ELECTRIC POWER
RESEARCH INSTITUTE

Enterprise Integration Functions Test Plan for Distributed Energy Resources, Phase 1

3002004681

Enterprise Integration Functions Test Plan for Distributed Energy Resources, Phase 1

3002004681

Technical Update, November 2014

EPRI Project Manager

B. Seal

All or a portion of the requirements of the EPRI Nuclear Quality Assurance Program apply to this product.

YES [✓ NO]

ELECTRIC POWER RESEARCH INSTITUTE
3420 Hillview Avenue, Palo Alto, California 94304-1338 ▪ PO Box 10412, Palo Alto, California 94303-0813 ▪ USA
800.313.3774 ▪ 650.855.2121 ▪ askepri@epri.com ▪ www.epri.com

DISCLAIMER OF WARRANTIES AND LIMITATION OF LIABILITIES

THIS DOCUMENT WAS PREPARED BY THE ORGANIZATION(S) NAMED BELOW AS AN ACCOUNT OF WORK SPONSORED OR COSPONSORED BY THE ELECTRIC POWER RESEARCH INSTITUTE, INC. (EPRI). NEITHER EPRI, ANY MEMBER OF EPRI, ANY COSPONSOR, THE ORGANIZATION(S) BELOW, NOR ANY PERSON ACTING ON BEHALF OF ANY OF THEM:

(A) MAKES ANY WARRANTY OR REPRESENTATION WHATSOEVER, EXPRESS OR IMPLIED, (I) WITH RESPECT TO THE USE OF ANY INFORMATION, APPARATUS, METHOD, PROCESS, OR SIMILAR ITEM DISCLOSED IN THIS DOCUMENT, INCLUDING MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, OR (II) THAT SUCH USE DOES NOT INFRINGE ON OR INTERFERE WITH PRIVATELY OWNED RIGHTS, INCLUDING ANY PARTY'S INTELLECTUAL PROPERTY, OR (III) THAT THIS DOCUMENT IS SUITABLE TO ANY PARTICULAR USER'S CIRCUMSTANCE; OR

(B) ASSUMES RESPONSIBILITY FOR ANY DAMAGES OR OTHER LIABILITY WHATSOEVER (INCLUDING ANY CONSEQUENTIAL DAMAGES, EVEN IF EPRI OR ANY EPRI REPRESENTATIVE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES) RESULTING FROM YOUR SELECTION OR USE OF THIS DOCUMENT OR ANY INFORMATION, APPARATUS, METHOD, PROCESS, OR SIMILAR ITEM DISCLOSED IN THIS DOCUMENT.

REFERENCE HEREIN TO ANY SPECIFIC COMMERCIAL PRODUCT, PROCESS, OR SERVICE BY ITS TRADE NAME, TRADEMARK, MANUFACTURER, OR OTHERWISE, DOES NOT NECESSARILY CONSTITUTE OR IMPLY ITS ENDORSEMENT, RECOMMENDATION, OR FAVORING BY EPRI.

THE ELECTRIC POWER RESEARCH INSTITUTE (EPRI) PREPARED THIS REPORT.

THE TECHNICAL CONTENTS OF THIS PRODUCT WERE **NOT** PREPARED IN ACCORDANCE WITH THE EPRI QUALITY PROGRAM MANUAL THAT FULFILLS THE REQUIREMENTS OF 10 CFR 50, APPENDIX B. THIS PRODUCT IS **NOT** SUBJECT TO THE REQUIREMENTS OF 10 CFR PART 21.

This is an EPRI Technical Update report. A Technical Update report is intended as an informal report of continuing research, a meeting, or a topical study. It is not a final EPRI technical report.

NOTE

For further information about EPRI, call the EPRI Customer Assistance Center at 800.313.3774 or e-mail askepri@epri.com.

ACKNOWLEDGMENTS

The Electric Power Research Institute (EPRI) prepared this report.

Principal Investigators
B. Seal
J. Simmins
G. Gray

This report describes research sponsored by EPRI.

EPRI would like to acknowledge the contributions of Boreas Group in the update to the semantic test harness to support these additional message types.

This publication is a corporate document that should be cited in the literature in the following manner:

Enterprise Integration Functions Test Plan for Distributed Energy Resources, Phase 1. EPRI, Palo Alto, CA: 2014. 3002004681.

ABSTRACT

Since 2012, the Electric Power Research Institute (EPRI) has facilitated a focus group of industry experts working to develop common functions for enterprise integration of distributed energy resources and to coordinate these functions with the relevant standards organizations. The activity is aimed at advancing industry efforts to bring inverter-connected distributed energy storage and generation into use as a grid resource. This report presents the results of the first phase of developments by this work group, addressing DER management in aggregate groups, including the discovery of capabilities, status monitoring, and dispatch of real and reactive power. These messages are intended for use in application-to-application or business-to-business interfaces through which DER are being monitored and managed in aggregate. It is intended to provide the industry with a point of reference for DER integration with other utility applications and to provide guidance to research and standards development organizations.

The electric power industry has taken previous steps to prepare for higher penetration of distributed energy resources (DER) by working collaboratively to develop smart inverter standards and field network protocols that may be used for monitoring and managing devices in the field. But standards do not yet exist to support the enterprise integration (software-to-software) of these device capabilities in a useful and manageable way.

The ongoing focus group activity is being coordinated with the Department of Energy (DOE), through the DOE SunShot SEGIS-AC (Solar Energy Grid Integration Systems – Advanced Concepts) program, and with the National Institute of Standards and Technology, through the Smart Grid Interoperability Panel's Distributed Renewables, Generators, and Storage Domain Expert Working Group. The initiative began with a face-to-face workshop held in Washington, DC in September, 2012 and continued through a series of teleconferences. These meetings brought together a group of utility distribution management experts, distribution management system (DMS) software, and distributed energy storage and generation specialists to identify a starting list (core set) of practical, enterprise-level interactions for DER.

This report is of highly technical content. It is the specific test plan that was prepared and used in a face-to-face workshop held at the National Renewable Energy Laboratory (NREL) Energy Systems Integration Facility (ESIF) in Golden, Colorado October 28th and 29th, 2014. This workshop was designed for software companies to come together and test actual implementations of these enterprise integration functions. The findings and outcome of this workshop are published separately.

Keywords

Distribution Management System
Distributed Energy Resources
DERMS
Enterprise Integration
Photovoltaics
Battery Storage

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1
INTRODUCTION

Technology advancements in solar photovoltaics (PV), energy storage and other distributed energy resources have driven increases in their evaluation and employment by utilities, consumers, and third parties. These Distributed Energy Storage and Generation Resources (DER) are often connected to the grid at the distribution level where their presence in large scale or volume could be disruptive if not designed, integrated, and managed properly. There is also a growing trend to understand how these resources might be leveraged as part of a microgrid.

Inverters, the power converter circuits that integrate solar PV and battery resources to the grid, are highly-capable devices with fast power controls and no inherent inertia such that they can respond quickly to commands and local conditions. Even small scale inverters tend to have substantial processing and memory resources and are capable of supporting a variety of communication protocols and advanced functions. Over the last few years, industry efforts have defined a wide range of standard grid-supportive functions that inverters may provide and standard communication protocols that allow these functions to be remotely monitored and managed.[1] Other types of DER are also becoming increasingly intelligent and include communication interfaces that make them able to participate as active elements of the power system.

If these distributed resource capabilities can be properly exposed and integrated with traditional utility system operations, high penetration DER can be transformed from problematic uncertainties to beneficial tools for grid and microgrid stabilization and distribution management. To achieve these potential benefits, it must be possible not only to communicate to individual DER devices using standard protocols, but also for the systems that manage DER, referred to herein as DER Management System or "DERMS," to effectively inform other software applications regarding the resources available and to exchange information that allows the DER to be managed effectively. DERMS might exist in many forms, being utility applications, inverter manufacturer head-end systems, or third-party aggregator systems and may manage at the local (microgrid), feeder, substation, or wide-area level.

Traditionally, distribution systems have been operated without extensive controls or centralized management. More advanced systems may have On-Load Tap Changing transformers (LTC's) at substations, line regulators, and/or capacitor banks that operate to help optimize distribution voltage and reactive power flow. In many cases, these devices may be fixed or configured to operate autonomously. In a growing number of cases, however, a more central Distribution Management System (DMS) has been used to coordinate their behavior for a more optimized overall effect. DMS functionality may reside at the utility operations center where a single large-scale software solution manages many circuits, or it may reside in a more limited fashion at the substation or other level, where smaller-scale systems act to manage individual feeders or circuits.

[1] *Common Functions for Smart Inverters.* EPRI, Palo Alto, CA: 2011. 1023059

Regardless of the particular scenario, the present generation of DMS systems is still evolving to take advantage of the capabilities that DER may offer. In most cases, DER support within a DMS is limited to monitoring the output of "utility scale" DERs (> one megawatt). In addition, existing industry standards define advanced functions for DER only at the individual device level, and lack the more aggregated, feeder-level representations that are useful for enterprise integration.

Since 2012, the Electric Power Research Institute (EPRI) has been working in coordination with the DOE SunShot program and the National Institute of Standards and Technology (NIST) Distributed Renewables, Generation, and Storage-Domain Expert Working Group (DRGS-DEWG) to facilitate a focus group of industry experts working to develop appropriate enterprise-level functions for the integration of distributed energy resources. These functions are intended to work in conjunction with the common functions for smart inverters that have previously been defined.

This report is the specific test plan that was prepared and used in a face-to-face workshop held at the National Renewable Energy Laboratory (NREL) Energy Systems Integration Facility (ESIF) in Golden, Colorado October 28 and 29, 2014. This workshop was designed for software companies to come together and test actual implementations of these enterprise integration functions. The findings and oucome of this workshop are published separately.

2
USE CASE DESCRIPTIONS

During the public workshop held in Washington, DC in 2012,[2] five high value use cases were identified as being the highest priority for development:

- DER Status Monitoring
- DER Capabilities Discovery
- Real Power Dispatch
- Reactive Power Dispatch
- DER Forecast

These five needs were developed into the specific interactions identified in Table 2-1 by the working group.

Table 2-1
Interactions to Support the Five Priority Topics

Enterprise Function	Description
Individual DER Representation in the System Model	This enables general representation in the system model and sharing of individual DER existence and capabilities among software applications in the enterprise environment.
DER Group Creation	This function allows a software entity to define a logical group of DER and to exchange the definition of this group (farm) with other applications. The purpose of grouping is subsequent monitoring and management at the group (farm) level.
DER Group Maintenance	The ability to add, update, or delete group members
Status Monitoring of DER Groups	This function allows the exchange of real-time status information for DER groups. This is dynamic data, including present set/generation value and present max/min dispatchable ranges.
Nameplate Capability of DER Groups	This function allows the exchange of information indicative of the installed (nameplate) capability of DER groups. This data is generally static, changing only as equipment changes are made.
Real Power Dispatch of DER Groups	This function allows the dispatch of real power from DER groups.
Reactive Power Dispatch of DER Groups	This function allows the dispatch of reactive power from DER groups.
DER Group Forecast	This function allows the exchange of forecast information regarding the availability of real and reactive power from DER groups.

[2] The findings from this workshop are documented in the EPRI report Enterprise Integration Functions for Distributed Energy Resources, Phase 1, Product ID: 3002001249, Technical Update, October 2013.

Setting the Test Plan Context

This test plan leverages the information contained in the EPRI Report, Enterprise Integration Functions for Distributed Energy Resources, Phase 1, Product ID: 3002001249, Technical Update, October 2013.

The elements shown in green at the bottom of Figure 2-1 represent devices (potentially numerous) in the field and communication systems that connect these devices (Point 1). The elements shown in blue represent the IT environment where a variety of companies and/or software applications may exchange information (Point 2).

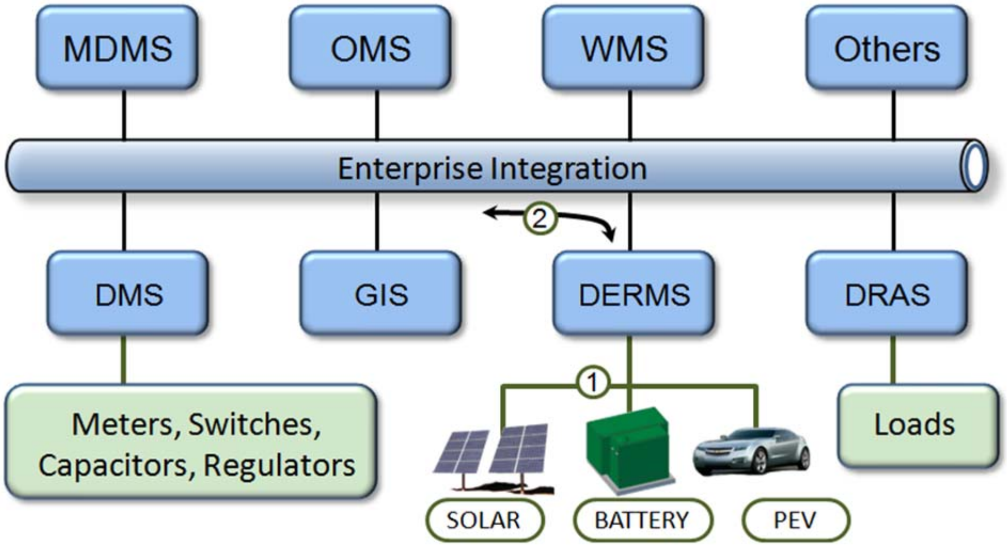


Figure 2-1
High Level Integration Architecture, Showing Field and Enterprise Interfaces

Regardless of where the DER management function is ultimately deployed (a dedicated DERMS or other) it will be necessary for the DER management function to share information with other relevant systems at the utility. This enterprise integration definition is fairly mature with IEC 61968-100 published in 2013 and MultiSpeak in the midst of an upgrade to version 5, so the challenge is not with the “how” to do enterprise integration *per se*. What will be needed are extensions to existing standards to support the new DER functions.

In terms of communication from the enterprise to devices in the distribution network, (Point 1 in Figure 2-1) there is an expectation that the DERMS will employ either a proprietary vendor interface, or interfaces based on standards such as the IEC 61850 and Distributed Network Protocol (DNP3).

What happens inside the DERMS will be up to each vendor and how the DER are managed will also be up to a given vendor. It is neither beneficial nor intended by this initiative to standardize the inner workings and DER optimization algorithms that might be employed inside a DERMS. The focus of this work is to enable communication from the DERMS to the enterprise using utility integration standards, CIM and MultiSpeak.

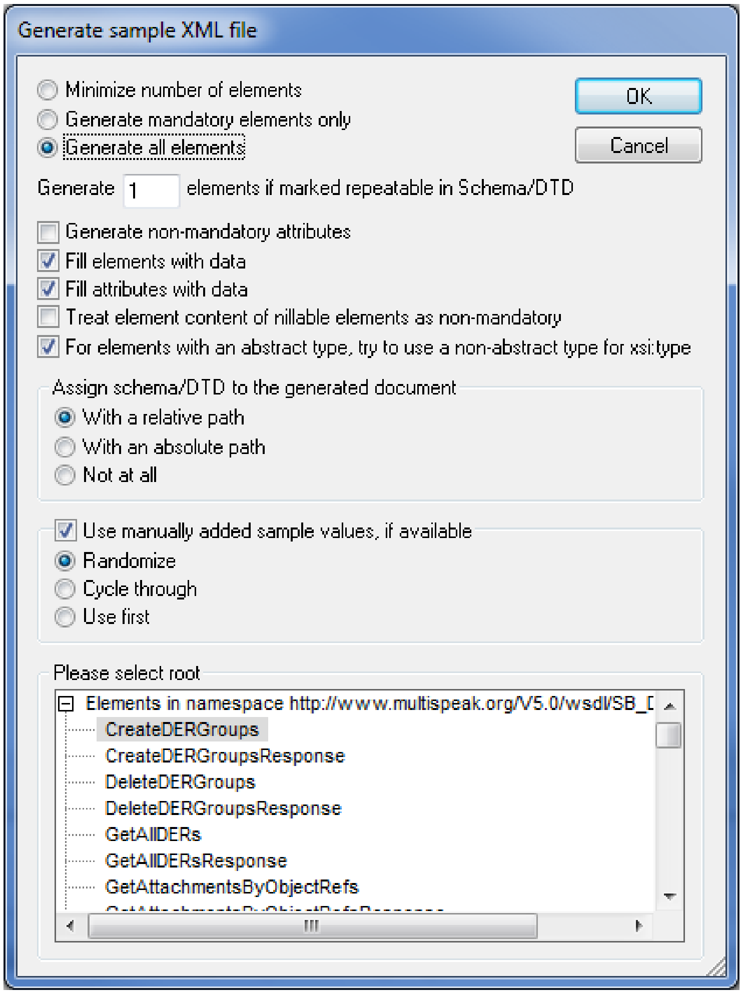
Use Cases

This interest group has recognized that a wide range of potential use cases have been developed for grid-integration of DER. Several primary sources for these use cases have been identified and documented separately. It has also been recognized that DER integration with the grid is an emerging field, that the value of presently identified use cases may be uncertain, and that many high-value use cases may be yet unidentified. The intention of this work is to take a useful first step, maintaining flexibility, with the intention of modifying and adding new services going forward as more understanding is gained by the industry.

3 TEST PLAN FOR ENTERPRISE FUNCTIONS OF DER

Generating example XML

These examples were generated using XMLSpy. They were creating by selecting the XSDs provided in *Enterprise Integration Functions for Distributed Energy Resources, Phase 1*. EPRI, Palo Alto, CA: 2013. 3002001249. Under the **DTD/Schema** option from the main menu, the **Generate Sample XML file** option shown in Figure 3-1 and Figure 3-2 was selected with the following attributes:



Generate sample XML file

○ Minimize number of elements
○ Generate mandatory elements only
● Generate all elements

Generate 1 elements if marked repeatable in Schema/DTD

□ Generate non-mandatory attributes
☑ Fill elements with data
☑ Fill attributes with data
□ Treat element content of nillable elements as non-mandatory
☑ For elements with an abstract type, try to use a non-abstract type for xsi:type

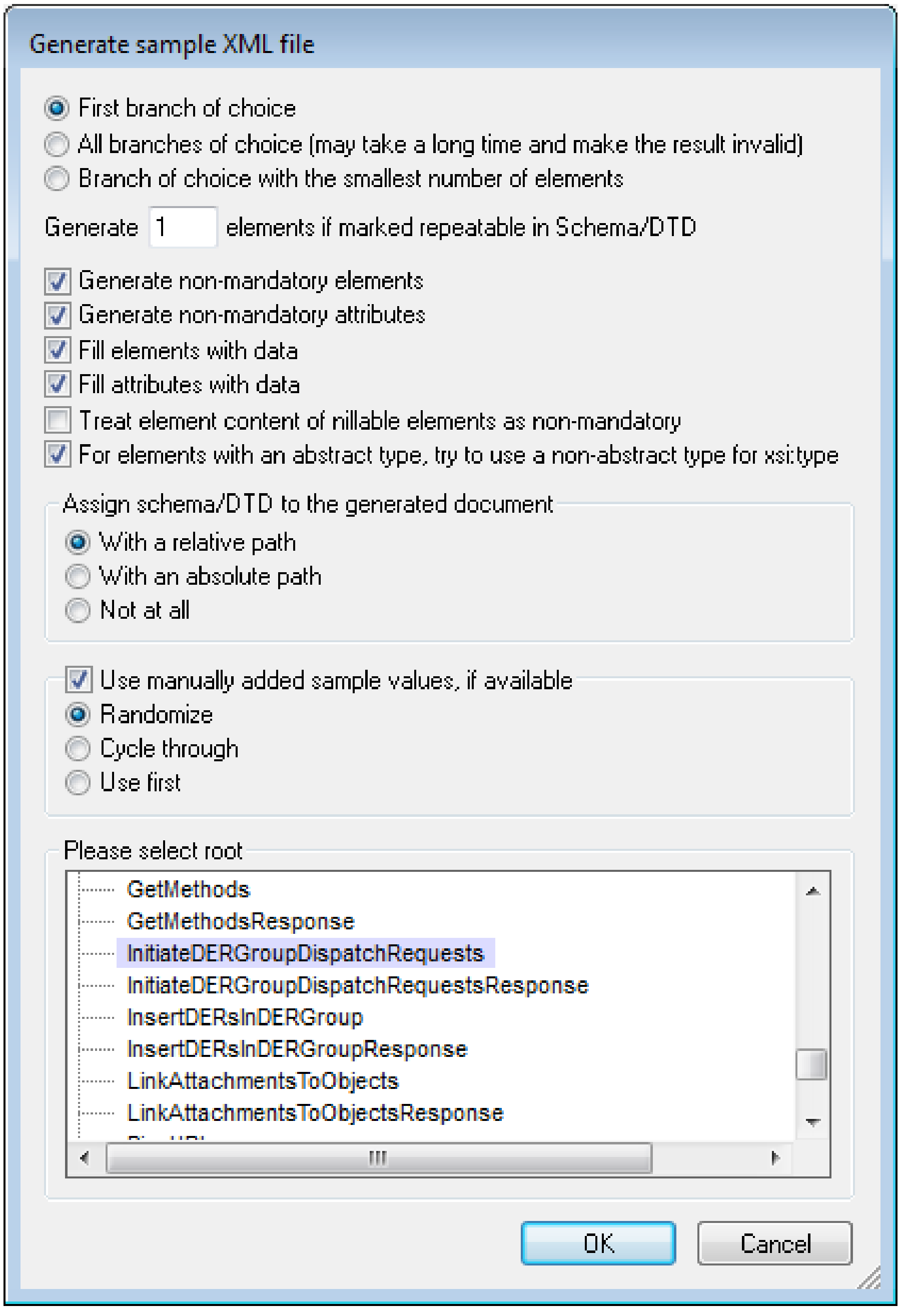
Assign schema/DTD to the generated document
● With a relative path
○ With an absolute path
○ Not at all

☑ Use manually added sample values, if available
● Randomize
○ Cycle through
○ Use first

Please select root

Elements in namespace http://www.multispeak.org/V5.0/wsdl/SB_D
- CreateDERGroups
- CreateDERGroupsResponse
- DeleteDERGroups
- DeleteDERGroupsResponse
- GetAllDERs
- GetAllDERsResponse
- GetAttachmentsByObjectRefs

Figure 3-1
Generating example XML using XMLSpy – v2012 options



Generate sample XML file

- ● First branch of choice
- ○ All branches of choice (may take a long time and make the result invalid)
- ○ Branch of choice with the smallest number of elements

Generate 1 elements if marked repeatable in Schema/DTD

- ☑ Generate non-mandatory elements
- ☑ Generate non-mandatory attributes
- ☑ Fill elements with data
- ☑ Fill attributes with data
- □ Treat element content of nillable elements as non-mandatory
- ☑ For elements with an abstract type, try to use a non-abstract type for xsi:type

Assign schema/DTD to the generated document

- ● With a relative path
- ○ With an absolute path
- ○ Not at all

☑ Use manually added sample values, if available

- ● Randomize
- ○ Cycle through
- ○ Use first

Please select root

- GetMethods
- GetMethodsResponse
- InitiateDERGroupDispatchRequests
- InitiateDERGroupDispatchRequestsResponse
- InsertDERsInDERGroup
- InsertDERsInDERGroupResponse
- LinkAttachmentsToObjects
- LinkAttachmentsToObjectsResponse

Figure 3-2
Generating example XML using XMLSpy – v2014 options

Use of Example XML

The example XML typically represents an example of the minimum XML required to satisfy a message requirement. These examples represent optimistic, positive tests. That is, no expectation is made for error-handling in this initial version of the test plan. Additionally, while both the CIM-based and MultiSpeak XSDs support extensions, this capability is also not being tested so extensions are not included.

Finally, it should be noted that these examples are for the messages *payloads* only. For WSDL examples, please see the associated appendix.

Sequence diagram conventions

For the purpose of this test plan, there is no "system of record" for the DERMS or other systems that might communicate with the DERMS; hence, generic actors such as "requesting entity" and "providing entity" will be shown in the various sequence diagrams. The idea is that a DERMS might be a standalone product, or perhaps subsumed into an existing DMS. The idea is that the specification is not prescriptive as to where the DERMS functionality resides, but only to test the communications that will need to occur.

One caveat for the purposes of MultiSpeak: Specific examples have been used as potential guidance that will be reflected in the new version 5.0.x specification.

A note on the use of MultiSpeak primary/secondary identifier

> While the objectID within MultiSpeak is the "linking" and "key" attribute for objects and how they are tracked, MultiSpeak also makes use of primary identifier and secondary identifier attributes which are human readable and may provide additional information about a given object. From the MultiSpeak XSD comments:
>
> Primary Identifier:
>
> Primary human-readable identifier for this instance of the object. For instance, if this object is an instance of an electricMeter, this should be the meterNumber. If this is a customer account, this should be the accountNumber. If this identifier is not applicable to a specific data instance then fill this field with "NA."
>
> Secondary Identifier:
>
> Additional human-readable identifier for this instance of the object. For instance, for a workOrder, the secondaryIdentifier might be the jobNumber.
>
> The primary identifier is required, but for these tests will be set to "NA."
>
> The secondary identifier is optional and for these tests will not be used.

Test 1 - DER Group Creation

The enterprise integration interest group found that a process for identifying the intended set of DER was a necessary precursor to status monitoring, capabilities discovery, and dispatch functions. This is necessary so that it becomes possible to monitor and manage DER at a higher level with a focus on the attributes, impacts, and opportunities as they relate to the distribution system rather than individual DER plants or devices. A local control system may manage units that belong to a group; for example, units that belong to a wind or PV farm. But again, the focus of this effort is a grouping of systems for communication to the back-office; how the communications to the resources in the field are managed will be up to a given vendor. The basic idea is to precede DER monitoring or management messages with a process to define a grouping of DER. This will make it possible for any application to define groupings of DER according to whatever rationale is of interest to that application. The grouping could be by type (wind, PV), geographic location, or other distribution assets such as a feeder or substation. It also makes it possible for the entity providing the DER service (e.g. a DERMS) to not be required to read-in and process the real-time connectivity model. Figure 3-3 shows a group creation sequence diagram.

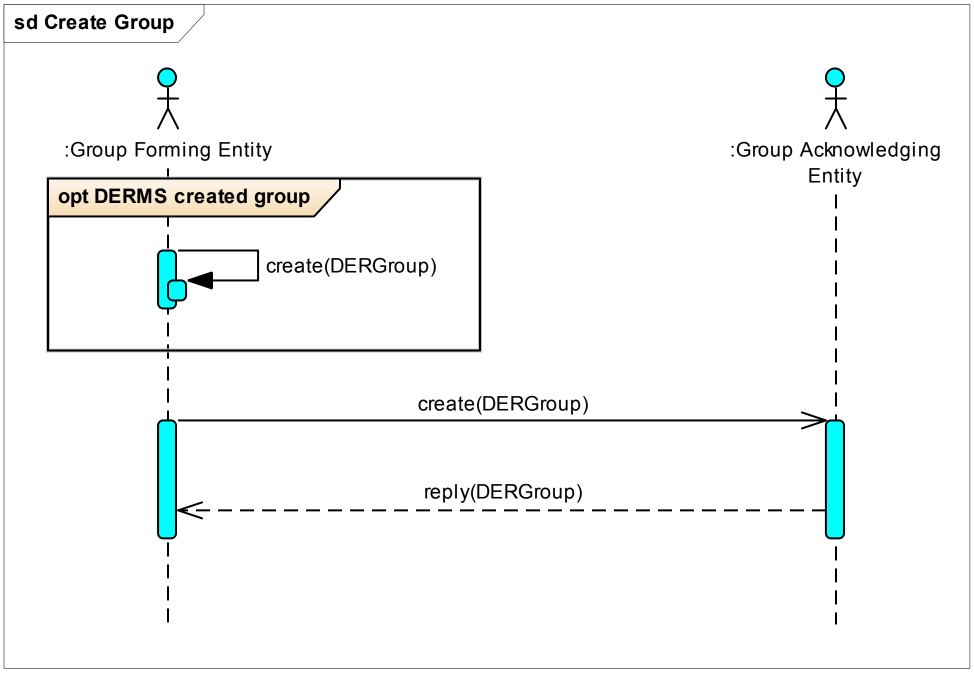


Figure 3-3
Group Creation Sequence Diagram

Note: Many of these example sequence diagrams use nomenclature from IEC 61968-100 for the naming conventions and verb use, e.g. CREATE, GET, CHANGED, etc. MultiSpeak has similar naming conventions for services that utilize verbs such as “Initiate,” “Notification,” or “Get.” The latest documentation for MultiSpeak services is available from the Multispeak website.[3]

The approach involves an interaction in which a DERGroup is defined by one entity (e.g. the Group Forming entity such as a DMS) and provided to one or more Group Acknowledging entities (e.g. one or more DERMS). This interaction could occur immediately before another transaction, such as a DER Status request and reply, or any time prior. The diagram also shows that optionally, the DER Group creation function may not be a request/reply to another application, but might be created within a single application. Figure 3-4 illustrates MultiSpeak DER Group creation.

[3] www.multispeak.org

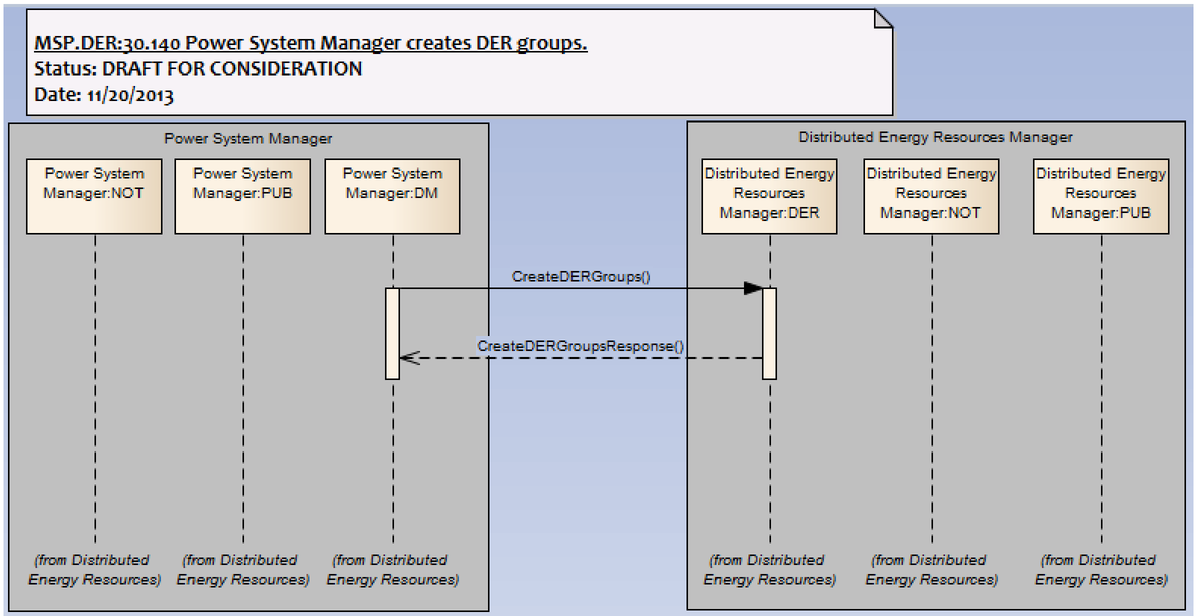

MSP.DER:30.140 Power System Manager creates DER groups.
Status: DRAFT FOR CONSIDERATION
Date: 11/20/2013
Power System Manager
Power System Manager:NOT
Power System Manager:PUB
Power System Manager:DM
Distributed Energy Resources Manager
Distributed Energy Resources Manager:DER
Distributed Energy Resources Manager:NOT
Distributed Energy Resources Manager:PUB
CreateDERGroups()
CreateDERGroupsResponse()
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)

Figure 3-4
Example of MultiSpeak DER Group creation

This interaction would provide a unique group name and a list of the DER that are included in the group. The DER would be identified by their unique names or unique terminals, as appropriate for the enterprise data integration model being utilized (e.g. MultiSpeak, CIM). One DER could be a member of any number of groups.

In this way, groups could be defined that relate to any level of aggregation desired, including by substation, feeder, line segment, or other. This approach is not prescriptive of a particular integration approach, allowing, for example:

- A DERMS could process the system model and define its own groupings, but does not require it to do so. (Shown in the "Opt" portion of the sequence diagram.)
- A DMS could define the specific groups that are of interest to its processes. This could include different groupings that would be associated with various power system configurations. For example, in the circuit arrangement illustrated in Figure 3-5 a DMS could define one group for all the DER connected to Substation 1 (along the blue line) and another for all the DER connected to Substation 2 (along the red line). If the open switch along this line is then closed and a different switch opened, then two different groups could be defined to represent that alternative circuit configuration.

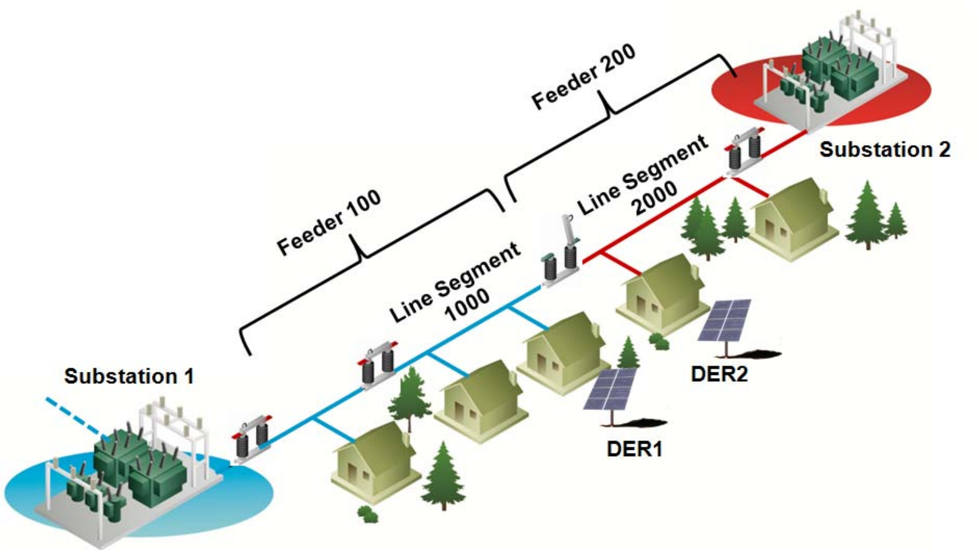


Figure 3-5
Example of Feeder with Alternate Substation

Pre-Condition(s) None

Post Condition(s): A new group is created with the name that was passed and with a unique id. The group will contain at least 1 DER with corresponding data.

Table 3-1, Table 3-2, and Table 3-3 show MultiSpeak and CIM Manual Group Creation.

Table 3-1 Test: MultiSpeak Manual Group Creation

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	Manual Group Creation	http://54.210.233.171/	Create	DERGroups	
2	Manual Group Creation	http://54.210.233.171/	Create	DERGroupsResponse	

Required data:

Data Element	Data Type	Notes
DERGroup.objectID	UUID	Identifies the DER Group
DERMember.objectID	UUID	Identifies the DERMember
DERGroup.primaryIdentifier	String	Set to “NA”

Table 3-2
Test: CIM Manual Group Creation

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	Manual Group Creation	http://54.210.233.171/	Create	DERGroup	
2	Manual Group Creation	http://54.210.233.171/	Reply	DERGroup	

Required data:

Data Element	Data Type	Notes
DERGroup.mRID	UUID	Identifies the DER Group
DERMember.mRID	UUID	Identifies the DERMember

Table 3-3
MultiSpeak Manual Group Creation

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	Manual Group Creation	http://54.210.233.171/	create	DERGroups	
2	Manual Group Creation	http://54.210.233.171/	create	DERGroupsResponse	

Example MultiSpeak-based XML- createDERGroup

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2012 rel. 2 (x64)
(http://www.altova.com)-->
<CreateDERGroups
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB DER Server
SB DER Server.xsd" xmlns="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server"
xmlns:msp="http://www.multispeak.org/V5.0"
xmlns:com="http://www.multispeak.org/V5.0/commonTypes"
xmlns:cpsm="http://www.multispeak.org/V5.0/cpsm"
xmlns:sbarrays="http://www.multispeak.org/V5.0/sandboxArrays"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <ArrayOfDERGroup>
            <sbarrays:DERGroup objectGUID="f975be36-a3b6-499c-9518-
5e74555b6db9">
                  <msp:comments>Your comment here</msp:comments>
                  <msp:primaryIdentifier identifierName="NA"/>
                  <DERIDs xmlns="">
                        <DERID objectGUID="c1357587-0a37-41a1-bdb6-
65af875972a4">Example DER member: PV array - 1</DERID>
                  </DERIDs>
            </sbarrays:DERGroup>
      </ArrayOfDERGroup>
      <responseURL>http://www.epri.com/blahblahblah</responseURL>
      <transactionID>Test1 DERGroup Creation - Attempt ##</transactionID>
</CreateDERGroups>
```

Example MultiSpeak-based XML- createDERGroupResponse

Note that in this response example that the objectID for the DERGroup and for the DERMember that were created are the same as in the request. In this way the requesting system can verify internally that the group/member were added.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2012 rel. 2 (x64)
(http://www.altova.com)-->
<createDERGroups
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server
SB_DER_Server.xsd" xmlns="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server"
xmlns:msp="http://www.multispeak.org/V5.0"
xmlns:com="http://www.multispeak.org/V5.0/commonTypes"
xmlns:cpsm="http://www.multispeak.org/V5.0/cpsm"
xmlns:sbarrays="http://www.multispeak.org/V5.0/sandboxArrays"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <ArrayOfDERGroup>
            <sbarrays:DERGroup objectGUID="f975be36-a3b6-499c-9518-
5e74555b6db9">
                  <msp:comments>Your comment here</msp:comments>
                  <msp:primaryIdentifier
identifierName="String">NA</msp:primaryIdentifier>
                  <DERIDs xmlns="">
                        <DERID objectGUID="c1357587-0a37-41a1-bdb6-
65af875972a4">Example DER member: PV array - 1</DERID>
                  </DERIDs>
            </sbarrays:DERGroup>
      </ArrayOfDERGroup>
      <responseURL>http://www.epri.com/blahblahblah</responseURL>
      <transactionID> Test1 DERGroup Creation response - Attempt
##</transactionID>
</createDERGroups>
```

Example CIM-based XML- createDERGroup

This example XML is for a single group called “Example Name” that contains a single member called “Example DER member: PV array – 1”

```
<?xml version="1.0" encoding="UTF-8"?>
<m:DERGroups xmlns:m="http://www.epri.com/2013/DERGroup#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.epri.com/2013/DERGroup#
file:///C:/Users/PGGR001/Documents/EPRI/067771%20-
%20DER/ExampleProfiles/createDERGroup.xsd">
      <m:DERGroup>
            <m:name>Example DERGroup Name</m:name>
            <!-- for MultiSpeak this would be an objectID not an mRID -->
            <m:mRID>f975be36-a3b6-499c-9518-5e74555b6db9</m:mRID>
            <m:DERMember>
                  <!-- The name, names class, and mRID (objectID for
MultiSpeak, is optional, but one of them MUST be used. This example uses a
name and a GUID -->
                  <m:mRID>c1357587-0a37-41a1-bdb6-65af875972a4</m:mRID>
                  <m:name>Example DER member: PV array - 1</m:name>
            </m:DERMember>
      </m:DERGroup>
</m:DERGroups>
```

Example CIM-based XML- ReplyDERGroup

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<DERGroups xmlns="http://www.epri.com/2013/DERGroup#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.epri.com/2013/DERGroup# createDERGroup.xsd">
      <DERGroup description="text" comment="text">
            <name>Example DERGroup name</name>
            <!-- identifier used for the DERGroup, so the requestor can marry
up to the request that was made -->
            <mRID>f975be36-a3b6-499c-9518-5e74555b6db9</mRID>
      </DERGroup>
</DERGroups>
```

Group Maintenance

Once a group is created, any system could also query to get information about group members or initiate an action to update the DER Group membership adding, removing, or updating information about a specific member as illustrated in Figure 3-6. This is accomplished using the same message profile that is used to create the group. This profile has the flexibility to support this DER Group maintenance.

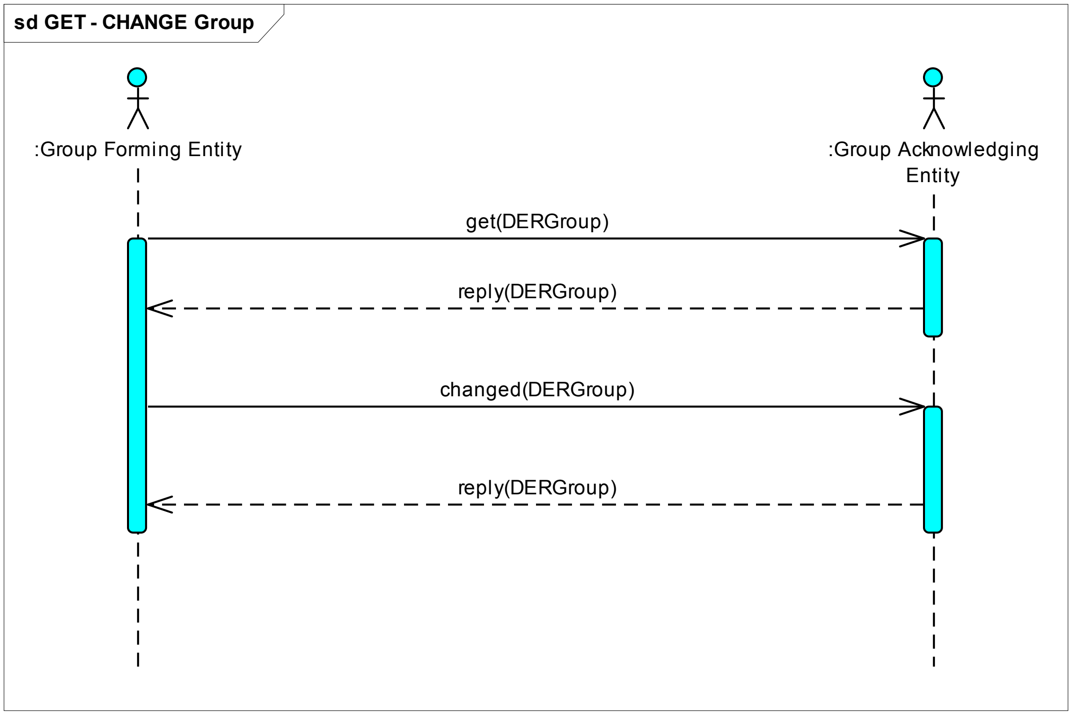


Figure 3-6
Querying for and Maintaining DERGroup Membership

MultiSpeak, while having a similar style as CIM-based integration in terms of naming convention, provides additional methods using the naming to identify alternative ways to query. CIM-based integration typically passes parameters in the message header to provide alternative means of querying. The MultiSpeak style of naming convention identifies the type of query by ID, by name, and in the method name as can be seen in the example sequence diagrams.

Test 2 – Querying a DER group

Once a group is created, any system could also query to get information about group members, or initiate an action to update the DER Group membership, adding, removing, or updating information about a specific member as illustrated in Figure 3-7 and Figure 3-8. This is accomplished using the same message profile that is used to create the group. This profile has the flexibility to support this DER Group maintenance.

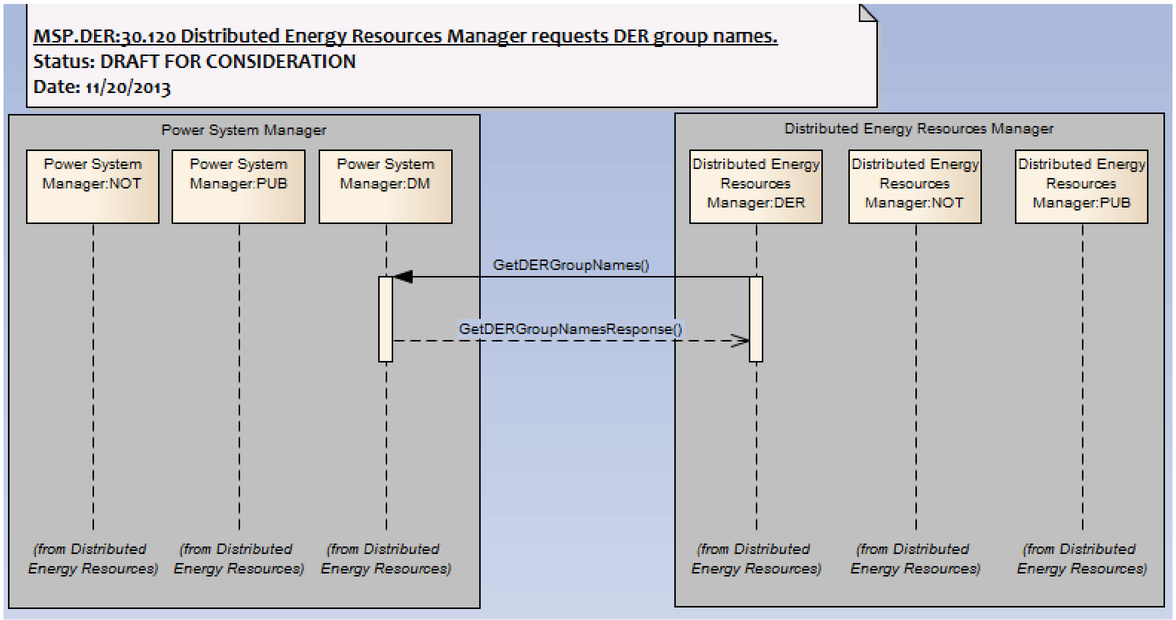

MSP.DER:30.120 Distributed Energy Resources Manager requests DER group names.
Status: DRAFT FOR CONSIDERATION
Date: 11/20/2013
Power System Manager
Power System Manager:NOT
Power System Manager:PUB
Power System Manager:DM
Distributed Energy Resources Manager
Distributed Energy Resources Manager:DER
Distributed Energy Resources Manager:NOT
Distributed Energy Resources Manager:PUB
GetDERGroupNames()
GetDERGroupNamesResponse()
(from Distributed Energy Resources)

Figure 3-7
Example of MultiSpeak query for DER Group names

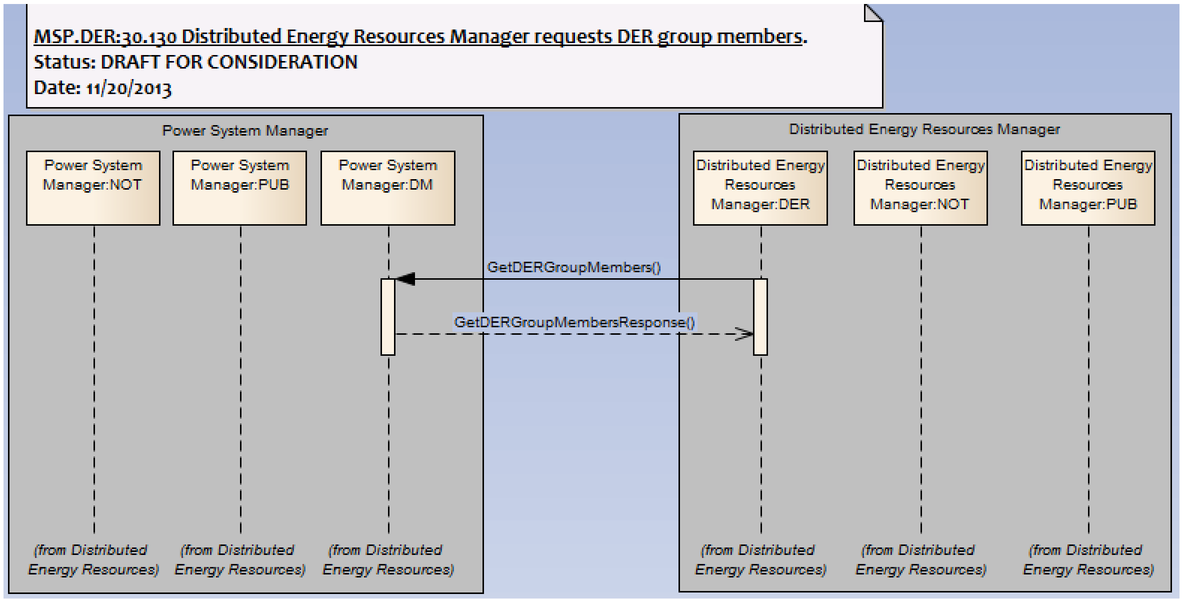

MSP.DER:30.130 Distributed Energy Resources Manager requests DER group members.
Status: DRAFT FOR CONSIDERATION
Date: 11/20/2013
Power System Manager
Power System Manager:NOT
Power System Manager:PUB
Power System Manager:DM
Distributed Energy Resources Manager
Distributed Energy Resources Manager:DER
Distributed Energy Resources Manager:NOT
Distributed Energy Resources Manager:PUB
GetDERGroupMembers()
GetDERGroupMembersResponse()
(from Distributed Energy Resources)

Figure 3-8
Example of MultiSpeak query for DER Group members

Pre-Condition(s) A DER Group exists and has at least one DER in it.

Post Condition(s): The requesting system has received the names and ID's of members of the requested group

Table 3-4 illustrates MultiSpeak DER Group query.

Table 3-4
Test: MultiSpeak DER Group query

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	DERGroup query	http://54.210.233.171/	get	DERsByDERID	
2	DERGroup response	http://54.210.233.171/	get	DERsByDERIDResponse	

Required data:

Data Element	Data Type	Notes
DERGroup.objectID	UUID	Identifies the DER Group for which DER information is requested

Example MultiSpeak-based XML – Get DERsByDERID

This example shows the a request to receive the DER for a specific group

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2015 sp1 (x64)
(http://www.altova.com)-->
<GetDERsByDERID xmlns="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:sbarrays="http://www.multispeak.org/V5.0/sandboxArrays"
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server
SB DER Server.xsd">
      <ArrayOfDERID>
            <sbarrays:DERID DERName=" Example DER Group Name "
objectGUID="8493bde3-afca-411a-8234-d66f550090c3" owner="String"
registeredName="" systemName="">String</sbarrays:DERID>
      </ArrayOfDERID>
</GetDERsByDERID>
```

Example MultiSpeak-based XML – Get DERGroups Response

Using the DERGroup IDs from the above request, this example response returns two members for the first group and three members for the second. This response could have optionally included information about each DER members' capability.

Using the DERGroup ID from the above request this example response returns the DER that is in the group, with some example capabilities.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2015 sp1 (x64)
(http://www.altova.com)-->
<n1:GetDERsByDERIDResponse xmlns="http://www.multispeak.org/V5.0/commonTypes"
xmlns:msp="http://www.multispeak.org/V5.0"
xmlns:gml="http://www.multispeak.org/V5.0/gml"
xmlns:cpsm="http://www.multispeak.org/V5.0/cpsm"
xmlns:xml="http://www.w3.org/XML/1998/namespace"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:sbarrays="http://www.multispeak.org/V5.0/sandboxArrays"
xmlns:n1="http://www.multispeak.org/V5.0/wsdl/SB DER Server"
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server
SB DER Server.xsd">
      <n1:ArrayOfDER>
            <sbarrays:DER objectGUID="8493bde3-afca-411a-8234-d66f550090c3"
utility="String">
                  <msp:comments>Feel free to add a comment here</msp:comments>
                  <msp:primaryIdentifier identifierName="String"
identifierLabel="String">NA</msp:primaryIdentifier>
                  <DERType xmlns="" otherKind="String">Cogeneration</DERType>
                  <realPowerRatings xmlns="">
                        <realPowerRating>
                              <realPower
units="microW">3.1415901184082031</realPower>
                              <description>String</description>
                        </realPowerRating>
                  </realPowerRatings>
                  <reactivePowerRatings xmlns="">
                        <reactivePowerRating>
                              <reactivePower
units="Other">3.1415901184082031</reactivePower>
                              <description>String</description>
                        </reactivePowerRating>
                  </reactivePowerRatings>
                  <apparentPowerRatings xmlns="">
                        <extensions
xmlns="http://www.multispeak.org/V5.0/commonTypes">
                              <auto-generated_for_wildcard xmlns=""/>
                        </extensions>
                        <extensionsList
xmlns="http://www.multispeak.org/V5.0/commonTypes">
                              <extensionsItem>
                                    <extName>String</extName>
                                    <extValue units="kQh">String</extValue>
                                    <extType>duration</extType>
                              </extensionsItem>
                        </extensionsList>
                        <apparentPowerRating>
                              <apparentPower
units="Unknown">3.1415901184082031</apparentPower>
                              <description>String</description>
                        </apparentPowerRating>
                  </apparentPowerRatings>
                  <storagePowerRating xmlns=""
units="Unknown">3.1415901184082031</storagePowerRating>
                  <notificationTime xmlns=""
units="Months">3.1415901184082031</notificationTime>
                  <DERRealPowerRampRate xmlns="" realPowerUnits="PerUnit"
timeUnits="Years">3.1415901184082031</DERRealPowerRampRate>
            </sbarrays:DER>
      </n1:ArrayOfDER>
</n1:GetDERsByDERIDResponse>
```

Example CIM-based XML - getDERGroups

In the GETDER XSD 1 to *n* requests for DER can be made. The request simply needs to pass the identifier for the DERGroup. The response is a listing of all the DERMembers within a group. In the example below, two identifiers are passed, so the expectation is that the members for both groups will be returned in the associated Reply.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2012 rel. 2 (x64)
(http://www.altova.com)-->
<m:DERGroups xsi:schemaLocation="http://www.epri.com/2013/DERGroup#
createDERGroup.xsd" xmlns:m="http://www.epri.com/2013/DERGroup#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <m:DERGroup>
            <m:name>Example DER Group Name</m:name>
            <!-- Either mRID needs to be used, or the Names class -->
            <m:mRID>8493bde3-afca-411a-8234-d66f550090c3</m:mRID>
      </m:DERGroup>
      <m:DERGroup>
            <m:name>Another example DER Group Name</m:name>
            <!-- Either mRID needs to be used, or the Names class -->
            <m:mRID>2d4310a5-94dd-4d85-98f1-cf4c14a2b94b</m:mRID>
      </m:DERGroup>
</m:DERGroups>
```

Example CIM-Based XML - replyDERGroups

Table 3-5 illustrates the CIM Group query.

Table 3-5
Test: CIM DER Group query

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	DER Group query	http://54.210.233.171/	get	DERGroup	
2	DER Group query response	http://54.210.233.171/	reply	DERGroup	

Required data

Data Element	Data Type	Notes
DERGroup.mRID	UUID	Identifies the DER Group for which DER information is requested

Test 3 – Adding DER to a group

Figure 3-9 shows MultiSpeak adding members to a DER group.

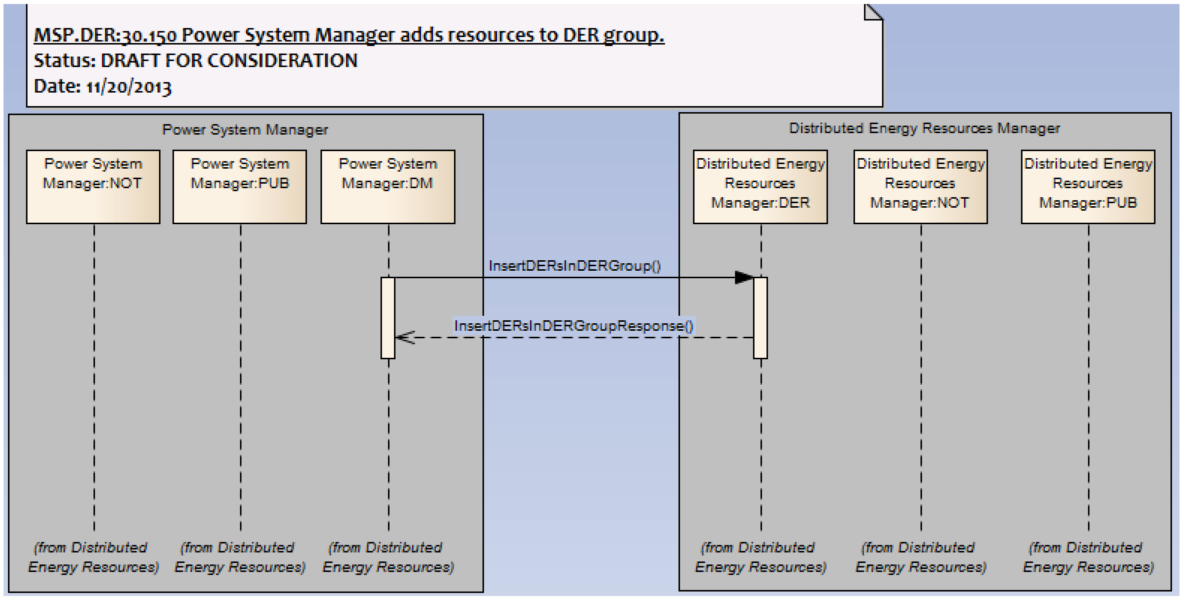

MSP.DER:30.150 Power System Manager adds resources to DER group.
Status: DRAFT FOR CONSIDERATION
Date: 11/20/2013
Power System Manager
Power System Manager:NOT
Power System Manager:PUB
Power System Manager:DM
Distributed Energy Resources Manager
Distributed Energy Resources Manager:DER
Distributed Energy Resources Manager:NOT
Distributed Energy Resources Manager:PUB
InsertDERsInDERGroup()
InsertDERsInDERGroupResponse()
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)

Figure 3-9
Example of MultiSpeak adding members to a DER group

Pre-Condition(s) A DER Group exists and has at least one DER in it.

Post Condition(s): The DER Group contains the all previous members, plus any additional members that were added as part of this test.

Table 3-6 shows MultiSpeak adding a DER member to a group; Table 3-7 shows CIM adding a DER member to a DERGroup; and Table 3-8 shows MultiSpeak adding a DER member to a DERGroup.

Table 3-6
Test: MultiSpeak - Adding a DER member to a group

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	DERGroup addition	http://54.210.233.171/	Insert	DERsInDERGroup	
2	DERGroup addition response	http://54.210.233.171/	Insert	DERsInDERGroupResponse	

Table 3-7
Test: CIM - Adding a DER member to a DERGroup

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	DERGroup addition	http://54.210.233.171/	change	DERGroup	
2	DERGroup addition response	http://54.210.233.171/	reply	DERGroup	

Table 3-8
Test: MultiSpeak Adding a DER member to a DERGroup

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	DERGroup addition	http://54.210.233.171/	Insert	DERsinDERGroup	
2	DERGroup addition response	http://54.210.233.171/	Insert	DERsinDERGroupResponse	

Example CIM-based XML changedDERGroup

This is similar to the initial createDERGroup example. The name and ID of this example is the same as the name and ID of the create example, but this adds an additional member to that “Example Group Name” group. Note that the new member is “Example Battery Storage” with its own unique mRID.

```
<?xml version="1.0" encoding="UTF-8"?>
<m:DERGroups xmlns:m="http://www.epri.com/2013/DERGroup#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.epri.com/2013/DERGroup#
file:///C:/Users/PGGR001/Documents/EPRI/067771%20-
%20DER/ExampleProfiles/CreateDERGroup.xsd">
      <m:DERGroup>
            <m:name>Example Group Name</m:name>
            <m:mRID>f975be36-a3b6-499c-9518-5e74555b6db9</m:mRID>
            <m:DERMember>
                  <m:mRID>77075dea-eb70-49fb-ae8b-16848c4767fa</m:mRID>
                  <m:name>Example Battery storage</m:name>
            </m:DERMember>
      </m:DERGroup>
</m:DERGroups>
```

Example MultiSpeak-based XML InsertDERsInDERGroup

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2015 sp1 (x64)
(http://www.altova.com)-->
<InsertDERsInDERGroup
xmlns="http://www.multispeak.org/V5.0/wsdl/SB DER Server"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:com="http://www.multispeak.org/V5.0/commonTypes"
xmlns:sbarrays="http://www.multispeak.org/V5.0/sandboxArrays"
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server
SB DER Server.xsd">
      <ArrayOfDERID>
            <sbarrays:DERID DERName="Example Battery storage"
objectGUID="77075dea-eb70-49fb-ae8b-16848c4767fa" owner="String"
registeredName="" systemName="">String</sbarrays:DERID>
      </ArrayOfDERID>
      <DERGroupID>
            <!-- ID of the DERGroup to insert into -->
            <com:objectGUID>f975be36-a3b6-499c-9518-
5e74555b6db9</com:objectGUID>
            <com:primaryIdentifier identifierName="String"
identifierLabel="NA">String</com:primaryIdentifier>
      </DERGroupID>
      <!-- example response url - this will depend on the test and on the
system sending the request -->
      <responseURL>www.epri.com/DERMS...</responseURL>
      <transactionID>String</transactionID>
</InsertDERsInDERGroup>
```

Example MultiSpeak XML – InsertDERsInDERGroupResponse

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<InsertDERsInDERGroupResponse
xmlns="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server
SB_DER_Server.xsd"/>
```

Test 4 – Removing a DER from a group

Figure 3-10 illustrates an example of MultiSpeak removal of members from a DER group.

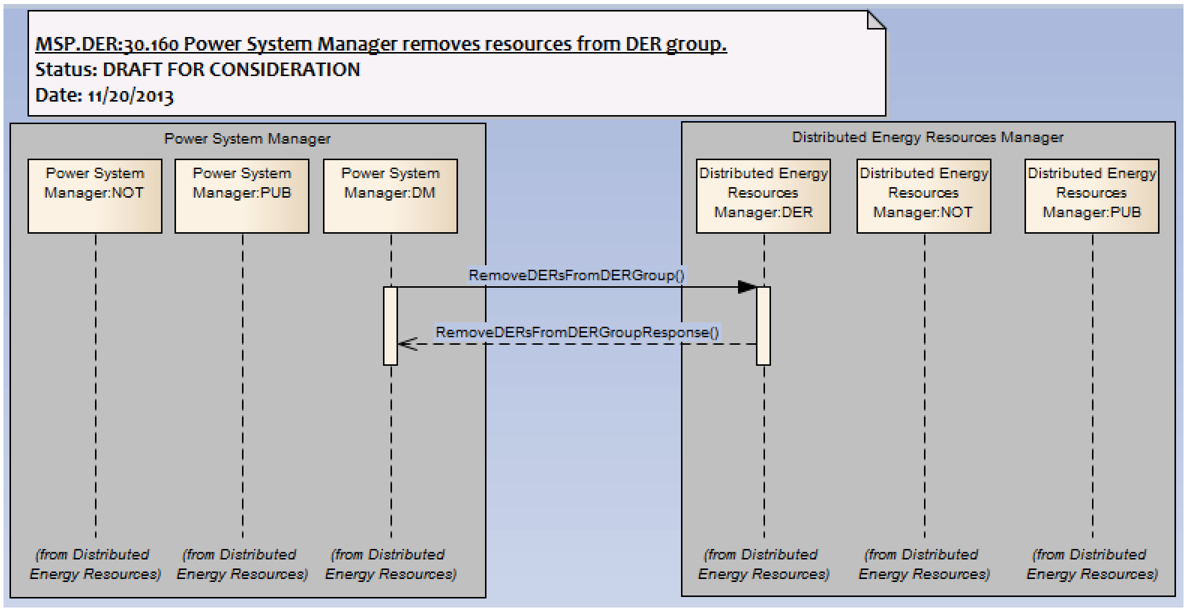

MSP.DER:30.160 Power System Manager removes resources from DER group.
Status: DRAFT FOR CONSIDERATION
Date: 11/20/2013
Power System Manager
Power System Manager:NOT
Power System Manager:PUB
Power System Manager:DM
Distributed Energy Resources Manager
Distributed Energy Resources Manager:DER
Distributed Energy Resources Manager:NOT
Distributed Energy Resources Manager:PUB
RemoveDERsFromDERGroup()
RemoveDERsFromDERGroupResponse()
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)

Figure 3-10
Example of MultiSpeak removal of members from a DER group

Pre-Condition(s) A DER Group exists and has at least one DER in it.

Post Condition(s): The group membership is reduced by the number of members for which it has received ID's for.

Table 3-9 shows MultiSpeak removing DER from DER group and Table 3-10 CIM removing DER from DER group.

Table 3-9
MultiSpeak - Remove DER from DER group

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	DERGroup member removal	http://54.210.233.171/	Remove	DERsFromDERGroup	
2	DERGroup removal response	http://54.210.233.171/	Remove	DERsFromDERGroupResponse	

Required Data

Data Element	Data Type	Notes
DERGroup.objectID	UUID	Identifies the DER Group for which updates will be made
DERMember.objectID	UUID	Identifies the DER member(s) to be removed from the group

Table 3-10
CIM - Remove DER from DER group

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	DERGroup member removal	http://54.210.233.171/	execute	DERGroup	
2	DERGroup removal response	http://54.210.233.171/	reply	DERGroup	

Note that removing a member, attribute, or element from an object in the CIM requires the use of the Execute verb and an OperationDataSet (otherwise the use of the remove action would assume that the entire object is being removed).

Required Data

Data Element	Data Type	Notes
DERGroup.mRID	UUID	Identifies the DER Group for which updates will be made
DERMember.mRID	UUID	Identifies the DER member(s) to be removed from the group

Example MultiSpeak XML – RemoveDERsFromDERGroup

Note that this example uses the same DER that was added to the group in the previous test step. Using the same DER is not required, only that a DER is pre-existing before any removal attempt.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<RemoveDERsFromDERGroup
xmlns="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:com="http://www.multispeak.org/V5.0/commonTypes"
xmlns:sbarrays="http://www.multispeak.org/V5.0/sandboxArrays"
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server
SB_DER_Server.xsd">
      <ArrayOfDERID>
            <sbarrays:DERID DERName="Example Battery storage"
objectGUID="77075dea-eb70-49fb-ae8b-16848c4767fa" owner="String"
registeredName="" systemName="">String</sbarrays:DERID>
      </ArrayOfDERID>
      <DERGroupID>
            <com:objectGUID>f975be36-a3b6-499c-9518-
5e74555b6db9</com:objectGUID>
            <com:primaryIdentifier identifierName="String"
identifierLabel="String">NA</com:primaryIdentifier>
      </DERGroupID>
      <responseURL>http://www.altova.com/</responseURL>
      <transactionID>Some made up identifier, e.g. DER-REMOVE-
1</transactionID>
</RemoveDERsFromDERGroup>
```

Example MultiSpeak XML – RemoveDERsFromDERGroupResponse

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<RemoveDERsFromDERGroupResponse
xmlns="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server
SB_DER_Server.xsd"/>
```

Example CIM-based XML – executeDERGroup

As noted above, removing a DER member from a DERGroup requires the use of the execute verb and an OperationSet (for an explanation of this use case refer to IEC 61968-9 Meter Reading and Control 2nd Edition, Annex L).

For clarity, an XML example of the entire CIM-based header is included showing the relationship between header, payload, and the OperationSet

```
<Header>
        <Verb>execute</Verb>
        <Noun>OperationSet</Noun>
        <Revision>1.0</Revision>
        <Timestamp>2014-12-20T09:30:47Z</Timestamp>
        <Source>DMS</Source>
        <AckRequired>true</AckRequired>
        <MessageID>07515735-060C-4125-9A1F-E75AEA19CA76</MessageID>
        <CorrelationID>07515735-060C-4125-9A1F-E75AEA19CA76</CorrelationID>
</Header>
<Payload>
        <OperationSet>
                   <Operation>
                              <operationID>1</operationID>
                              <noun>DERGroup</noun>
                              <verb>delete</verb>
                              <!—tell receiving system the operation is on the
element, not the object→
                              <elementOperation>true</elementOperation>
      <m:DERGroup>
                  <!—name/ID of the group to be changed →
                  <m:name>Example Group Name</m:name>
                  <m:mRID>f975be36-a3b6-499c-9518-5e74555b6db9</m:mRID>
                  <m:DERMember>
                        <m:mRID>77075dea-eb70-49fb-ae8b-16848c4767fa</m:mRID>
                        <m:name>Example Battery storage</m:name>
                  </m:DERMember>
            </m:DERGroup>
      </m:DERGroups>
                        </Operation>
   </OperationSet>
</Payload>
```

Example CIM-based XML - replyDERGroup

```
<?xml version="1.0" encoding="UTF-8"?>
<ResponseMessage
    xmlns = "http://iec.ch/TC57/2011/schema/message"
    xmlns:xsi = "http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation = "http://iec.ch/TC57/2011/schema/message Message.xsd">
    <Header>
        <Verb>reply</Verb>
        <Noun>DERGroup</Noun>
        <Revision>1.0</Revision>
        <Timestamp>2014-01-21T09:30:48Z</Timestamp>
        <Source>DERMS</Source>
        <MessageID>F5BA23D4-E1F7-4889-AB2A-CF3ED2BB06C9</MessageID>
        <CorrelationID>805C655B-4429-44C4-91C5-7692810627A7</CorrelationID>
    </Header>
    <Reply>
        <Result>OK</Result>
        <Error>
            <code>0.0</code>
        </Error>
    </Reply>
</ResponseMessage>
```

Test 5 – DER Group deletion

Just a as groups can be created, there may be a need to delete a group, as shown in the MultiSpeak sequence diagram in Figure 3-11.

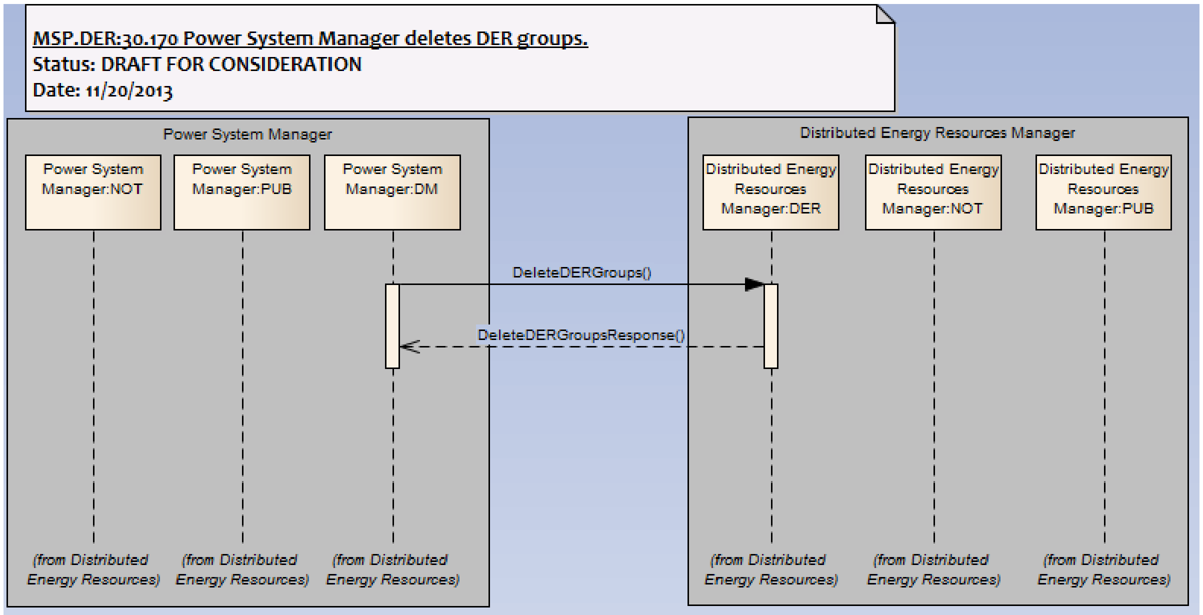

MSP.DER:30.170 Power System Manager deletes DER groups.
Status: DRAFT FOR CONSIDERATION
Date: 11/20/2013
Power System Manager
Power System Manager:NOT
Power System Manager:PUB
Power System Manager:DM
Distributed Energy Resources Manager
Distributed Energy Resources Manager:DER
Distributed Energy Resources Manager:NOT
Distributed Energy Resources Manager:PUB
DeleteDERGroups()
DeleteDERGroupsResponse()
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)

Figure 3-11
Example of MultiSpeak deletion of a DER group

Pre-Condition(s)	A DER Group exists and has at least one DER in it.
Post Condition(s):	The group is deleted

Table 3-11 and Table 3-12 show MultiSpeak DER group deletion and CIM DER group deletion.

Table 3-11
MultiSpeak - DER group deletion

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	DERGroup deletion	http://54.210.233.171/	Delete	DERGroups	
2	DERGroup deletion	http://54.210.233.171/	Delete	DERGroupsResponse	

Required Data

Data Element	Data Type	Notes
DERGroup.objectID	UUID	Identifies the DER Group to be deleted
DERGroup.name	string	

Table 3-12
CIM - DER Group deletion

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	DERGroup deletion	http://54.210.233.171/	delete	DERGroup	
2	DERGroup deletion	http://54.210.233.171/	reply	DERGroup	

Required Data

Data Element	Data Type	Notes
DERGroup.mRID	UUID	Identifies the DER Group to be deleted

Example MultiSpeak XML – RemoveDERsFromDERGroup

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<DeleteDERGroups xmlns="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:com="http://www.multispeak.org/V5.0/commonTypes"
xmlns:sbarrays="http://www.multispeak.org/V5.0/sandboxArrays"
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server
SB DER Server.xsd">
      <ArrayOfDERGroupID>
            <sbarrays:DERGroupID>
                  <com:objectGUID>f975be36-a3b6-499c-9518-
```

```
5e74555b6db9</com:objectGUID>
                  <com:primaryIdentifier identifierName="String"
identifierLabel="String">NA</com:primaryIdentifier>
            </sbarrays:DERGroupID>
      </ArrayOfDERGroupID>
      <responseURL>http://www.altova.com/</responseURL>
      <transactionID>Made up identifier, e.g. MSP-REMOVE DERGroup-
1</transactionID>
</DeleteDERGroups>
```

Example MultiSpeak XML – RemoveDERsFromDERGroupResponse

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<DeleteDERGroupsResponse
xmlns="http://www.multispeak.org/V5.0/wsdl/SB DER Server"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server
SB_DER_Server.xsd"/>
```

CIM-based Example XML - deleteDERGroup

Note: The example XML for the deletion of a group is the same as for the creation of a group. The only thing that has changed is the *name* of the end point. Instead of using CreateDERGroup (for CIM-based web services) DeleteDERGroup is used.

Example CIM-based XML - deleteDERGroup

```
<?xml version="1.0" encoding="UTF-8"?>
<m:DERGroups xmlns:m="http://www.epri.com/2013/DERGroup#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.epri.com/2013/DERGroup#
file:///C:/Users/PGGR001/Documents/EPRI/067771%20-
%20DER/ExampleProfiles/DeleteDERGroup.xsd">
      <m:DERGroup>
            <m:name>Example Group Name</m:name>
            <!-- for MultiSpeak this would be an objectID not an mRID -->
            <m:mRID>f975be36-a3b6-499c-9518-5e74555b6db9</m:mRID>
      </m:DERGroup>
</m:DERGroups>
```

Example CIM-based XML – replyDERGroup

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<DERGroups xmlns="http://www.epri.com/2013/DERGroup#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.epri.com/2013/DERGroup# DERGroup.xsd">
      <DERGroup description="text" comment="text">
            <name>Example DERGroup name</name>
            <!-- identifying used for the DERGroup, so the requestor can match
to the request that was made -->
            <mRID>f975be36-a3b6-499c-9518-5e74555b6db9</mRID>
      </DERGroup>
</DERGroups>
```

Test 6 – DERGroup Notification

Finally, there is a notification when changes have been made to DER groups for interested end points as illustrated in Figure 3-12. For more information about the Publication and Notification server implementation, see Section 7, *MultiSpeak Extension to Support Requirements*.

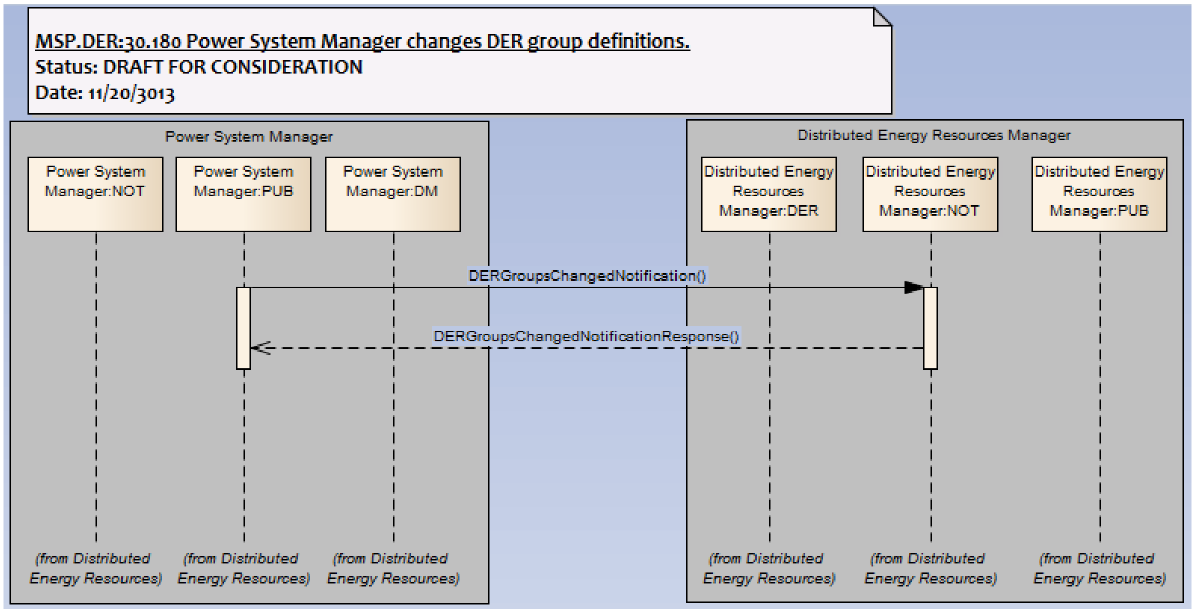

MSP.DER:30.180 Power System Manager changes DER group definitions.
Status: DRAFT FOR CONSIDERATION
Date: 11/20/3013
Power System Manager
Power System Manager:NOT
Power System Manager:PUB
Power System Manager:DM
Distributed Energy Resources Manager
Distributed Energy Resources Manager:DER
Distributed Energy Resources Manager:NOT
Distributed Energy Resources Manager:PUB
DERGroupsChangedNotification()
DERGroupsChangedNotificationResponse()
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)

Figure 3-12
Example MultiSpeak integration for publishing changes to DER Groups

Pre-Condition(s) A DER Group exists and has at least one DER in it.

Post Condition(s): For systems that have subscribed to change notifications, when a change occurs to a group, the subscribed system receives the notification

Note: For MultiSpeak, notification methods are in the NOT_Server.xsd, not in the DER_Server.xsd

Table 3-13 shows MultiSpeak DERGroup notification.

Table 3-13
MultiSpeak DERGroup Notification

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	DERGroup notification	http://54.210.233.171/		DERGroupsChangedNotification	
2	DERGroup notification	http://54.210.233.171/		DERGroupsChangedNotificationResponse	

Example MultiSpeak XML – DERGroupsChangedNotification

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2015 sp1 (x64)
(http://www.altova.com)-->
<n1:DERGroupsChangedNotification
xmlns="http://www.multispeak.org/V5.0/commonTypes"
xmlns:msp="http://www.multispeak.org/V5.0"
xmlns:cpsm="http://www.multispeak.org/V5.0/cpsm"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:sbarrays="http://www.multispeak.org/V5.0/sandboxArrays"
xmlns:n1="http://www.multispeak.org/V5.0/wsdl/SB NOT Server"
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_NOT_Server
SB NOT Server.xsd">
      <n1:ArrayOfDERGroup>
            <sbarrays:DERGroup objectGUID="f975be36-a3b6-499c-9518-
5e74555b6db9" utility="String">
                  <msp:comments>Place your comment here</msp:comments>
                  <msp:primaryIdentifier identifierName="String"
identifierLabel="String">NA</msp:primaryIdentifier>
                  <DERIDs xmlns="">
                        <DERID DERName="Example Battery storage"
objectGUID="77075dea-eb70-49fb-ae8b-16848c4767fa" owner="String"
registeredName="" systemName="">String</DERID>
                  </DERIDs>
            </sbarrays:DERGroup>
      </n1:ArrayOfDERGroup>
      <n1:transactionID>Some made up ID, e.g. DERGroupsChangedNotification-
1</n1:transactionID>
</n1:DERGroupsChangedNotification>
```

Example MultiSpeak XML – DERGroupsChangedNotificationResponse

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2015 sp1 (x64)
(http://www.altova.com)-->
<DERGroupsChangedNotificationResponse
xmlns="http://www.multispeak.org/V5.0/wsdl/SB NOT Server"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_NOT_Server
SB_NOT_Server.xsd"/>
```

Note: There is no standard CIM integration convention for publish/subscribe notification. To get the latest group information, a system would query for groups/members as in Test 2.

Test 7 - DER Group Status Monitoring

This section describes a method by which the present status of DER groups may be exchanged between software applications in an enterprise integration environment. It requires that the referenced DER group definition exists in both the status-requesting and status-providing entities. As described in the previous section, the makeup of the group could have been defined by the requestor, the provider, or any other entity, and could have been a manual or automated process.

Figure 3-13 illustrates a pulled status monitoring sequence, with the group creation process assumed to have occurred at some point prior. Figure 3-14 illustrates a pushed status update.

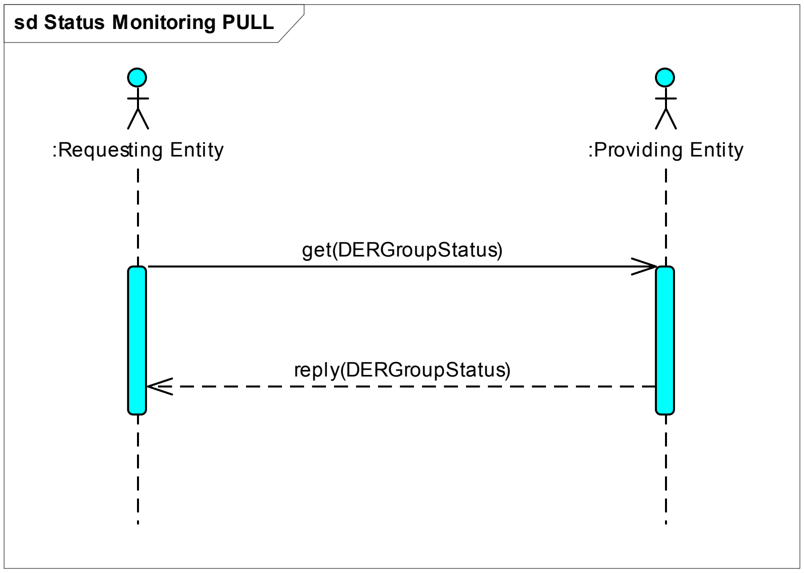


Figure 3-13
Pulled Status Monitoring Sequence Diagram

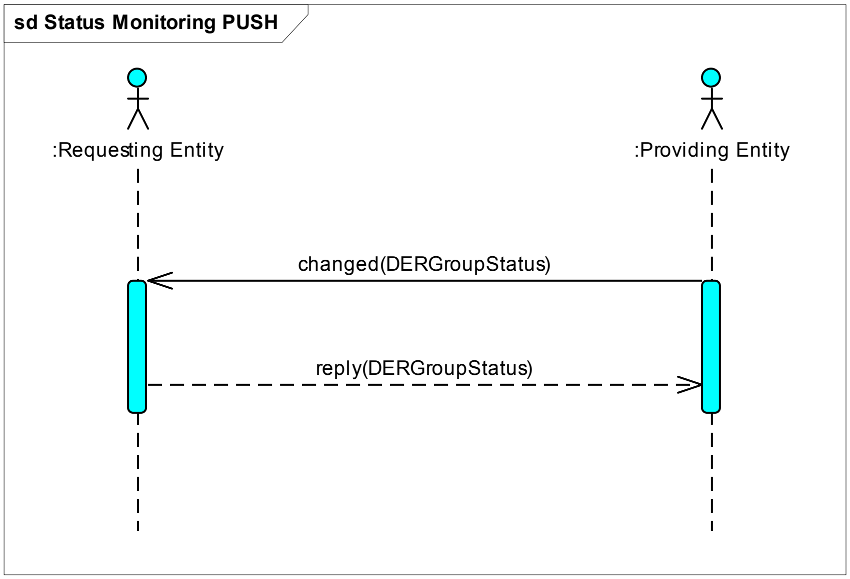


Figure 3-14
Pushed Status Monitoring Sequence Diagram

Table 3-14 illustrates CIM status monitoring (PULL).

Table 3-14
CIM Status Monitoring (PULL)

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	DER status	http://54.210.233.171/	get	DERStatus	
2	DER status	http://54.210.233.171/	reply	DERStatus	

Required Data

Data Element	Data Type	Notes
DERGroup.name	String	
DERGroup.mRID	UUID	Identifies the DER Group for which status is requested; not required per se, but if not used the Names class must be used

Example MultiSpeak-based XML – GetDERGroupStatusesByDERGroupIDs

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<n1:GetDERGroupStatusesByDERGroupIDs
xmlns="http://www.multispeak.org/V5.0/commonTypes"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:sbarrays="http://www.multispeak.org/V5.0/sandboxArrays"
xmlns:n1="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server"
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server
SB_DER_Server.xsd">
      <n1:ArrayOfDERGroupID>
            <sbarrays:DERGroupID>
                  <objectGUID>4152684d-2640-4935-a96c-
449edbd9aff7</objectGUID>
                  <primaryIdentifier identifierName="String"
identifierLabel="String">NA</primaryIdentifier>
            </sbarrays:DERGroupID>
      </n1:ArrayOfDERGroupID>
</n1:GetDERGroupStatusesByDERGroupIDs>
```

Example MultiSpeak-based XML – GetDERGroupStatusesByDERGroupIDsResponse

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<n1:GetDERGroupStatusesByDERGroupIDsResponse
xmlns="http://www.multispeak.org/V5.0/commonTypes"
xmlns:msp="http://www.multispeak.org/V5.0"
xmlns:cpsm="http://www.multispeak.org/V5.0/cpsm"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:sbarrays="http://www.multispeak.org/V5.0/sandboxArrays"
xmlns:n1="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server"
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server
```

```
SB_DER_Server.xsd">
      <n1:ArrayOfDERGroupStatus>
            <sbarrays:DERGroupStatus referableID="String">
                  <!-- the time and pertinent status information -->
                  <msp:timeStamp>2001-12-17T09:30:47Z</msp:timeStamp>
                  <msp:eventDescription>Example
description</msp:eventDescription>
                  <msp:eventReason>Example reason</msp:eventReason>
                  <!-- ID of the DERGroup -->
                  <DERGroup xmlns="" objectGUID="4152684d-2640-4935-a96c-
449edbd9aff7" utility="String">
                        <msp:comments>an example comment</msp:comments>
                        <msp:primaryIdentifier identifierName="String"
identifierLabel="String">NA</msp:primaryIdentifier>
                  </DERGroup>
                  <DERGroupCapabilities xmlns="">
                        <!-- capabilities at the time of this status -->
                        <realPower units="mW">3.1415901184082031</realPower>
                        <reactivePower
units="Other">3.1415901184082031</reactivePower>
                        <apparentPower
units="kVA">3.1415901184082031</apparentPower>
                  </DERGroupCapabilities>
            </sbarrays:DERGroupStatus>
      </n1:ArrayOfDERGroupStatus>
</n1:GetDERGroupStatusesByDERGroupIDsResponse>
```

Example CIM-based XML - getDERGroupStatus

Note: In this example a single request has been used to request status from two different DERGroups, using the mRID as the identifier.

```
<?xml version="1.0" encoding="UTF-8"?>
<m:DERGroupStatuses xmlns:m="http://www.epri.com/2013/DERStatus#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.epri.com/2013/DERStatus#
file:///C:/Users/PGGR001/Documents/EPRI/067771%20-
%20DER/ExampleProfiles/GetDERGroupStatus.xsd">
      <m:DERGroupStatus>
            <m:RequestedCapability>Watts</m:RequestedCapability>
            <m:DERGroup>
                  <m:name>Example DERGroup Name</m:name>
                  <!-- an mRID is not required; if one isn't used then Names
class must be used -->
                  <m:mRID>f975be36-a3b6-499c-9518-5e74555b6db9</m:mRID>
            </m:DERGroup>
      </m:DERGroupStatus>
      <m:DERGroupStatus>
            <m:RequestedCapability>ApparentPower</m:RequestedCapability>
            <m:DERGroup>
                  <m:name>Another Example Group Name</m:name>
                  <!-- mrID is not required, but if it isn't used the Names
class must be used -->                    <m:mRID>3cc0687b-b530-4924-9542-
f51518e44504</m:mRID>
            </m:DERGroup>
      </m:DERGroupStatus>
</m:DERGroupStatuses>
```

Example CIM-based XML – replyDERGroupStatus

In this example note that since two statuses were asked for, two were received.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<DERGroups xmlns="http://www.epri.com/2013/DERGroup#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.epri.com/2013/DERGroup# CreateDERGroup.xsd">
      <DERGroup description="text" comment="text">
            <status>
                  <dateTime>2001-12-17T09:30:47Z</dateTime>
                  <reason>Because you asked for a status</reason>
                  <remark>No remark</remark>
                  <value>0 - OK</value>
            </status>
            <name>Example DERGroup</name>
            <mRID>f975be36-a3b6-499c-9518-5e74555b6db9</mRID>
      </DERGroup>
      <DERGroup description="text" comment="text">
            <status>
                  <dateTime>2001-12-17T09:30:47Z</dateTime>
                  <reason>Because you asked for a status</reason>
                  <remark>No remark</remark>
                  <value>-1 - Something has gone awry</value>
            </status>
            <name>Another Example DERGroup name</name>
            <mRID>3cc0687b-b530-4924-9542-f51518e44504</mRID>
      </DERGroup>
</DERGroups>
```

Table 3-15 shows CIM status monitoring (PUSH).

Table 3-15
CIM Status Monitoring (PUSH)

While any service pattern could be architected in either PUSH or PULL pattern, this particular test is shown as an example only. For the 2014 test event, only the PULL pattern will be tested.

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	DER status	http://54.210.233.171/	changed	DERStatus	
2	DER status	http://54.210.233.171/	reply	DERStatus	

Required Data

Data Element	Data Type	Notes
DERGroup.mRID	UUID	Identifies the DER Group for which status is being reported
Capabilities .Apparent Power .Real Power		This will be 1 to many capabilities for each member of the group

Figure 3-15 shows a MultiSpeak example of DER status monitoring.

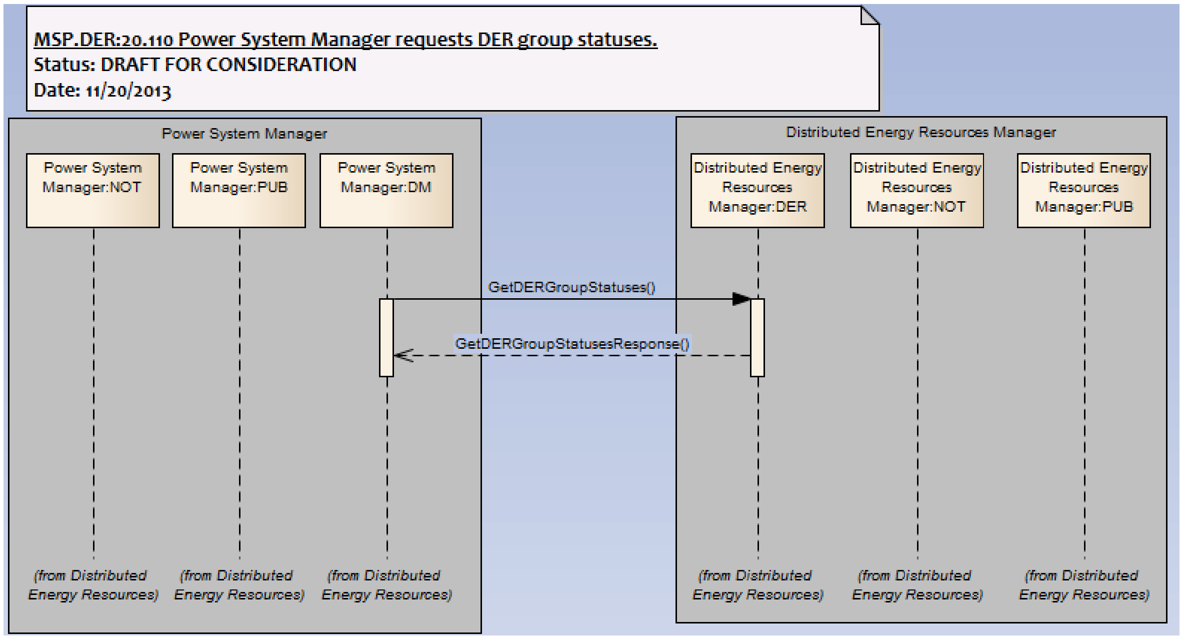

MSP.DER:20.110 Power System Manager requests DER group statuses.
Status: DRAFT FOR CONSIDERATION
Date: 11/20/2013
Power System Manager
Power System Manager:NOT
Power System Manager:PUB
Power System Manager:DM
Distributed Energy Resources Manager
Distributed Energy Resources Manager:DER
Distributed Energy Resources Manager:NOT
Distributed Energy Resources Manager:PUB
GetDERGroupStatuses()
GetDERGroupStatusesResponse()
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)

Figure 3-15
MultiSpeak example of DER status monitoring

Status Request Timing Types

The requesting entity may specify that the status is one of two types:

- **Latest Available:** A request for the latest status available for the DER group. This would notionally be based on information that the providing entity already collected from the individual DER. The response to this kind of status request could typically be provided immediately.
- **Refreshed Status:** A request that instructs the providing entity to go out and get updated status information from the DER group before responding. The response to this kind of request would be delayed until the refreshed status could be collected from the DER in the field.

Figure 3-13 and Figure 3-14 display a "PULL" sequence (a requestor making a request of the system holding the data), and a "PUSH" model in which the providing entity could publish the information on a schedule, or perhaps whenever the status changed.

Clarifying the Difference between Capabilities and Status

During the project kickoff workshop, "Status Monitoring" and "Capabilities Discovery" were identified and defined as separate services. Based on discussion from the workshop, the distinction between these two was:

- "Status" represents the present state and can be time-variant in the sense of the potential for moment-to-moment changes during operation.

- "Capabilities" are name-plate oriented. Capabilities change when infrastructure is added or deleted and the associated newly installed capabilities are entered into GIS (or other system of record).

For some parameters, the present status may be represented by three quantities: the present value, the maximum value to which it can presently be adjusted, and the minimum value to which it can presently be adjusted. All are included as part of the status information. Figure 3-16 illustrates the concept.

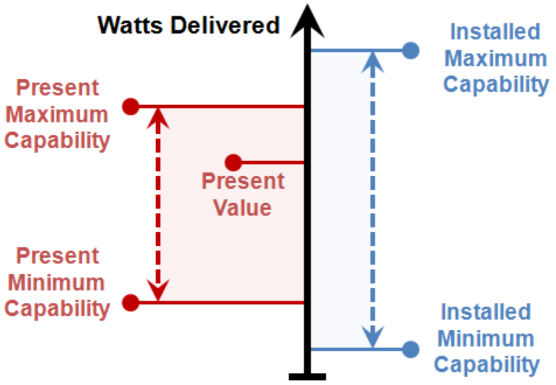

Watts Delivered
Installed Maximum Capability
Present Maximum Capability
Present Value
Present Minimum Capability
Installed Minimum Capability

Figure 3-16
Installed (Blue) and Present (Red) Capabilities

The quantities shown in red will be part of the "status" information and those shown in blue will be part of the "capabilities" information. Defined in this way, the present values will always fall inside (or equal to) the installed capabilities range. For example, the "Present Maximum Capability" of a PV group could be less than the "Installed Maximum Capability" on a cloudy day, or if some of the DER in the group are presently offline or impaired in some way.

Likewise, the "Present Value" will always fall inside (or equal to) the present capabilities range. For example, the present Watts Delivered output from a PV group could be equal to the "Present Maximum Capability" if the group is not curtailed in any way. Alternatively, the present Watt output of the same group could be less than the "Present Maximum Capability" if some units are operating in a curtailed mode.

All three "present" quantities can be time-varying, based on a variety of factors, such as battery systems being fully charged or fully discharged, solar irradiance being high or low, equipment being online/offline, contracts, or local voltage limits.

The objective in this approach is that a requesting entity (DMS for example) could request the Watt status (for a group of DER), and get three numbers: a present value, a capability to deliver and a capability to receive. The requesting entity could then make a request for a Watt value within the red range and get the expected response, more or less.

Test 8 DER Group Capabilities Discovery

This section describes a method by which the capabilities of Distributed Energy Resources might be exchanged between software applications in an enterprise integration environment. This function is specifically focused on installed capability (static, Non-variable quantities), not real-time status data which are variable in nature. The "Status Monitoring" function is defined in the previous section.

This proposal is not intended to be architecturally-prescriptive in terms of what software applications exist, what functionality is grouped in any one software application, or what interactions are employed between applications. It is intended to identify a range of common enterprise interactions, flexible such that any architecture might be supported. This "DER Capabilities" function may or may not be found useful in all cases.

"Installed Capabilities" vs. "Initial Entry or Discovery"

This "Installed Capabilities" function is intended as a mechanism for the exchange of information between software applications in an enterprise environment. This function is not intended to relate directly to communication with DER in the field, or to provide a mechanism by which DER capabilities might be automatically registered into the utility system (plug-n-play DER). Figure 3-17 illustrates the difference. An initial discovery process (Step 1) is assumed to have taken place. This initial process could occur in a number of ways, all of which are out of scope for this body of work:

- A human process by which DER capabilities data is manually entered into the system model
- An automated process by which newly connected DER are discovered and described in a plug-n-play fashion
- Any number of other methods

Once the capabilities are known to some enterprise application, a subsequent process (Step 2) may occur in which these capabilities are shared between software applications. It is this subsequent process that this function is intended to address.

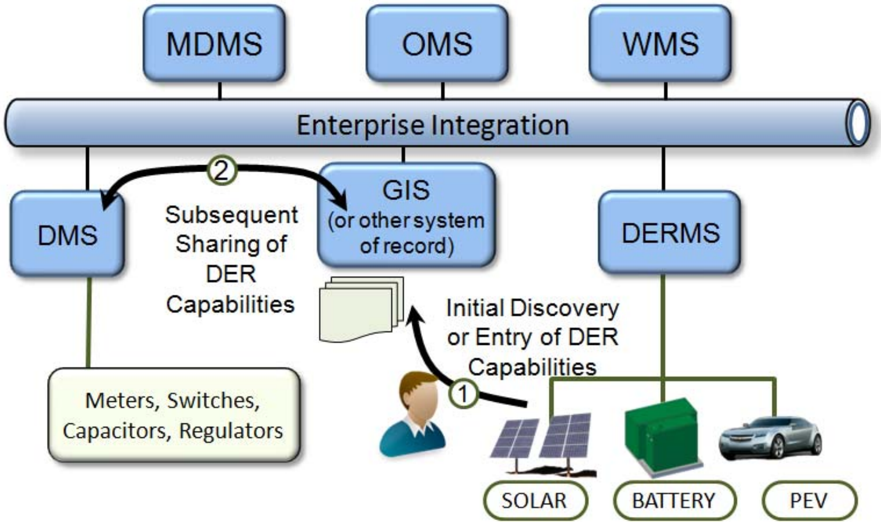

MDMS
OMS
WMS
Enterprise Integration
DMS
2
Subsequent Sharing of DER Capabilities
GIS (or other system of record)
DERMS
Meters, Switches, Capacitors, Regulators
Initial Discovery or Entry of DER Capabilities
1
SOLAR
BATTERY
PEV

Figure 3-17
Enterprise Exchange of DER Group Capabilities

Requesting DER Group Capabilities

This function envisions an environment in which one entity knows the present installed capabilities of the DER group and another entity seeks to understand this capability. As an example, consider a scenario in which an application called a Geospatial Information System (GIS) is the system of record for installed DER. This system has information regarding the installed capabilities of DER. At some point in time, a separate application, such as a Distribution Management System, seeks to understand the installed capabilities of a particular group of DER, and this function is utilized.

Like the "Status Monitoring" function described in the previous section, this function has a range of requirements for how capabilities might be requested, including various levels of aggregation. To satisfy these needs, the groups-based approach is used for this function.

As illustrated inFigure 3-18, the sequence is similar, except that the GET and REPLY are for capabilities information.

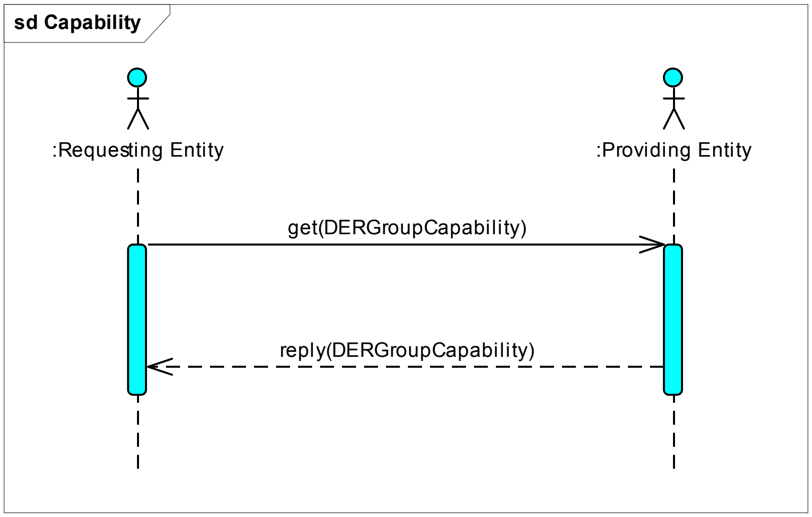


Figure 3-18
CIM Capabilities Sequence Diagram

The approach involves a first interaction in which a DERGroup is defined by one entity (e.g. the DERCapability-requesting entity) and provided to the DERCapability-providing entity. Group creation could occur immediately before the second interaction (DERCapability request and reply) or any time prior. The group definitions used for capabilities may be the same as those used for status monitoring or other dispatch calls.

DER Group Capabilities Information

The difference between capabilities information and status information has been described in the status monitoring section, and illustrated in Figure 3-16.

Summary of DER Group Capability Information Items

The following information is passed in a request for DER Group Capabilities:

This message profile contains the same classes and attributes as the DERStatus profile, the only difference being the intent. DERStatus conveys what the DER group is doing at a moment in time; while, as noted earlier in this section, the DERCapability reflects the installed/nameplate characteristics of the group of DER. The designed flexibility of the DERStatus profile allows for a range of capabilities to be passed between systems. The other difference being that a DERStatus request might be for a single capability, e.g. RealPower, while the DERCapabilities request would more likely be asking for everything that the DER was capable of.

Since the DERStatus profile provides the option to pass 1-to-many capabilities, plus additional capabilities that are not envisioned at this time, no new profile needs to be created to support the response to this request, only the name of the payload will change to reflect the capabilities method.

Pre-Condition(s)	A DER Group must exist in the responding system with corresponding members that have their capabilities stored.
Post Condition(s):	The requesting system will receive the DER capabilities of the DERGroup for which it asked

Note: There is no test for the DERGroup Capabilities for MultiSpeak

Table 3-16 shows a CIM DERGroup capabilities request.

Table 3-16
CIM DERGroup Capabilities Request

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	Get	http://54.210.233.171/	create	DERGroupCapabilities	
2	Reply	http://54.210.233.171/	reply	DERGroupCapabilitiesResponse	

Required data:

Data Element	Data Type	Notes
DERGroup.objectID	UUID	Identifies the DER Group
DERGroup.primaryIdentifier	String	Set to “NA”

Example CIM-based XML – getDERGroupCapabilities

See: DERGroupStatus

Note: In CIM-based exchanges this is the same pattern and XSD as for getDERGroupStatus. The contents of the message will change. For instance, there will be no status passed, only the current capabilities and the information object. DERGroupCapabilities will be used instead of DERGroupStatus. See the CIM-based status example from Test 7.

Test 9 DER Group Dispatch

This section describes a method by which the real power level (or other capability) of a DER group may be managed. This method is in the form of a request that the real power for the group be set to a specified level.

This function is intended to apply between software applications in an enterprise integration environment. As such, it does not have direct bearing on how individual DERs within the group are managed. For example, if this function requests that the real power output from a group of 10 DER be reduced to a level that is 100kW less than the present value, it may be satisfied by each DER being reduced by 10kW, one DER being reduced by the whole 100kW, or any other distribution. The algorithms and methods by which individual DER are managed is out of scope and is viewed as the responsibility of the entity directly managing the DER, such as a DERMS.

The previously-defined method of using arbitrarily-defined groups is also used for this function. It requires that the referenced DER group definition (the list of which DER make up the group) is known and agreed-to by both the real-power-requesting and real power-providing entities. As described previously, the makeup of the group could have been defined by the requestor, the provider, or any other entity, and its creation could have been a manual or automated process.

Figure 3-19 illustrates a real power dispatch sequence, with the group creation process assumed to have occurred at some point prior. In addition, this example shows that a status monitoring request may have also preceded the real power dispatch (shown as the UML "rectangle" referencing the status monitoring use case) so that the requesting entity may know what range of adjustability is presently possible.

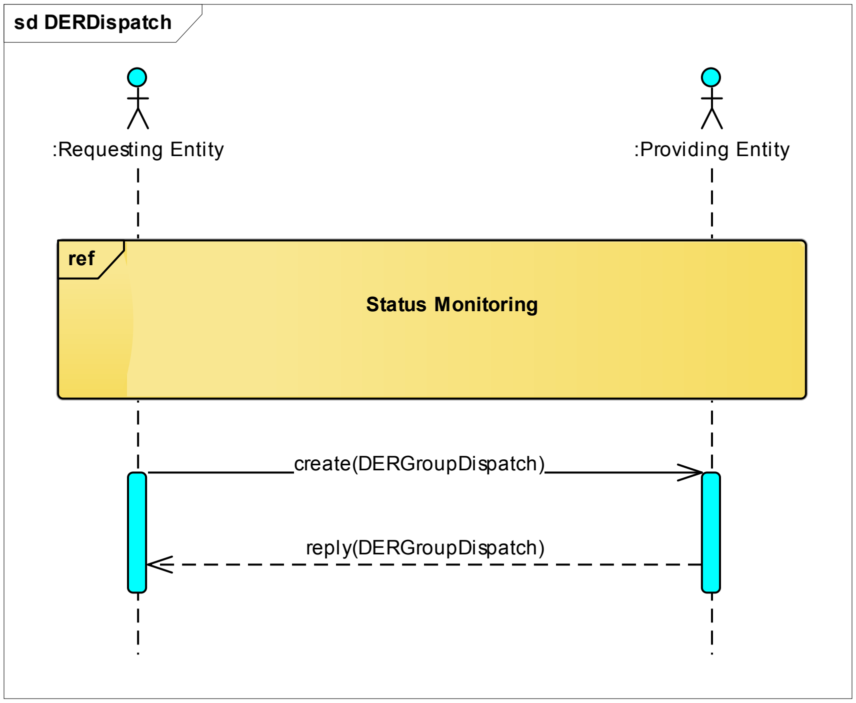


Figure 3-19
CIM DERGroup Dispatch

Figure 3-20 shows an example of a MultiSpeak DER Group dispatch request.

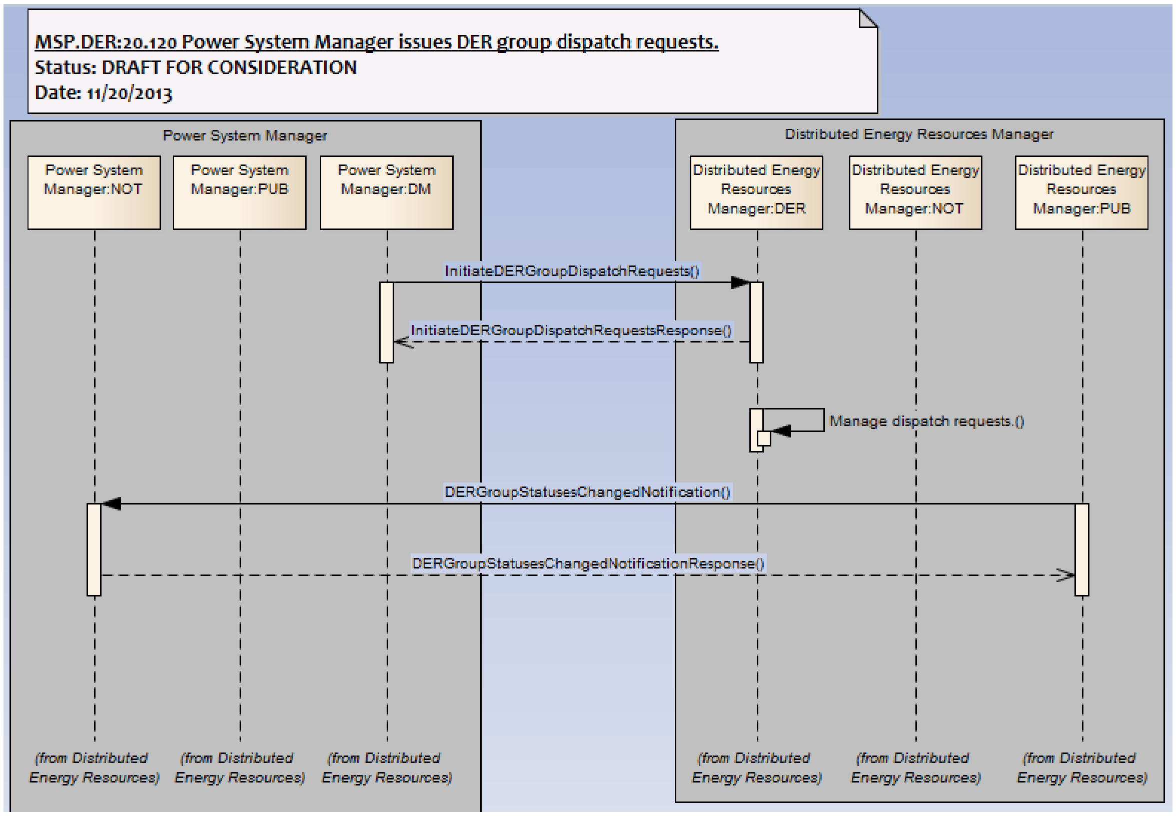

MSP.DER:20.120 Power System Manager issues DER group dispatch requests.
Status: DRAFT FOR CONSIDERATION
Date: 11/20/2013
Power System Manager
Power System Manager:NOT
Power System Manager:PUB
Power System Manager:DM
Distributed Energy Resources Manager
Distributed Energy Resources Manager:DER
Distributed Energy Resources Manager:NOT
Distributed Energy Resources Manager:PUB
InitiateDERGroupDispatchRequests()
InitiateDERGroupDispatchRequestsResponse()
Manage dispatch requests.()
DERGroupStatusesChangedNotification()
DERGroupStatusesChangedNotificationResponse()
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)

Figure 3-20
Example of MultiSpeak DER Group dispatch request

Group creation (required) and status monitoring (optional) could occur immediately before the Real Power Dispatch (request and reply) or any time prior. The group definitions used for real power dispatch would notionally be the same as those used for status monitoring. Table 3-17 shows MultiSpeak DERGroup dispatch and Table 3-18 shows CIM DERGroup Dispatch.

Table 3-17
MultiSpeak DERGroup Dispatch

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	Get	http://54.210.233.171/	Create	DERGroupCapabilities	
2	Reply	http://54.210.233.171/	Create	DERGroupCapabilitiesResponse	

Required data:

Data Element	Data Type	Notes
DERGroup.objectID	UUID	Identifies the DER Group
DERGroup.primaryIdentifier	String	Set to “NA”

Table 3-18
CIM DERGroup Dispatch

Step	Function	End Point	Verb	Payload Name	Pass / Fail
1	Get	http://54.210.233.171/	create	DERGroupCapabilities	
2	Reply	http://54.210.233.171/	reply	DERGroupCapabilitiesResponse	

Required data:

Data Element	Data Type	Notes
DERGroup.objectID	UUID	Identifies the DER Group
DERGroup.primaryIdentifier	String	Set to “NA”

Example MultiSpeak-based XML - InitiateDERGroupDispatchRequests

While DERGroup dispatch could be for any capability that is supported by the DERGroup, this particular example is for Vars.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<n1:InitiateDERGroupDispatchRequests
xmlns="http://www.multispeak.org/V5.0/commonTypes"
xmlns:msp="http://www.multispeak.org/V5.0"
xmlns:cpsm="http://www.multispeak.org/V5.0/cpsm"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:sbarrays="http://www.multispeak.org/V5.0/sandboxArrays"
xmlns:n1="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server"
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server
SB_DER_Server.xsd">
      <n1:ArrayOfDERGroupDispatchRequest>
            <!-- this is the GUID of the request -->
            <sbarrays:DERGroupDispatchRequest objectGUID="b1b699a9-1fc4-44de-
9113-00e6a8f0435d" utility="String">
                  <msp:comments>Your comments here</msp:comments>
                  <msp:primaryIdentifier identifierName="String"
identifierLabel="String">NA</msp:primaryIdentifier>
                  <DERGroupID xmlns="">
                        <!-- this is the GUID of the DERGroup -->
                        <objectGUID
xmlns="http://www.multispeak.org/V5.0/commonTypes">3832cf79-0c5a-405a-90f8-
f6cc6b37806e</objectGUID>
                        <primaryIdentifier
xmlns="http://www.multispeak.org/V5.0/commonTypes" identifierName="String"
identifierLabel="String">NA</primaryIdentifier>
                  </DERGroupID>
                  <dispatchTargets xmlns="">
                        <realPowerDispatchTarget
units="microW">3.1415901184082031</realPowerDispatchTarget>
                        <reactivePowerTarget
units="VAr">3.1415901184082031</reactivePowerTarget>
                  </dispatchTargets>
            </sbarrays:DERGroupDispatchRequest>
      </n1:ArrayOfDERGroupDispatchRequest>
      <n1:responseURL>http://www.altova.com/</n1:responseURL>
      <n1:transactionID>Test 9-InitiateDERDispatch request. Attempt:
#</n1:transactionID>
</n1:InitiateDERGroupDispatchRequests>
```

Example MultiSpeak-based XML - InitiateDERGroupDispatchRequestsResponse

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<InitiateDERGroupDispatchRequestsResponse
xmlns="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server
SB_DER_Server.xsd"/>
```

Example CIM-based XML – createDERGroupDispatch

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<DERGroupDispatches xmlns="http://www.epri.com/2013/DERGroup#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.epri.com/2013/DERGroup#
createDERGroupDispatch.xsd">
      <DERGroupDispatch>
            <DERGroup>
                  <name>Example DERGroup Name</name>
                  <mRID>b2dd9e07-6062-41b6-b5c8-afe1250beb9e</mRID>
            </DERGroup>
            <RequestedCapability>
                  <capabilityType>RealPower</capabilityType>
                  <value>3.1415901184082031</value>
                  <capabilityUnits>VA</capabilityUnits>
                  <capabilityMultiplier>k</capabilityMultiplier>
            </RequestedCapability>
      </DERGroupDispatch>
</DERGroupDispatches>
```

Example CIM-based XML – replyDERGroupDispatch

This reply simply echoes back the DERGroup name, and assumes a positive result (this requested dispatch occurred) unless an error occurred.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<DERGroups xmlns="http://www.epri.com/2013/DERGroup#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.epri.com/2013/DERGroup# CreateDERGroup.xsd">
      <DERGroup description="text" comment="text">
            <name>Example DERGroup</name>
            <mRID>b2dd9e07-6062-41b6-b5c8-afe1250beb9e </mRID>
      </DERGroup>
</DERGroups>
```

Test 10 DER Forecasting

This section describes a method by which forecasts of DER availability may be exchanged between software applications. Specifically, this method addresses forecasts for the availability of real and reactive power from a DER group. In future additions, forecasts could be made available in similar fashion for any monitorable or manageable DER parameter.

This function only defines how DER forecast data is exchanged and does not specify how forecasts are determined. Some DER forecast-providing entities could, for example, have access to detailed weather forecast information, including satellite or sky-viewing capabilities to enable prediction of solar variability. Others could monitor DER health or analyze historical data in order to determine forecasts with greater accuracy. Regardless of the forecasting methods that may be used, this function only addresses the exchange of the forecast of the DER availability (real and reactive power) and does not address the exchange of weather or other related data.

As described in a previous section, the present status for real and reactive power includes three parts: a present value, a maximum, and a minimum range of adjustability. Forecasting is relevant for the maximum and minimum values. Forecasting is not relevant for the present value because it is dispatchable and bounded only by the maximum and minimum.

Forecasts may involve varying degrees of uncertainty. To represent this, the forecast for a given parameter can be described as an envelope, a range of uncertainty, possibly widening further into the future as the forecast becomes less certain. Figure 3-21 illustrates a CIM-based DER group forecast sequence and Figure 3-22 illustrates a MultiSpeak example of a request for DER Group forecasts.

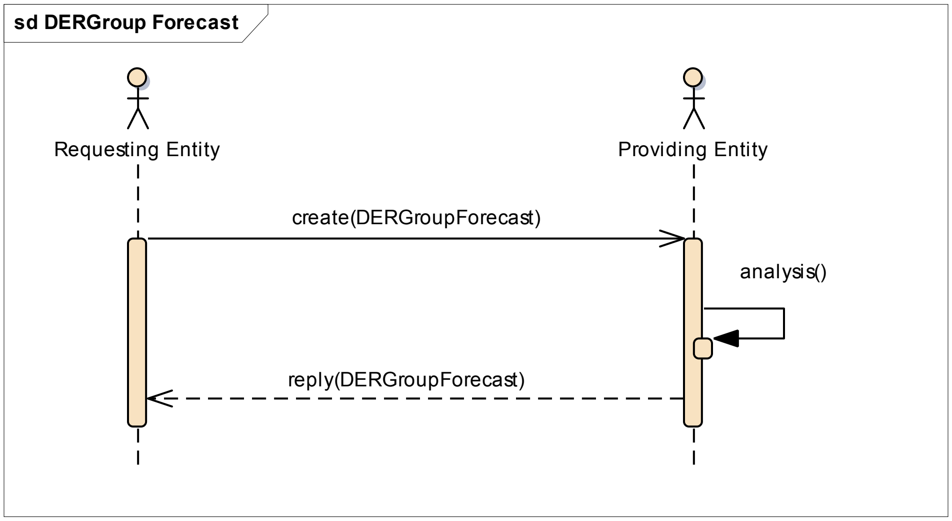


Figure 3-21
CIM-based DER Group Forecast Sequence Diagram

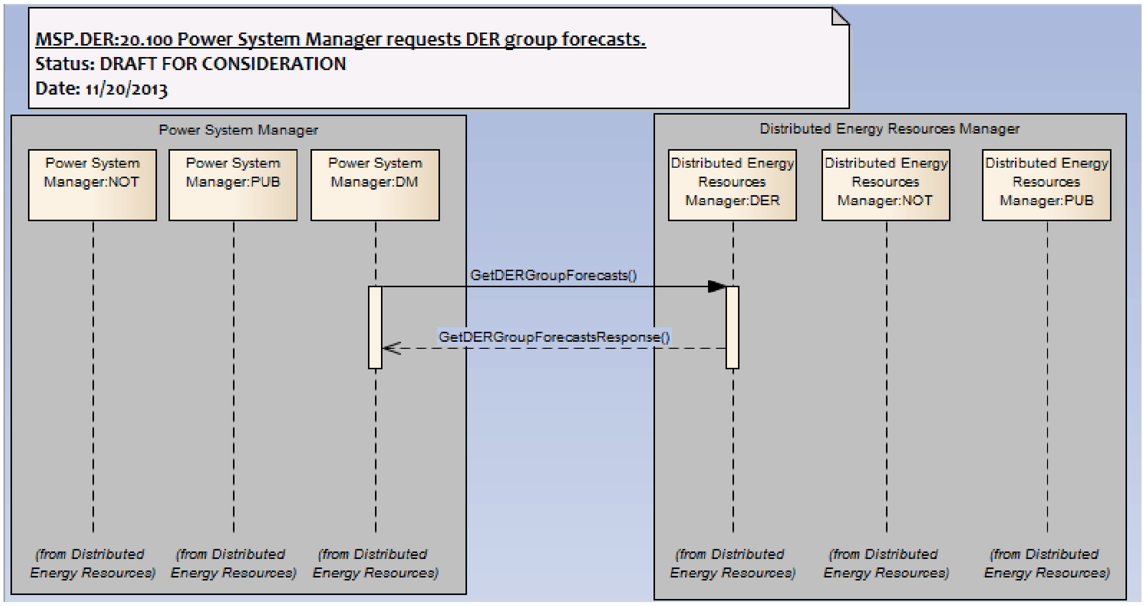

MSP.DER:20.100 Power System Manager requests DER group forecasts.
Status: DRAFT FOR CONSIDERATION
Date: 11/20/2013
Power System Manager
Power System Manager:NOT
Power System Manager:PUB
Power System Manager:DM
Distributed Energy Resources Manager
Distributed Energy Resources Manager:DER
Distributed Energy Resources Manager:NOT
Distributed Energy Resources Manager:PUB
GetDERGroupForecasts()
GetDERGroupForecastsResponse()
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)
(from Distributed Energy Resources)

Figure 3-22
MultiSpeak example of a request for DER Group forecasts

The forecast-requesting entity will pass to the forecast-providing entity:

- The quantity to be forecasted (shown as Maximum Watts Capability in this example)
- A forecast start time
- Interval time
- End time

The data that is returned for each interval of time may take on several forms, according to the design of the forecasting entity. The form will be identified by an enumeration, also included in the returned data, and allowing, at a minimum, for the following options:

- Best guess, midpoint only
- Best guess, plus 90% confidence high/low points
- Normal (Gaussian) distribution. Providing mean and standard deviation.
- Central Chi-square distribution. Providing midpoint and degrees of freedom K.

For illustrated examples of these forecast types see: *Enterprise Integration Functions for Distributed Energy Resources*, Phase 1, Product ID: 3002001088, Technical Update, October 2013, Electric Power Research Institute, Palo Alto, CA.

Pre-Condition(s) A DER Group must exist in the responding system with corresponding members

Post Condition(s): The requesting system will receive the DER forecast, with the degree of confidence indicated, for the capability requested

Table 3-19 shows a MultiSpeak DERGroup forecast and Table 3-20 shows a CIM DERGroup Forecast.

Table 3-19
MultiSpeak DERGroup Forecast

Step	End Point	Verb	Payload Name	Pass / Fail
1	http://54.210.233.171/	Get	DERGroupForecasts	
2	http://54.210.233.171/		DERGroupForecastsResponse	

Required data:

Data Element	Data Type	Notes
DERGroup.objectID	UUID	Identifies the DER Group
DERGroup.primaryIdentifier	String	Set to “NA”

Table 3-20
CIM DERGroup Forecast

Step	End Point	Verb	Payload Name	Pass / Fail
1	http://54.210.233.171/	create	DERGroupForecasts	
2	http://54.210.233.171/	reply	DERGroupForecastsResponse	

Required data:

Data Element	Data Type	Notes
DERGroup.objectID	UUID	Identifies the DER Group
DERGroup.primaryIdentifier	String	Set to “NA”

Example MultiSpeak-based XML - GetDERGroupForecast

This forecast request is being made for 30 days, for RealPower.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<n1:GetDERGroupForecasts xmlns="http://www.multispeak.org/V5.0/commonTypes"
xmlns:msp="http://www.multispeak.org/V5.0"
xmlns:cpsm="http://www.multispeak.org/V5.0/cpsm"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:sbarrays="http://www.multispeak.org/V5.0/sandboxArrays"
xmlns:n1="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server"
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server
SB_DER_Server.xsd">
      <n1:forecastInterval units="Days">3</n1:forecastInterval>
      <n1:forecastPeriod>
            <startTime>2014-01-17T09:30:47Z</startTime>
            <endTime>2014-01-17T09:30:47Z</endTime>
      </n1:forecastPeriod>
      <n1:ArrayOfDERGroup>
```

```
            <sbarrays:DERGroup objectGUID="3832cf79-0c5a-405a-90f8-
f6cc6b37806e" utility="String">
                  <msp:comments>Your comments here</msp:comments>
                  <msp:primaryIdentifier identifierName="String"
identifierLabel="String">NA</msp:primaryIdentifier>
            </sbarrays:DERGroup>
      </n1:ArrayOfDERGroup>
      <n1:ArrayOfRequestedCapability>
            <sbarrays:requestedCapability
otherKind="String">RealPower</sbarrays:requestedCapability>
      </n1:ArrayOfRequestedCapability>
</n1:GetDERGroupForecasts>
```

Example MultiSpeak-based XML - GetDERGroupForecastResponse

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<n1:GetDERGroupForecastsResponse
xmlns="http://www.multispeak.org/V5.0/commonTypes"
xmlns:msp="http://www.multispeak.org/V5.0"
xmlns:cpsm="http://www.multispeak.org/V5.0/cpsm"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:sbarrays="http://www.multispeak.org/V5.0/sandboxArrays"
xmlns:n1="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server"
xsi:schemaLocation="http://www.multispeak.org/V5.0/wsdl/SB_DER_Server
SB_DER_Server.xsd">
      <n1:ArrayOfDERGroupForecast>
            <sbarrays:DERGroupForecast referableID="String">
                  <msp:timeStamp>2001-12-17T09:30:47Z</msp:timeStamp>
                  <forecastDate xmlns="">2001-12-17T09:30:47Z</forecastDate>
                  <forecastInterval xmlns=""
units="Minutes">3.1415901184082031</forecastInterval>
                  <DERGroup xmlns="" objectGUID="3832cf79-0c5a-405a-90f8-
f6cc6b37806e" utility="String">
                        <msp:comments>example comment</msp:comments>
                        <msp:primaryIdentifier identifierName="String"
identifierLabel="String">NA</msp:primaryIdentifier>
                  </DERGroup>
                  <DERGroupPrediction xmlns="">
                        <sequence>0</sequence>
                        <confidence>95</confidence>
                        <DERGroupCapabilities>
                              <realPower
units="PerUnit">3.1415901184082031</realPower>
                              <reactivePower
units="microVAr">3.1415901184082031</reactivePower>
                              <apparentPower
units="PerUnit">3.1415901184082031</apparentPower>
                        </DERGroupCapabilities>
                  </DERGroupPrediction>
                  <DERGroupPrediction xmlns="">
                        <sequence>1</sequence>
                        <confidence>95</confidence>
                        <DERGroupCapabilities>
                              <realPower
units="PerUnit">3.1415901184082031</realPower>
                              <reactivePower
units="microVAr">3.1415901184082031</reactivePower>
                              <apparentPower
```

```
units="PerUnit">3.1415901184082031</apparentPower>
                        </DERGroupCapabilities>
                  </DERGroupPrediction>
                  <DERGroupPrediction xmlns="">
                        <sequence>2</sequence>
                        <confidence>95</confidence>
                        <DERGroupCapabilities>
                              <realPower
units="PerUnit">3.1415901184082031</realPower>
                              <reactivePower
units="microVAr">3.1415901184082031</reactivePower>
                              <apparentPower
units="PerUnit">3.1415901184082031</apparentPower>
                        </DERGroupCapabilities>
                  </DERGroupPrediction>
            </sbarrays:DERGroupForecast>
      </n1:ArrayOfDERGroupForecast>
</n1:GetDERGroupForecastsResponse>
```

Example CIM-based XML - createDERGroupForecast

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<DERGroupForecasts xmlns="http://www.epri.com/2013/DERGroup#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.epri.com/2013/DERGroup#
getDERGroupForecast.xsd">
      <DERGroupForecast>
            <DERGroup>
                  <name>Example DERGroup name</name>
                  <!-- identifier for the DERGroup being requested -->
                  <mRID>b2dd9e07-6062-41b6-b5c8-afe1250beb9e</mRID>
            </DERGroup>
            <TimeInterval>
                  <!-- this request is for 3 days -->
                  <value>3</value>
                  <TimeUnit>DD</TimeUnit>
            </TimeInterval>
            <forecastBegin>2014-12-17T09:30:47Z</forecastBegin>
            <forecastEnd>2014-12-20T09:30:47Z</forecastEnd>
            <RequestedCapability>Watts</RequestedCapability>
            <!-- this is the mRID of the requested information object -->
            <mRID>33b4d3ed-b683-495c-8672-e377c2328e94</mRID>
      </DERGroupForecast>
</DERGroupForecasts>
```

Example CIM-based XML replyDERGroupForecast

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!--Sample XML file generated by XMLSpy v2014 rel. 2 sp1 (x64)
(http://www.altova.com)-->
<DERGroupForecasts xmlns="http://www.epri.com/2013/DERStatus#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.epri.com/2013/DERStatus# DERGroupForecast.xsd">
	<DERGroupForecast>
		<DERGroup>
			<!-- this is the identifier of the DERGroup -->
			<mRID>b2dd9e07-6062-41b6-b5c8-afe1250beb9e</mRID>
		</DERGroup>
		<DERForecastDate>2001-12-17T09:30:47Z</DERForecastDate>
		<DERGroupPrediction>
			<sequence>0</sequence>
			<confidence>95</confidence>
			<CapabilityList>
				<Watts>
					<value>10</value>
					<unit>W</unit>
					<multipler>k</multipler>
				</Watts>
			</CapabilityList>
		</DERGroupPrediction>
		<DERGroupPrediction>
			<sequence>1</sequence>
			<confidence>95</confidence>
			<CapabilityList>
				<Watts>
					<value>9</value>
					<unit>W</unit>
					<multipler>k</multipler>
				</Watts>
			</CapabilityList>
		</DERGroupPrediction>
		<DERGroupPrediction>
			<sequence>2</sequence>
			<confidence>90</confidence>
			<CapabilityList>
				<Watts>
					<value>12</value>
					<unit>W</unit>
					<multipler>k</multipler>
				</Watts>
			</CapabilityList>
		</DERGroupPrediction>
		<TimeInterval>
			<value>3</value>
			<TimeUnit>DD</TimeUnit>
		</TimeInterval>
		<!-- this is when the forecast begins -->
		<PredictionStartDate>2001-12-17T09:30:47Z</PredictionStartDate>
		<!-- this is the mRID of the response object -->
		<mRID>33b4d3ed-b683-495c-8672-e377c2328e94</mRID>
	</DERGroupForecast>
</DERGroupForecasts>
```

A
REFERENCES

While the test harness and test procedures will be updated to accommodate these tests, for an understanding of the prior work that has been accomplished, the EPRI report “Common Information Model (CIM) Conformity and Interoperability Test Procedure Development,” is informative, particularly the appendices that describe the Test Harness setup.

Available
[Online] http://www.epri.com/abstracts/Pages/ProductAbstract.aspx?ProductId=000000000001024448

The CIM-based messages leverage the verb-object naming convention and follow the integration guidance based on IEC 61968-100 Application Integration at electric utilities. This international standard is available from the IEC store.

Available
[Online] http://webstore.iec.ch/Webstore/webstore.nsf/ArtNum_PK/48349!opendocument&preview=1

The MultiSpeak messages leverage the verb-object naming convention and the methods specified in the “sandbox” of the release candidate v5.0.4

Available [Online] www.multispeak.org (for MultiSpeak members)

The original open source code for the test harness is available at SourceForge.

Available [Online] http://usstestharness.sourceforge.net/

The use case patterns that describes how to remove an association between an element and a parent object are described in detail in IEC 61968-9 Meter Reading and Control,” 2nd Edition, Annex L.

Available [Online] http://webstore.iec.ch/webstore/webstore.nsf/artnum/048719!opendocument

B
CIM MESSAGE ADDITIONS/CORRECTIONS

CIM DERGroupDispatch

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSpy v2014 rel. 2 sp1 (x64) (http://www.altova.com) by
Gerald Gray (private) -->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:a="http://langdale.com.au/2005/Message#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl"
xmlns="http://langdale.com.au/2005/Message#"
xmlns:m="http://www.epri.com/2013/DERGroup#"
targetNamespace="http://www.epri.com/2013/DERGroup#"
elementFormDefault="qualified" attributeFormDefault="unqualified">
      <xs:annotation>
            <xs:documentation/>
      </xs:annotation>
      <xs:element name="DERGroupDispatches" type="m:DERGroupDispatches"/>
      <xs:complexType name="DERGroupDispatches">
            <xs:sequence>
                  <xs:element name="DERGroupDispatch"
type="m:DERGroupDispatch" minOccurs="1" maxOccurs="unbounded"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="DERGroupDispatch">
            <xs:sequence>
                  <xs:element name="DERGroup" type="m:DERGroup" minOccurs="1"
maxOccurs="1">
                        <xs:annotation>
                              <xs:documentation>The DERGroup that the forecast
is requested for</xs:documentation>
                        </xs:annotation>
                  </xs:element>
                  <xs:element name="RequestedCapability"
type="m:RequestedCapability" minOccurs="1" maxOccurs="1">
                        <xs:annotation>
                              <xs:documentation>An enumeration for the
capability types, watts, real or aparent power</xs:documentation>
                        </xs:annotation>
                  </xs:element>
                  <xs:element name="extensionsList" type="m:extensionsList"
minOccurs="0"/>
                  <xs:element name="Names" type="m:Name" minOccurs="0"
maxOccurs="unbounded" sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-
schema-cim15#IdentifiedObject.Names">
                        <xs:annotation>
                              <xs:documentation>All names of this identified
object.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
                  <xs:element name="mRID" type="xs:string" minOccurs="0"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#IdentifiedObject.mRID">
                        <xs:annotation>
                              <xs:documentation>A Model Authority issues
mRIDs. Given that each Model Authority has a unique id and this id is part of
the mRID, then the mRID is globally unique.</xs:documentation>
```

```
                              <xs:documentation>Global uniqeness is easily
achived by using a UUID for the mRID. It is strongly recommended to do
this.</xs:documentation>
                              <xs:documentation>For CIMXML data files the mRID
is mapped to rdf:ID or rdf:about attributes that identifies CIM object
elements.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
                  <xs:element name="otherCapability" type="m:otherCapability"
minOccurs="0">
                        <xs:annotation>
                              <xs:documentation>Other capabilities not
originally foreseen for this request</xs:documentation>
                        </xs:annotation>
                  </xs:element>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="otherCapability">
            <xs:sequence>
                  <xs:element name="otherCapabilityType" type="xs:string"
minOccurs="1" maxOccurs="1"/>
                  <xs:element name="otherCapabilitySubType" type="xs:string"
minOccurs="0" maxOccurs="1"/>
                  <xs:element name="otherCapabilityValue" type="xs:float"
minOccurs="0" maxOccurs="1"/>
                  <xs:element name="otherCapabilityUnits" type="xs:string"
minOccurs="0" maxOccurs="1"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="extensionsList">
            <xs:annotation>
                  <xs:documentation>This is an optional, self-describing means
to extend any MultiSpeak object.</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element name="extensionsItem" type="m:extensionsItem"
minOccurs="1" maxOccurs="unbounded"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="extensionsItem">
            <xs:annotation>
                  <xs:documentation>This is an optional means to add self-
describing extensions to any class that inherits from the
mspObject.</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element name="extName" type="xs:string" minOccurs="1"
maxOccurs="1"/>
                  <xs:element name="extValue" type="xs:string" minOccurs="1"
maxOccurs="1"/>
                  <xs:element name="extType" type="xs:string" minOccurs="0"
maxOccurs="1"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="extValue">
            <xs:annotation>
                  <xs:documentation>This is the value of the extensionsItem.
The units of this extensionsItem are included in the Units attrbute on
extValue.</xs:documentation>
            </xs:annotation>
            <xs:simpleContent>
                  <xs:extension base="xs:string">
                        <xs:attribute name="units" type="xs:string"
use="optional">
```

```
                              <xs:annotation>
                                    <xs:documentation>If used, this attribute
carries the units in which the extValue is expressed. </xs:documentation>
                              </xs:annotation>
                        </xs:attribute>
                  </xs:extension>
            </xs:simpleContent>
      </xs:complexType>
      <xs:complexType name="RequestedCapability">
            <xs:sequence>
                  <xs:element name="capabilityType" type="m:capabilityType"
minOccurs="1" maxOccurs="1"/>
                  <xs:element name="value" type="xs:float" minOccurs="1"
maxOccurs="1"/>
                  <xs:element name="capabilityUnits" type="xs:string"
default="VA" minOccurs="1" maxOccurs="1"/>
                  <xs:element name="capabilityMultiplier"
type="m:UnitMultiplier" minOccurs="1" maxOccurs="1"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="DERGroup">
            <xs:annotation>
                  <xs:documentation>DERGroup is a convention for a Distributed
Energy Resource Management System (DERMS) to group DER devices to ease
communication with other utility systems. Groups could be created with any
organizing principle, by device type, by location, by nameplate
characteristics, etc.</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element name="name" type="xs:string" minOccurs="1"
maxOccurs="1"/>
                  <xs:element name="mRID" type="xs:string" minOccurs="0"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#IdentifiedObject.mRID">
                        <xs:annotation>
                              <xs:documentation>A Model Authority issues
mRIDs. Given that each Model Authority has a unique id and this id is part of
the mRID, then the mRID is globally unique.</xs:documentation>
                              <xs:documentation>Global uniqeness is easily
achived by using a UUID for the mRID. It is strongly recommended to do
this.</xs:documentation>
                              <xs:documentation>For CIMXML data files the mRID
is mapped to rdf:ID or rdf:about attributes that identifies CIM object
elements.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
                  <xs:element name="Names" type="m:Name" minOccurs="0"
maxOccurs="unbounded" sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-
schema-cim15#IdentifiedObject.Names">
                        <xs:annotation>
                              <xs:documentation>All names of this identified
object.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="Name"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-cim15#Name">
            <xs:annotation>
                  <xs:documentation>The Name class provides the means to
define any number of human readable  names for an object. A name is
&lt;b&gt;not&lt;/b&gt; to be used for defining inter-object relationships. For
inter-object relationships instead use the object identification
'mRID'.</xs:documentation>
```

```
            </xs:annotation>
            <xs:sequence>
                  <xs:element name="name" type="xs:string" minOccurs="1"
maxOccurs="1" sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#Name.name">
                        <xs:annotation>
                              <xs:documentation>Any free text that name the
object.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
                  <xs:element name="NameType" type="m:NameType" minOccurs="0"
maxOccurs="1" sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#Name.NameType">
                        <xs:annotation>
                              <xs:documentation>Type of this
name.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="NameType"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-cim15#NameType">
            <xs:annotation>
                  <xs:documentation>Type of name. Possible values for
attribute 'name' are implementation dependent but standard profiles may
specify types. An enterprise may have multiple IT systems each having its own
local name for the same object, e.g. a planning system may have different
names from an EMS. An object may also have different names within the same IT
system, e.g. localName and aliasName as defined in CIM version 14. Their
definitions from CIM14 are</xs:documentation>
                  <xs:documentation>The localName is a human readable name of
the object. It is only used with objects organized in a naming hierarchy.
localName: A free text name local to a node in a naming hierarchy similar to a
file directory structure. A power system related naming hierarchy may be:
Substation, VoltageLevel, Equipment etc. Children of the same parent in such a
hierarchy have names that typically are unique among them.</xs:documentation>
                  <xs:documentation>aliasName: A free text alternate name
typically used in tabular reports where the column width is
limited.</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element name="description" type="xs:string"
minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#NameType.description">
                        <xs:annotation>
                              <xs:documentation>Description of the name
type.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
                  <xs:element name="name" type="xs:string" minOccurs="1"
maxOccurs="1" sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#NameType.name">
                        <xs:annotation>
                              <xs:documentation>Name of the name
type.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
                  <xs:element name="NameTypeAuthority"
type="m:NameTypeAuthority" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#NameType.NameTypeAuthority">
                        <xs:annotation>
```

```
                              <xs:documentation>Authority responsible for
managing names of this type.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="NameTypeAuthority"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#NameTypeAuthority">
            <xs:annotation>
                  <xs:documentation>Authority responsible for creation and
management of names of a given type; typically an organization or an
enterprise system.</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element name="description" type="xs:string"
minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#NameTypeAuthority.description">
                        <xs:annotation>
                              <xs:documentation>Description of the name type
authority.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
                  <xs:element name="name" type="xs:string" minOccurs="1"
maxOccurs="1" sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#NameTypeAuthority.name">
                        <xs:annotation>
                              <xs:documentation>Name of the name type
authority.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
            </xs:sequence>
      </xs:complexType>
      <xs:simpleType name="Percentage">
            <xs:restriction base="xs:integer">
                  <xs:minInclusive value="0"/>
                  <xs:maxInclusive value="100"/>
            </xs:restriction>
      </xs:simpleType>
      <xs:simpleType name="capabilityType">
            <xs:restriction base="xs:string">
                  <xs:enumeration value="RealPower"/>
                  <xs:enumeration value="ApparentPower"/>
            </xs:restriction>
      </xs:simpleType>
      <xs:simpleType name="UnitMultiplier"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#UnitMultiplier">
            <xs:annotation>
                  <xs:documentation>The unit multipliers defined for the
CIM.</xs:documentation>
            </xs:annotation>
            <xs:restriction base="xs:string">
                  <xs:enumeration value="G">
                        <xs:annotation>
                              <xs:documentation>Giga 10**9</xs:documentation>
                        </xs:annotation>
                  </xs:enumeration>
                  <xs:enumeration value="M">
                        <xs:annotation>
                              <xs:documentation>Mega 10**6</xs:documentation>
                        </xs:annotation>
                  </xs:enumeration>
                  <xs:enumeration value="T">
                        <xs:annotation>
                              <xs:documentation>Tera 10**12</xs:documentation>
                        </xs:annotation>
                  </xs:enumeration>
                  <xs:enumeration value="c">
                        <xs:annotation>
                              <xs:documentation>Centi 10**-
2</xs:documentation>
                        </xs:annotation>
                  </xs:enumeration>
                  <xs:enumeration value="d">
                        <xs:annotation>
                              <xs:documentation>Deci 10**-1</xs:documentation>
                        </xs:annotation>
                  </xs:enumeration>
                  <xs:enumeration value="k">
                        <xs:annotation>
                              <xs:documentation>Kilo 10**3</xs:documentation>
                        </xs:annotation>
                  </xs:enumeration>
                  <xs:enumeration value="m">
                        <xs:annotation>
                              <xs:documentation>Milli 10**-
3</xs:documentation>
                        </xs:annotation>
                  </xs:enumeration>
                  <xs:enumeration value="micro">
                        <xs:annotation>
                              <xs:documentation>Micro 10**-
6</xs:documentation>
                        </xs:annotation>
                  </xs:enumeration>
                  <xs:enumeration value="n">
                        <xs:annotation>
                              <xs:documentation>Nano 10**-9</xs:documentation>
                        </xs:annotation>
                  </xs:enumeration>
                  <xs:enumeration value="none">
                        <xs:annotation/>
                  </xs:enumeration>
                  <xs:enumeration value="p">
                        <xs:annotation>
                              <xs:documentation>Pico 10**-
12</xs:documentation>
                        </xs:annotation>
                  </xs:enumeration>
            </xs:restriction>
      </xs:simpleType>
</xs:schema>
```

CIM DERGroupStatus

The original XSD was missing the core status class from the CIM. While the current capabilities present useful information, the status class is how actual status, verbiage that represents anything one system will tell another, is passed. Figure B-1 shows a schema view of the update to DERGroupStatus.

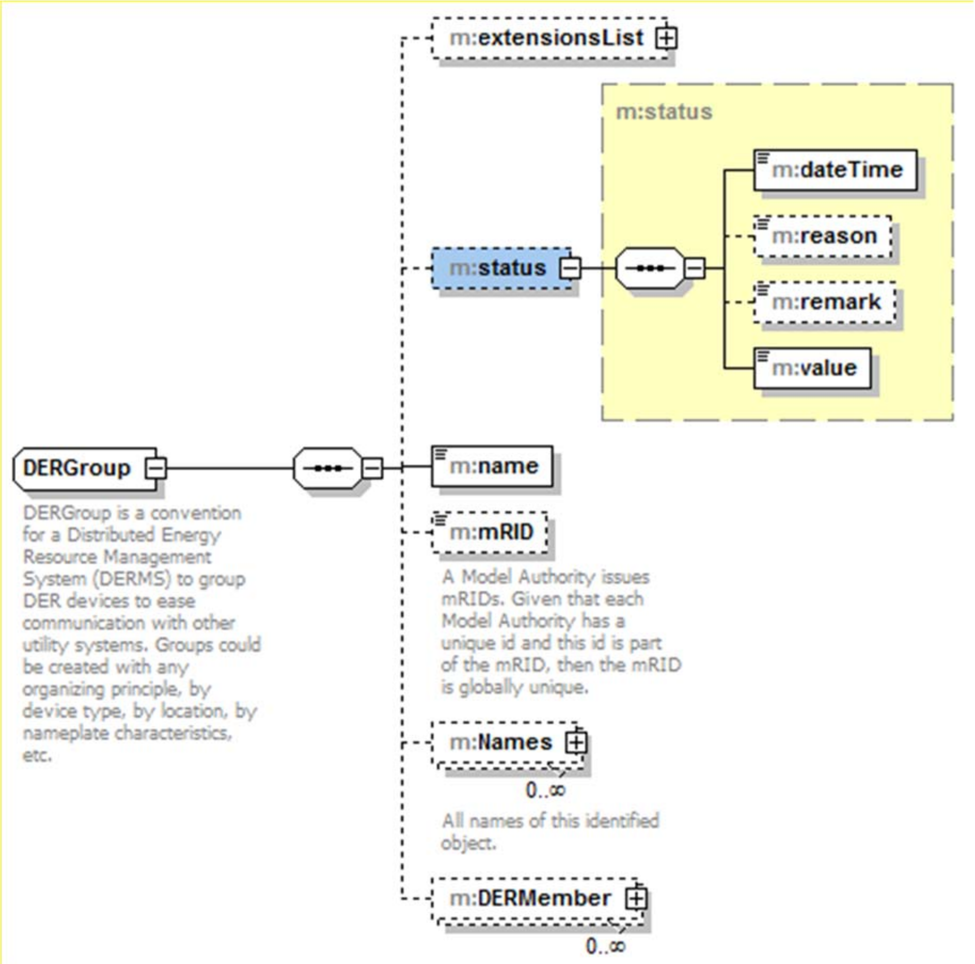

m:extensionsList
m:status
m:dateTime
m:reason
m:remark
m:value
m:status
DERGroup
DERGroup is a convention for a Distributed Energy Resource Management System (DERMS) to group DER devices to ease communication with other utility systems. Groups could be created with any organizing principle, by device type, by location, by nameplate characteristics, etc.
m:name
m:mRID
A Model Authority issues mRIDs. Given that each Model Authority has a unique id and this id is part of the mRID, then the mRID is globally unique.
m:Names
0..∞
All names of this identified object.
m:DERMember
0..∞

Figure B-1
Schema view of the update to DERGroupStatus

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSpy v2012 rel. 2 sp1 (x64) (http://www.altova.com) by
Gerald Gray (private) -->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:a="http://langdale.com.au/2005/Message#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl"
xmlns="http://langdale.com.au/2005/Message#"
xmlns:m="http://www.epri.com/2013/DERGroup#"
targetNamespace="http://www.epri.com/2013/DERGroup#"
elementFormDefault="qualified" attributeFormDefault="unqualified">
      <xs:annotation>
            <xs:documentation/>
      </xs:annotation>
      <xs:element name="DERGroups" type="m:DERGroups"/>
      <xs:complexType name="DERGroups">
            <xs:sequence>
```

```
                  <xs:element name="DERGroup" minOccurs="1"
maxOccurs="unbounded">
                        <xs:complexType>
                              <xs:complexContent>
                                    <xs:extension base="m:DERGroup">
                                          <xs:attribute name="description"/>
                                          <xs:attribute name="comment"/>
                                    </xs:extension>
                              </xs:complexContent>
                        </xs:complexType>
                  </xs:element>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="DERGroup">
            <xs:annotation>
                  <xs:documentation>DERGroup is a convention for a Distributed
Energy Resource Management System (DERMS) to group DER devices to ease
communication with other utility systems. Groups could be created with any
organizing principle, by device type, by location, by nameplate
characteristics, etc.</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element name="extensionsList" type="m:extensionsList"
minOccurs="0"/>
                  <xs:element name="status" type="m:status" minOccurs="0"
maxOccurs="1"/>
                  <xs:element name="name" type="xs:string" minOccurs="1"
maxOccurs="1"/>
                  <xs:element name="mRID" type="xs:string" minOccurs="0"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#IdentifiedObject.mRID">
                        <xs:annotation>
                              <xs:documentation>A Model Authority issues
mRIDs. Given that each Model Authority has a unique id and this id is part of
the mRID, then the mRID is globally unique.</xs:documentation>
                              <xs:documentation>Global uniqeness is easily
achived by using a UUID for the mRID. It is strongly recommended to do
this.</xs:documentation>
                              <xs:documentation>For CIMXML data files the mRID
is mapped to rdf:ID or rdf:about attributes that identifies CIM object
elements.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
                  <xs:element name="Names" type="m:Name" minOccurs="0"
maxOccurs="unbounded" sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-
schema-cim15#IdentifiedObject.Names">
                        <xs:annotation>
                              <xs:documentation>All names of this identified
object.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
                  <xs:element name="DERMember" type="m:DERMember"
minOccurs="0" maxOccurs="unbounded"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="Name"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-cim15#Name">
            <xs:annotation>
                  <xs:documentation>The Name class provides the means to
define any number of human readable  names for an object. A name is
&lt;b&gt;not&lt;/b&gt; to be used for defining inter-object relationships. For
inter-object relationships instead use the object identification
'mRID'.</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element name="name" type="xs:string" minOccurs="1"
maxOccurs="1" sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#Name.name">
                        <xs:annotation>
                              <xs:documentation>Any free text that name the
object.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
                  <xs:element name="NameType" type="m:NameType" minOccurs="0"
maxOccurs="1" sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#Name.NameType">
                        <xs:annotation>
                              <xs:documentation>Type of this
name.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="NameType"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-cim15#NameType">
            <xs:annotation>
                  <xs:documentation>Type of name. Possible values for
attribute 'name' are implementation dependent but standard profiles may
specify types. An enterprise may have multiple IT systems each having its own
local name for the same object, e.g. a planning system may have different
names from an EMS. An object may also have different names within the same IT
system, e.g. localName and aliasName as defined in CIM version 14. Their
definitions from CIM14 are</xs:documentation>
                  <xs:documentation>The localName is a human readable name of
the object. It is only used with objects organized in a naming hierarchy.
localName: A free text name local to a node in a naming hierarchy similar to a
file directory structure. A power system related naming hierarchy may be:
Substation, VoltageLevel, Equipment etc. Children of the same parent in such a
hierarchy have names that typically are unique among them.</xs:documentation>
                  <xs:documentation>aliasName: A free text alternate name
typically used in tabular reports where the column width is
limited.</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element name="description" type="xs:string"
minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#NameType.description">
                        <xs:annotation>
                              <xs:documentation>Description of the name
type.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
                  <xs:element name="name" type="xs:string" minOccurs="1"
maxOccurs="1" sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#NameType.name">
                        <xs:annotation>
                              <xs:documentation>Name of the name
type.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
                  <xs:element name="NameTypeAuthority"
type="m:NameTypeAuthority" minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#NameType.NameTypeAuthority">
                        <xs:annotation>
                              <xs:documentation>Authority responsible for
managing names of this type.</xs:documentation>
```

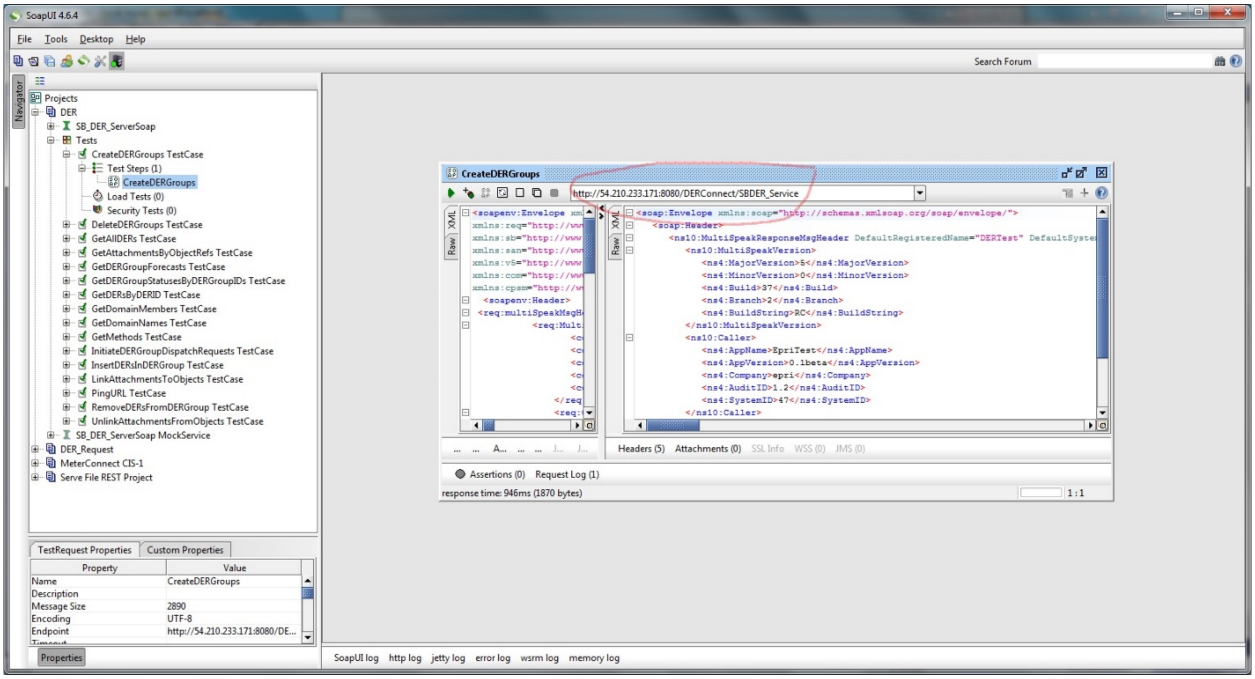
```
                        </xs:annotation>
                  </xs:element>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="NameTypeAuthority"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#NameTypeAuthority">
            <xs:annotation>
                  <xs:documentation>Authority responsible for creation and
management of names of a given type; typically an organization or an
enterprise system.</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element name="description" type="xs:string"
minOccurs="0" maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#NameTypeAuthority.description">
                        <xs:annotation>
                              <xs:documentation>Description of the name type
authority.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
                  <xs:element name="name" type="xs:string" minOccurs="1"
maxOccurs="1" sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#NameTypeAuthority.name">
                        <xs:annotation>
                              <xs:documentation>Name of the name type
authority.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="DERMember">
            <xs:sequence>
                  <xs:element name="mRID" type="xs:string" minOccurs="0"
sawsdl:modelReference="http://iec.ch/TC57/2010/CIM-schema-
cim15#IdentifiedObject.mRID">
                        <xs:annotation>
                              <xs:documentation>A Model Authority issues
mRIDs. Given that each Model Authority has a unique id and this id is part of
the mRID, then the mRID is globally unique.</xs:documentation>
                              <xs:documentation>Global uniqeness is easily
achived by using a UUID for the mRID. It is strongly recommended to do
this.</xs:documentation>
                              <xs:documentation>For CIMXML data files the mRID
is mapped to rdf:ID or rdf:about attributes that identifies CIM object
elements.</xs:documentation>
                        </xs:annotation>
                  </xs:element>
                  <xs:element name="name" type="xs:string" minOccurs="0"/>
                  <xs:element name="Names" type="m:Name" minOccurs="0"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="extensionsList">
            <xs:annotation>
                  <xs:documentation>This is an optional, self-describing means
to extend any MultiSpeak object.</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element name="extensionsItem" type="m:extensionsItem"
minOccurs="1" maxOccurs="unbounded"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="extensionsItem">
```

```
            <xs:annotation>
                  <xs:documentation>This is an optional means to add self-
describing extensions to any class that inherits from the
mspObject.</xs:documentation>
            </xs:annotation>
            <xs:sequence>
                  <xs:element name="extName" type="xs:string" minOccurs="1"
maxOccurs="1"/>
                  <xs:element name="extValue" type="xs:string" minOccurs="1"
maxOccurs="1"/>
                  <xs:element name="extType" type="xs:string" minOccurs="0"
maxOccurs="1"/>
            </xs:sequence>
      </xs:complexType>
      <xs:complexType name="extValue">
            <xs:annotation>
                  <xs:documentation>This is the value of the extensionsItem.
The units of this extensionsItem are included in the Units attrbute on
extValue.</xs:documentation>
            </xs:annotation>
            <xs:simpleContent>
                  <xs:extension base="xs:string">
                        <xs:attribute name="units" type="xs:string"
use="optional">
                              <xs:annotation>
                                    <xs:documentation>If used, this attribute
carries the units in which the extValue is expressed. </xs:documentation>
                              </xs:annotation>
                        </xs:attribute>
                  </xs:extension>
            </xs:simpleContent>
      </xs:complexType>
      <xs:complexType name="status"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status">
            <xs:sequence>
                  <xs:element name="dateTime" type="xs:dateTime" minOccurs="1"
maxOccurs="1"
sawsdl:modelReference="http://iec.ch/TC57/2009#Status.dateTime"/>
                  <xs:element name="reason" type="xs:string" minOccurs="0"
maxOccurs="1" sawsdl:modelReference="http://iec.ch/TC57/2009#Status.reason"/>
                  <xs:element name="remark" type="xs:string" minOccurs="0"
maxOccurs="1" sawsdl:modelReference="http://iec.ch/TC57/2009#Status.remark"/>
                  <xs:element name="value" type="xs:string" minOccurs="1"
maxOccurs="1" sawsdl:modelReference="http://iec.ch/TC57/2009#Status.value"/>
            </xs:sequence>
      </xs:complexType>
</xs:schema>
```

C

USING THE EPRI SEMANTIC TEST HARNESS

In soupUI, to change the URL for control type messages – simply bring up each test case by double clicking, then click the pulldown bar for the url as illustrated in Figure C-1.



http://54.210.233.171:8080/DERConnect/SBDER_Service

```
<soapenv:Envelope xm
xmlns:req="http://ww
xmlns:sb="http://www
xmlns:san="http://ww
xmlns:v5="http://www
xmlns:com="http://ww
xmlns:cpsm="http://w
  <soapenv:Header>
 <req:multiSpeakMsgH
          <req:Mult
                 <c
                 <c
                 <c
                 <c
                 <c
              </req
              <req:
```

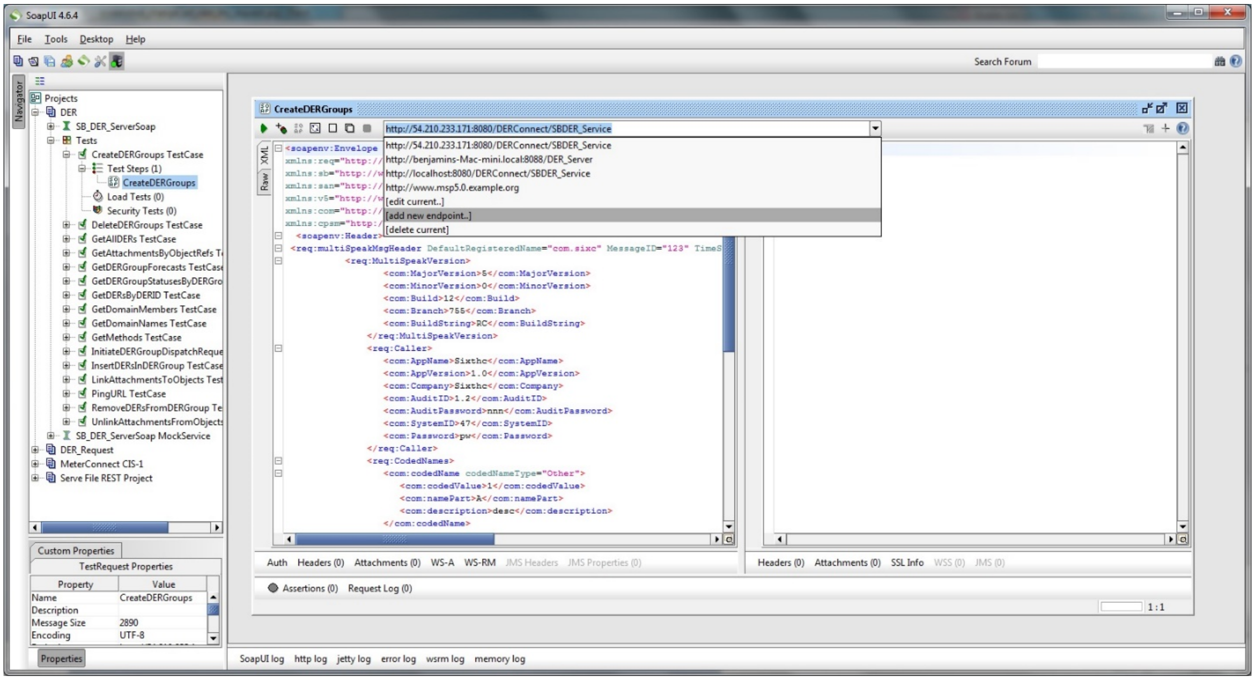
```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
   <soap:Header>
      <ns10:MultiSpeakResponseMsgHeader DefaultRegisteredName="DERTest" DefaultSyste
         <ns10:MultiSpeakVersion>
            <ns4:MajorVersion>5</ns4:MajorVersion>
            <ns4:MinorVersion>0</ns4:MinorVersion>
            <ns4:Build>37</ns4:Build>
            <ns4:Branch>2</ns4:Branch>
            <ns4:BuildString>RC</ns4:BuildString>
         </ns10:MultiSpeakVersion>
         <ns10:Caller>
            <ns4:AppName>EpriTest</ns4:AppName>
            <ns4:AppVersion>0.1beta</ns4:AppVersion>
            <ns4:Company>epri</ns4:Company>
            <ns4:AuditID>1.2</ns4:AuditID>
            <ns4:SystemID>47</ns4:SystemID>
         </ns10:Caller>
```

response time: 946ms (1870 bytes)

Property	Value
Name	CreateDERGroups
Description	
Message Size	2890
Encoding	UTF-8
Endpoint	http://54.210.233.171:8080/DE...

Figure C-1
soupUI screenshot with the URL example

Then click the “*add new endpoint,”* change the IP/hostname to the EPRI EC2 instance, which is 54.210.233.171 (in these screenshots) as shown in Figure C-2.


SoapUI 4.6.4
File Tools Desktop Help
Search Forum
Navigator
Projects
DER
SB_DER_ServerSoap
Tests
CreateDERGroups TestCase
Test Steps (1)
CreateDERGroups
Load Tests (0)
Security Tests (0)
DeleteDERGroups TestCase
GetAllDERs TestCase
GetAttachmentsByObjectRefs T
GetDERGroupForecasts TestCase
GetDERGroupStatusesByDERGro
GetDERsByDERID TestCase
GetDomainMembers TestCase
GetDomainNames TestCase
GetMethods TestCase
InitiateDERGroupDispatchReque
InsertDERsInDERGroup TestCase
LinkAttachmentsToObjects Test
PingURL TestCase
RemoveDERsFromDERGroup Te
UnlinkAttachmentsFromObjects
SB_DER_ServerSoap MockService
DER_Request
MeterConnect CIS-1
Serve File REST Project
Custom Properties
TestRequest Properties
Properties
CreateDERGroups
XML
Raw
Auth Headers (0) Attachments (0) WS-A WS-RM JMS Headers JMS Properties (0)
Headers (0) Attachments (0) SSL Info WSS (0) JMS (0)
Assertions (0) Request Log (0)
1 : 1
SoapUI log http log jetty log error log wsrm log memory log

http://54.210.233.171:8080/DERConnect/SBDER_Service

- http://54.210.233.171:8080/DERConnect/SBDER_Service
- http://benjamins-Mac-mini.local:8088/DER_Server
- http://localhost:8080/DERConnect/SBDER_Service
- http://www.msp5.0.example.org
- [edit current..]
- [add new endpoint..]
- [delete current]

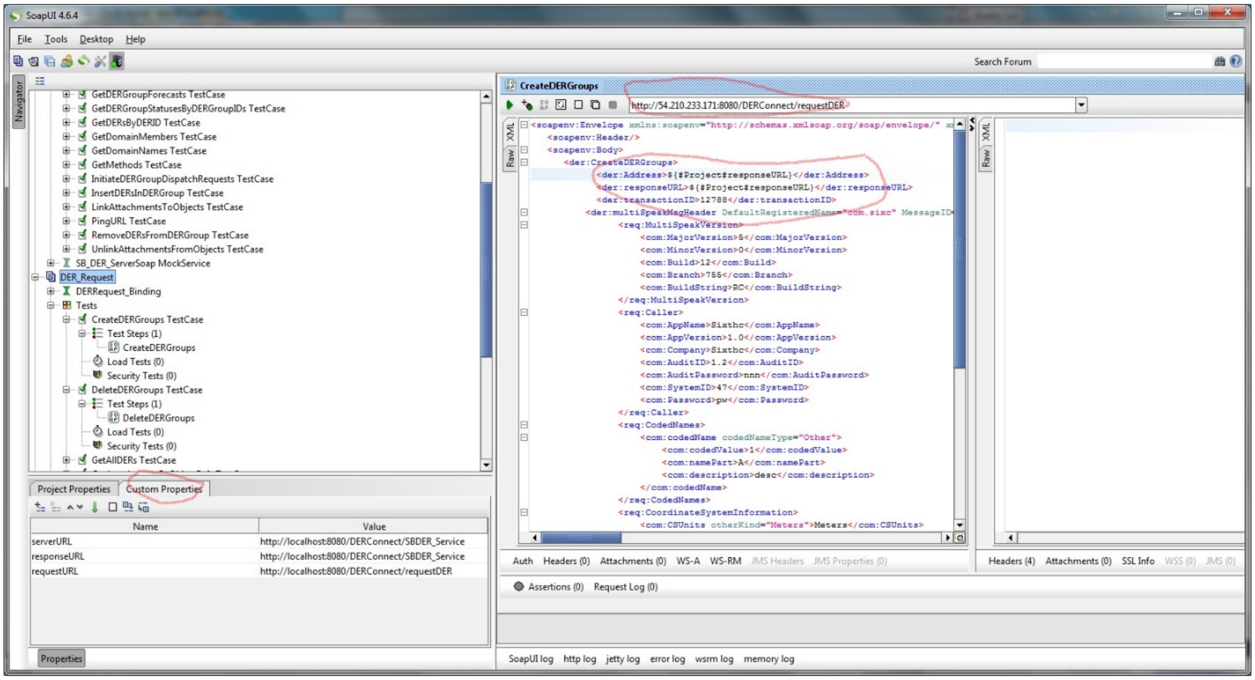
```
<soapenv:Envelope
xmlns:req="http://
xmlns:sb="http://w
xmlns:san="http://
xmlns:v5="http://w
xmlns:com="http://
xmlns:cpsm="http:/
  <soapenv:Header>
 <req:multiSpeakMsgHeader DefaultRegisteredName="com.sixc" MessageID="123" TimeS
          <req:MultiSpeakVersion>
               <com:MajorVersion>5</com:MajorVersion>
               <com:MinorVersion>0</com:MinorVersion>
               <com:Build>12</com:Build>
               <com:Branch>755</com:Branch>
               <com:BuildString>RC</com:BuildString>
            </req:MultiSpeakVersion>
            <req:Caller>
               <com:AppName>Sixthc</com:AppName>
               <com:AppVersion>1.0</com:AppVersion>
               <com:Company>Sixthc</com:Company>
               <com:AuditID>1.2</com:AuditID>
               <com:AuditPassword>nnn</com:AuditPassword>
               <com:SystemID>47</com:SystemID>
               <com:Password>pw</com:Password>
            </req:Caller>
            <req:CodedNames>
               <com:codedName codedNameType="Other">
                  <com:codedValue>1</com:codedValue>
                  <com:namePart>A</com:namePart>
                  <com:description>desc</com:description>
               </com:codedName>
```

Property	Value
Name	CreateDERGroups
Description	
Message Size	2890
Encoding	UTF-8

Figure C-2
soupUI screenshot for the request type example

For request type messages, edit the URL as above, but also supply the response URL. The response URL is the endpoint that the test harness will use when sending the test message. It may be pointed back to the server for a sort of loopback test, but each customer will be pointing that at their own receiving server/code.

Using soapUI variables will make the exchanges easier to manage. Clicking on the high level project (DER Request in the screenshot below), then clicking on custom properties will bring up the variable list as illustrated in Figure C-3. Editing these should change settings for all the test cases. The user can still change them in the individual messages by direct edit, however.



```
GetDERGroupForecasts TestCase
GetDERGroupStatusesByDERGroupIDs TestCase
GetDERsByDERID TestCase
GetDomainMembers TestCase
GetDomainNames TestCase
GetMethods TestCase
InitiateDERGroupDispatchRequests TestCase
InsertDERsInDERGroup TestCase
LinkAttachmentsToObjects TestCase
PingURL TestCase
RemoveDERsFromDERGroup TestCase
UnlinkAttachmentsFromObjects TestCase
SB_DER_ServerSoap MockService
DER_Request
  DERRequest_Binding
  Tests
    CreateDERGroups TestCase
      Test Steps (1)
        CreateDERGroups
      Load Tests (0)
      Security Tests (0)
    DeleteDERGroups TestCase
      Test Steps (1)
        DeleteDERGroups
      Load Tests (0)
      Security Tests (0)
    GetAllDERs TestCase
```

Project Properties | Custom Properties

Name	Value
serverURL	http://localhost:8080/DERConnect/SBDER_Service
responseURL	http://localhost:8080/DERConnect/SBDER_Service
requestURL	http://localhost:8080/DERConnect/requestDER

CreateDERGroups

http://54.210.233.171:8080/DERConnect/requestDER

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" x
   <soapenv:Header/>
   <soapenv:Body>
      <der:CreateDERGroups>
         <der:Address>${#Project#responseURL}</der:Address>
         <der:responseURL>${#Project#responseURL}</der:responseURL>
         <der:transactionID>12788</der:transactionID>
      <der:multiSpeakMsgHeader DefaultRegisteredName="com.sixc" MessageID=
         <req:MultiSpeakVersion>
            <com:MajorVersion>5</com:MajorVersion>
            <com:MinorVersion>0</com:MinorVersion>
            <com:Build>12</com:Build>
            <com:Branch>755</com:Branch>
            <com:BuildString>RC</com:BuildString>
         </req:MultiSpeakVersion>
         <req:Caller>
            <com:AppName>Sixthc</com:AppName>
            <com:AppVersion>1.0</com:AppVersion>
            <com:Company>Sixthc</com:Company>
            <com:AuditID>1.2</com:AuditID>
            <com:AuditPassword>nnn</com:AuditPassword>
            <com:SystemID>47</com:SystemID>
            <com:Password>pw</com:Password>
         </req:Caller>
         <req:CodedNames>
            <com:codedName codedNameType="Other">
               <com:codedValue>1</com:codedValue>
               <com:namePart>A</com:namePart>
               <com:description>desc</com:description>
            </com:codedName>
         </req:CodedNames>
         <req:CoordinateSystemInformation>
            <com:CSUnits otherKind="Meters">Meters</com:CSUnits>
```

Figure C-3
soupUI example illustrating how to change variables

D
EXAMPLES WSDLS (CIM AND MULTISPEAK)

These WSDLs are included as examples for the CIM-based tests and reflect guidance from IEC 61968-100. Only a few examples are included to provide the developer with an idea what a finished WSDL should look like. There is no intent in this test plan to make an exhaustive list.

CIM createDERGroup WSDL

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSpy v2014 rel. 2 sp1 (http://www.altova.com) by Ben
Goodwin (person) -->
<!-- IEC 61968 WSDL for Generic, Type-Independent Web Services -->
<!-- Uses document wrapped WSDL style -->
<wsdl:definitions xmlns:ns="http://iec.ch/TC57/2011/abstract"
xmlns:ns2="http://www.iec.ch/TC57/2011/schema/message"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:ns1="http://www.epri.com/2013/DERGroup#"
targetNamespace="http://iec.ch/TC57/2011/abstract">
      <wsdl:import namespace="http://www.epri.com/2013/DERGroup#"
location="DERGroup.xsd"/>
      <wsdl:message name="createDERGroupSoapIn">
            <wsdl:part name="Message" element="ns1:DERGroups"/>
      </wsdl:message>
      <wsdl:message name="createDERGroupSoapOut">
            <wsdl:part name="Message" element="ns1:DERGroups"/>
      </wsdl:message>
      <wsdl:portType name="Operations">
            <wsdl:operation name="createDERGroup">
                  <wsdl:input message="ns:createDERGroupSoapIn"/>
                  <wsdl:output message="ns:createDERGroupSoapOut"/>
            </wsdl:operation>
      </wsdl:portType>
      <wsdl:binding name="createDERGroup Binding" type="ns:Operations">
            <soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
            <!-- Operation for publication of events -->
            <!-- Operation for request/reply interactions -->
            <wsdl:operation name="createDERGroup">
                  <soap:operation soapAction="http://iec.ch/61968/Request"/>
                  <wsdl:input>
                        <soap:body use="literal"/>
                  </wsdl:input>
                  <wsdl:output>
                        <soap:body use="literal"/>
                  </wsdl:output>
            </wsdl:operation>
            <!-- Operation for asynchronous responses -->
      </wsdl:binding>
      <wsdl:service name="createDERGroup Port">
            <wsdl:port name="SOAP" binding="ns:createDERGroup Binding">
                  <soap:address location="http://iec.ch/61968/"/>
            </wsdl:port>
      </wsdl:service>
</wsdl:definitions>
```

CIM getDERGroup WSDL

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSpy v2014 rel. 2 sp1 (http://www.altova.com) by Ben
Goodwin (person) -->
<!-- IEC 61968 WSDL for Generic, Type-Independent Web Services -->
<!-- Uses document wrapped WSDL style -->
<!-- VERB = GET -->
<wsdl:definitions xmlns:ns="http://iec.ch/TC57/2011/abstract"
xmlns:ns2="http://www.iec.ch/TC57/2011/schema/message"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:ns1="http://www.epri.com/2013/DERGroup#"
targetNamespace="http://iec.ch/TC57/2011/abstract">
      <wsdl:import namespace="http://www.epri.com/2013/DERGroup#"
location="DERGroup.xsd"/>
      <wsdl:message name="getDERGroupsRequest">
            <wsdl:part name="Message" element="ns1:DERGroups"/>
      </wsdl:message>
      <wsdl:message name="getDERGroupsResponse">
            <wsdl:part name="Message" element="ns1:DERGroups"/>
      </wsdl:message>
      <wsdl:portType name="Operations">
            <wsdl:operation name="getDERGroups">
                  <wsdl:documentation>Returns list of DERS found in a DER
Group</wsdl:documentation>
                  <wsdl:input message="ns:getDERGroupsRequest"/>
                  <wsdl:output message="ns:getDERGroupsResponse"/>
            </wsdl:operation>
      </wsdl:portType>
      <wsdl:binding name="getDERGroups Binding" type="ns:Operations">
            <soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
            <!-- Operation for publication of events -->
            <!-- Operation for request/reply interactions -->
            <wsdl:operation name="getDERGroups">
                  <soap:operation soapAction="http://iec.ch/61968/Request"/>
                  <wsdl:input>
                        <soap:body use="literal"/>
                  </wsdl:input>
                  <wsdl:output>
                        <soap:body use="literal"/>
                  </wsdl:output>
            </wsdl:operation>
            <!-- Operation for asynchronous responses -->
      </wsdl:binding>
      <wsdl:service name="getDERGroups Port">
            <wsdl:port name="SOAP" binding="ns:GetDERGroups Binding">
                  <soap:address location="http://iec.ch/61968/"/>
            </wsdl:port>
      </wsdl:service>
</wsdl:definitions>
```

CIM getDERGroupStatus

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSpy v2014 rel. 2 sp1 (http://www.altova.com) by Ben
Goodwin (person) -->
<!-- IEC 61968 WSDL for Generic, Type-Independent Web Services -->
<!-- Uses document wrapped WSDL style -->
<!-- VERB = GET -->
<wsdl:definitions xmlns:ns="http://iec.ch/TC57/2011/abstract"
xmlns:ns2="http://www.iec.ch/TC57/2011/schema/message"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:ns1="http://www.epri.com/2013/DERStatus#"
targetNamespace="http://iec.ch/TC57/2011/abstract">
      <wsdl:import namespace="http://www.epri.com/2013/DERStatus#"
location="DERGroupStatus.xsd"/>
      <wsdl:message name="getDERStatusRequest">
            <wsdl:part name="Message" element="ns1:DERGroupStatuses"/>
      </wsdl:message>
      <wsdl:message name="getDERStatusResponse">
            <wsdl:part name="Message" element="ns1:DERGroupStatuses"/>
      </wsdl:message>
      <wsdl:portType name="Operations">
            <wsdl:operation name="getDERStatus">
                  <wsdl:documentation>Gets DER Status</wsdl:documentation>
                  <wsdl:input message="ns:getDERStatusRequest"/>
                  <wsdl:output message="ns:getDERStatusResponse"/>
            </wsdl:operation>
      </wsdl:portType>
      <wsdl:binding name="getDERStatus Binding" type="ns:Operations">
            <soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
            <!-- Operation for publication of events -->
            <!-- Operation for request/reply interactions -->
            <wsdl:operation name="getDERStatus">
                  <soap:operation soapAction="http://iec.ch/61968/Request"/>
                  <wsdl:input>
                        <soap:body use="literal"/>
                  </wsdl:input>
                  <wsdl:output>
                        <soap:body use="literal"/>
                  </wsdl:output>
            </wsdl:operation>
            <!-- Operation for asynchronous responses -->
      </wsdl:binding>
      <wsdl:service name="getDERStatus Port">
            <wsdl:port name="SOAP" binding="ns:getDERStatus Binding">
                  <soap:address location="http://iec.ch/61968/"/>
            </wsdl:port>
      </wsdl:service>
</wsdl:definitions>
```

CIM getDERGroupCapabilities

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSpy v2014 rel. 2 sp1 (http://www.altova.com) by Ben
Goodwin (person) -->
<!-- IEC 61968 WSDL for Generic, Type-Independent Web Services -->
<!-- Uses document wrapped WSDL style -->
<!-- VERB = GET -->
<wsdl:definitions xmlns:ns="http://iec.ch/TC57/2011/abstract"
xmlns:ns2="http://www.iec.ch/TC57/2011/schema/message"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:ns1="http://www.epri.com/2013/DERStatus#"
targetNamespace="http://iec.ch/TC57/2011/abstract">
      <wsdl:import namespace="http://www.epri.com/2013/DERStatus#"
location="DERGroupStatus.xsd"/>
      <wsdl:message name="getDERCapabilitiesRequest">
            <wsdl:part name="Message" element="ns1:DERGroupStatuses"/>
      </wsdl:message>
      <wsdl:message name="getDERCapabilitiesResponse">
            <wsdl:part name="Message" element="ns1:DERGroupStatuses"/>
      </wsdl:message>
      <wsdl:portType name="Operations">
            <wsdl:operation name="getDERCapabilities">
                  <wsdl:documentation>Returns list of DERS found in a DER
Group</wsdl:documentation>
                  <wsdl:input message="ns:getDERCapabilitiesRequest"/>
                  <wsdl:output message="ns:getDERCapabilitiesResponse"/>
            </wsdl:operation>
      </wsdl:portType>
      <wsdl:binding name="getDERCapabilities_Binding" type="ns:Operations">
            <soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
            <!-- Operation for publication of events -->
            <!-- Operation for request/reply interactions -->
            <wsdl:operation name="getDERCapabilities">
                  <soap:operation soapAction="http://iec.ch/61968/Request"/>
                  <wsdl:input>
                        <soap:body use="literal"/>
                  </wsdl:input>
                  <wsdl:output>
                        <soap:body use="literal"/>
                  </wsdl:output>
            </wsdl:operation>
            <!-- Operation for asynchronous responses -->
      </wsdl:binding>
      <wsdl:service name="getDERCapabilities Port">
            <wsdl:port name="SOAP" binding="ns:getDERCapabilities_Binding">
                  <soap:address location="http://iec.ch/61968/"/>
            </wsdl:port>
      </wsdl:service>
</wsdl:definitions>
```

Export Control Restrictions

The Electric Power Research Institute, Inc. (EPRI, www.epri.com) conducts research and development relating to the generation, delivery and use of electricity for the benefit of the public. An independent, nonprofit organization, EPRI brings together its scientists and engineers as well as experts from academia and industry to help address challenges in electricity, including reliability, efficiency, affordability, health, safety and the environment. EPRI also provides technology, policy and economic analyses to drive long-range research and development planning, and supports research in emerging technologies. EPRI's members represent approximately 90 percent of the electricity generated and delivered in the United States, and international participation extends to more than 30 countries. EPRI's principal offices and laboratories are located in Palo Alto, Calif.; Charlotte, N.C.; Knoxville, Tenn.; and Lenox, Mass.

Together…Shaping the Future of Electricity

3002004681

Electric Power Research Institute
3420 Hillview Avenue, Palo Alto, California 94304-1338 • PO Box 10412, Palo Alto, California 94303-0813 • USA
800.313.3774 • 650.855.2121 • askepri@epri.com • www.epri.com