:

QEI's TDMS Plus SCADA system is capable of providing SCADA data to Milsoft's DisSPatch Outage Management System in several different ways.

First of all, DisSPatch is able to request a list of SCADA points from TDMS Plus. The DisSPatch user can then map each SCADA point to the appropriate corresponding circuit element in DisSPatch. This is necessary in order to ensure that any SCADA data DisSPatch receives from TDMS Plus are associated with the correct circuit elements in the electrical model.

The interesting data TDMS Plus can send to DisSPatch are: breaker operations (known as SCADA status) and substation voltages and currents (known as SCADA analogs). When the Milsoft Web Server is used, TDMS Plus may publish a changed SCADA status or a changed SCADA analog directly to the outage system as soon as the event occurs. This will immediately appear in DisSPatch as a new outage event or a restoration.

Alternately, the DisSPatch user may request an individual SCADA status or a SCADA analog for some particular circuit element. TDMS Plus sends the requested information to DisSPatch and it is immediately available to the DisSPatch user for viewing or (in the case of fault current data) to run the Fault Locator utility.

Prerequisites:

For these interfaces to be successfully implemented, the TDMS Plus SCADA system must be configured as follows:

- A server subscription must be defined. This subscription specifies which status and analog points will be visible on the SCADA-OA web services server.
- A client subscription must be defined. This subscription specifies which status and analog points are published by the OA-SCADA web services client, as well as the frequency at which changes will be published. The client subscription also specifies one or multiple server URLs to which changes will be published. Multiple URLs are allowed to support a redundant server configuration. The client uses the pingURL method to determine which server is active.
- Mappings must be configured for the following:
 - QEI Condition Codes → MultiSpeak Quality Qualifiers
 - QEI Engineering Units → MultiSpeak Units Of Measure

- QEI Status Strings (Suffix Codes) → MultiSpeak Status Identifiers

Specific Vendor Assertions:

- 1) **DisSPatch receives breaker outage reports from TDMS Plus SCADA and automatically creates verified outage/restored records.**

Importance to user: Outage/restored records are generated in real-time, with a high degree of certainty.

How Achieved: When the TDMS Plus SCADA system detects a change in a subscribed status point, it calls the SCADAStatusChangedNotification method on the Milsoft Web Server to publish the change to DisSPatch.

- 2) **DisSPatch receives substation voltage and current reports from TDMS Plus SCADA and automatically creates verified outage/restored records.**

Importance to user: Outage/restored records are generated in real-time, with a high degree of certainty.

How Achieved: When the TDMS Plus SCADA system detects a change in a subscribed analog point, it calls the SCADAAnalogChangedNotification method on the Milsoft Web Server to publish the change to DisSPatch.

- 3) **DisSPatch requests a list of SCADA points from TDMS Plus SCADA.**

Importance to user: SCADA points are mapped to DisSPatch circuit elements, thereby associating SCADA points with circuit model devices.

How Achieved: The DisSPatch user chooses “Get Real-time SCADA Points” from the SCADA menu. DisSPatch then calls the GetAllSCADAPoints method on the TDMS Plus server. The resulting list of SCADA points is displayed to the DisSPatch user in a list. The SCADA point IDs in this list can be dragged into the empty fields on the SCADA page of the Circuit Element Editor for the appropriate DisSPatch circuit elements. This process creates the mapping between SCADA point IDs and DisSPatch circuit elements.

- 4) **DisSPatch requests SCADA analog values from TDMS Plus SCADA.**

Importance to user: SCADA analog values, such as fault currents and feeder currents, can be used by the dispatcher to locate a fault.

How Achieved: After SCADA point IDs have been assigned to their appropriate DisSPatch elements, the DisSPatch user can click the “Read SCADA Analogs” or the “Read SCADA Fault Currents” button on the SCADA page of the Circuit Element Editor of the device they wish to read. DisSPatch then calls the GetSCADAAnalogBySCADAPointId method on the TDMS Plus SCADA server. The returned SCADA analog values are displayed in the Circuit Element Editor. If fault current analog data is available from SCADA, then a fault current analysis study can be performed using these values in order to locate the suspected fault.

- 5) **DisSPatch requests SCADA status values from TDMS Plus SCADA.**

Importance to user: The state of a SCADA status point is used to determine the state of a circuit model device.

How Achieved: Once SCADA point IDs have been assigned to their appropriate DisSPatch circuit elements, the DisSPatch user can click the "Read SCADA Status" button on the SCADA page of the Circuit Element Editor of the device they wish to read. DisSPatch then calls the GetSCADAStatusBySCADAPointId method on the TDMS SCADA server. The returned SCADA status(es) (possibly more than one if multiple phases are desired) are displayed in the Circuit Element Editor.

Summary of Interoperability Test Results Interface #9 SCADA→OA

**Table 1
Recommended MultiSpeak Methods**

Method Name	Importance to User	Supported by Server ¹ (SCADA)	Supported by Client ² (OA)	Verified Inter-operable ³
GetMethods	Requests a list of methods supported by the server.	X	X	X
PingURL	Verifies that the server is running and reachable.	X	X	X
GetAllSCADAPoints	Returns a list of SCADA Point definitions.	X	X	X
GetSCADAAnalogBySCADAPointID	Returns a specific SCADA Analog by SCADAPointID.	X	X	X
GetSCADAStatusBySCADAPointID	Returns a specific SCADA Status by SCADAPointID.	X	X	X

**Table 2
Optional MultiSpeak Methods**

Method Name	Importance to User	Supported by Server ¹ (SCADA)	Supported by Client ² (OA)	Verified Inter-operable ³
GetAllSCADAAnalog	Returns all SCADA Analogs.	X		
GetAllSCADAStatus	Returns all SCADA Status data.	X		
GetDomainMembers	Returns a list of members of a specific domain supported by the server.			
GetDomainNames	Returns a list of domain names supported by the server.			
GetModifiedSCADAPoints	Returns a list of SCADA Point definitions that have changed since the session identified by sessionID.			
OutageEventChangedNotification	OA Notifies SCADA of a change in OutageEvent by sending an array of changed OutageEvent objects.			

- 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.

**Summary of Interoperability Test Results
Interface #9 OA→SCADA**

**Table 3
Recommended MultiSpeak Methods**

Method Name	Importance to User	Supported by Server ¹ (OA)	Supported by Client ² (SCADA)	Verified Inter-operable ³
GetMethods	Requests a list of methods supported by the server.	X	X	X
PingURL	Verifies that the server is running and reachable.	X	X	X
SCADAPointChangedNotification	SCADA Notifies OA of changes in SCADA point definitions by sending an array of changed scadaPoint objects.			
SCADAStatusChangedNotification	SCADA Notifies OA of changes in point status by sending an array of changed scadaStatus objects.	X	X	X

**Table 4
Optional MultiSpeak Methods**

Method Name	Importance to User	Supported by Server ¹ (OA)	Supported by Client ² (SCADA)	Verified Inter-operable ³
GetDomainMembers	Returns a list of members of a specific domain supported by the server.			
GetDomainNames	Returns a list of domain names supported by the server.			
GetActiveOutages	Returns the outageEventIDs for all active outage events.	X		
GetOutageEventStatus	Returns the current status of an outage event, given the outage event ID.	X		
GetOutageEventStatusByOutageLocation	Returns the current status of an outage event, given the outage location.	X		
SCADAAnalogChangedNotification	SCADA Notifies OA of changes in analog values by sending an array of changed scadaAnalog objects.	X	X	X
SCADAAnalogChangedNotificationByPointID	SCADA Notifies OA of changes in a specific analog value, chosen by scadaPointID, by sending a changed scadaAnalog object.			
SCADAAnalogChangedNotificationForPower	SCADA Notifies OA of changes in power analogs, by sending an array of changed scadaAnalog objects.			
SCADAAnalogChangedNotificationForVoltage	SCADA Notifies OA of changed voltage analogs, by sending an array of changed scadaAnalog objects.			

SCADAPointChangedNotificationForAnalog	SCADA Notifies OA of changes in SCADA point definitions, limited to Analog points, by sending an array of changed scadaPoint objects.			
SCADAPointChangedNotificationForStatus	SCADA Notifies OA of changes in SCADA point definitions, limited to Status points, by sending an array of changed scadaPoint objects.			
SCADAStatusChangedNotificationByPointID	SCADA Notifies OA of changes in the status of a specific point, chosen by PointID, by sending a changed scadaStatus object.			

- 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 QEI, Inc.:



Senior Software Engineer

Name: Mark Boxall

Date: 12/15/2006

For Milsoft Utility Solutions, Inc.:



Executive Vice President

Name: Luis Malave

Date: 12/15/2006

Assertions Verified by:



Name: Hannu Huhdanpaa

UISOL, Inc.

Testing Agent

Date: 12/15/2006

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, 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, Cornice Engineering, Inc., or the Testing Agent.

December 5, 2006