

# [MS-PPSDECO]:

## PerformancePoint Services Decomposition Tree Protocol

---

### Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft [Open Specification Promise](#) or the [Community Promise](#). If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting [iplg@microsoft.com](mailto:iplg@microsoft.com).
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit [www.microsoft.com/trademarks](http://www.microsoft.com/trademarks).
- **Fictitious Names.** The example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

**Reservation of Rights.** All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

**Tools.** The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

**Preliminary Documentation.** This Open Specification provides documentation for past and current releases and/or for the pre-release version of this technology. This Open Specification is final documentation for past or current releases as specifically noted in the document, as applicable; it is preliminary documentation for the pre-release versions. Microsoft will release final documentation in connection with the commercial release of the updated or new version of this technology. As the documentation may change between this preliminary version and the final version of this technology, there are risks in relying on preliminary documentation. To the extent that you incur additional

development obligations or any other costs as a result of relying on this preliminary documentation, you do so at your own risk.

Preliminary

## Revision Summary

Date	Revision History	Revision Class	Comments
7/13/2009	0.1	Major	Initial Availability
8/28/2009	0.2	Editorial	Revised and edited the technical content
11/6/2009	0.3	Editorial	Revised and edited the technical content
2/19/2010	1.0	Editorial	Revised and edited the technical content
3/31/2010	1.01	Editorial	Revised and edited the technical content
4/30/2010	1.02	Editorial	Revised and edited the technical content
6/7/2010	1.03	Editorial	Revised and edited the technical content
6/29/2010	1.04	Editorial	Changed language and formatting in the technical content.
7/23/2010	1.04	No Change	No changes to the meaning, language, or formatting of the technical content.
9/27/2010	1.04	No Change	No changes to the meaning, language, or formatting of the technical content.
11/15/2010	1.04	No Change	No changes to the meaning, language, or formatting of the technical content.
12/17/2010	1.04	No Change	No changes to the meaning, language, or formatting of the technical content.
3/18/2011	1.04	No Change	No changes to the meaning, language, or formatting of the technical content.
6/10/2011	1.04	No Change	No changes to the meaning, language, or formatting of the technical content.
1/20/2012	1.5	Minor	Clarified the meaning of the technical content.
4/11/2012	1.5	No Change	No changes to the meaning, language, or formatting of the technical content.
7/16/2012	1.5	No Change	No changes to the meaning, language, or formatting of the technical content.
9/12/2012	1.5	No Change	No changes to the meaning, language, or formatting of the technical content.
10/8/2012	1.5.1	Editorial	Changed language and formatting in the technical content.
2/11/2013	1.5.1	No Change	No changes to the meaning, language, or formatting of the technical content.
7/30/2013	1.5.1	No Change	No changes to the meaning, language, or formatting of the technical content.
11/18/2013	1.6	Minor	Clarified the meaning of the technical content.
2/10/2014	1.6	No Change	No changes to the meaning, language, or formatting of the technical content.
4/30/2014	1.6	No Change	No changes to the meaning, language, or formatting of the technical content.

<b>Date</b>	<b>Revision History</b>	<b>Revision Class</b>	<b>Comments</b>
7/31/2014	1.6	No Change	No changes to the meaning, language, or formatting of the technical content.
8/24/2015	2.0	Major	Significantly changed the technical content.

Preliminary

# Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>8</b>
1.1	Glossary .....	8
1.2	References .....	9
1.2.1	Normative References .....	9
1.2.2	Informative References .....	10
1.3	Overview .....	10
1.4	Relationship to Other Protocols .....	10
1.5	Prerequisites/Preconditions .....	11
1.6	Applicability Statement .....	11
1.7	Versioning and Capability Negotiation .....	11
1.8	Vendor-Extensible Fields .....	11
1.9	Standards Assignments.....	11
<b>2</b>	<b>Messages.....</b>	<b>12</b>
2.1	Transport .....	12
2.2	Common Message Syntax .....	12
2.2.1	Namespaces .....	12
2.2.2	Messages.....	12
2.2.3	Elements .....	13
2.2.4	Complex Types.....	13
2.2.4.1	ArrayOfDecompRenderResultLevel .....	13
2.2.4.2	ArrayOfDecompRenderResultNode .....	13
2.2.4.3	DecompRenderResultLevel .....	14
2.2.4.4	DecompRenderResultNode .....	14
2.2.4.5	DecompRenderResultRecord .....	15
2.2.5	Simple Types .....	16
2.2.6	Attributes .....	16
2.2.7	Groups .....	16
2.2.8	Attribute Groups.....	16
2.2.9	Common Data Structures .....	16
<b>3</b>	<b>Protocol Details .....</b>	<b>17</b>
3.1	Server Details.....	17
3.1.1	Abstract Data Model.....	18
3.1.2	Timers .....	18
3.1.3	Initialization.....	18
3.1.4	Message Processing Events and Sequencing Rules .....	18
3.1.4.1	AutoExpand.....	19
3.1.4.1.1	Messages .....	19
3.1.4.1.1.1	AutoExpandSoapIn.....	19
3.1.4.1.1.2	AutoExpandSoapOut .....	20
3.1.4.1.2	Elements.....	20
3.1.4.1.2.1	AutoExpand .....	20
3.1.4.1.2.2	AutoExpandResponse .....	20
3.1.4.1.3	Complex Types .....	21
3.1.4.1.4	Simple Types .....	21
3.1.4.1.5	Attributes .....	21
3.1.4.1.6	Groups.....	21
3.1.4.1.7	Attribute Groups.....	21
3.1.4.2	CreateNavigationList .....	21
3.1.4.2.1	Messages .....	21
3.1.4.2.1.1	CreateNavigationListSoapIn .....	22
3.1.4.2.1.2	CreateNavigationListSoapOut .....	22
3.1.4.2.2	Elements.....	22
3.1.4.2.2.1	CreateNavigationList .....	22

3.1.4.2.2.2	CreateNavigationListResponse .....	23
3.1.4.2.3	Complex Types .....	23
3.1.4.2.3.1	DecompNavigationList .....	23
3.1.4.2.3.2	ArrayOfDecompNavigationGroup .....	24
3.1.4.2.3.3	DecompNavigationGroup .....	24
3.1.4.2.3.4	ArrayOfDecompNavigationAction .....	24
3.1.4.2.3.5	DecompNavigationAction .....	25
3.1.4.2.4	Simple Types .....	25
3.1.4.2.5	Attributes .....	25
3.1.4.2.6	Groups .....	25
3.1.4.2.7	Attribute Groups .....	25
3.1.4.3	Expand .....	25
3.1.4.3.1	Messages .....	26
3.1.4.3.1.1	ExpandSoapIn .....	26
3.1.4.3.1.2	ExpandSoapOut .....	26
3.1.4.3.2	Elements .....	26
3.1.4.3.2.1	Expand .....	26
3.1.4.3.2.2	ExpandResponse .....	27
3.1.4.3.3	Complex Types .....	27
3.1.4.3.4	Simple Types .....	27
3.1.4.3.5	Attributes .....	27
3.1.4.3.6	Groups .....	27
3.1.4.3.7	Attribute Groups .....	28
3.1.4.4	GetMemberProperties .....	28
3.1.4.4.1	Messages .....	28
3.1.4.4.1.1	GetMemberPropertiesSoapIn .....	28
3.1.4.4.1.2	GetMemberPropertiesSoapOut .....	28
3.1.4.4.2	Elements .....	29
3.1.4.4.2.1	GetMemberProperties .....	29
3.1.4.4.2.2	GetMemberPropertiesResponse .....	29
3.1.4.4.3	Complex Types .....	29
3.1.4.4.3.1	DecompMemberProperties .....	30
3.1.4.4.3.2	ArrayOfDecompMemberProperty .....	30
3.1.4.4.3.3	DecompMemberProperty .....	30
3.1.4.4.4	Simple Types .....	31
3.1.4.4.5	Attributes .....	31
3.1.4.4.6	Groups .....	31
3.1.4.4.7	Attribute Groups .....	31
3.1.4.5	GetRootLevel .....	31
3.1.4.5.1	Messages .....	31
3.1.4.5.1.1	GetRootLevelSoapIn .....	32
3.1.4.5.1.2	GetRootLevelSoapOut .....	32
3.1.4.5.2	Elements .....	32
3.1.4.5.2.1	GetRootLevel .....	32
3.1.4.5.2.2	GetRootLevelResponse .....	33
3.1.4.5.3	Complex Types .....	33
3.1.4.5.3.1	DecompTupleRecord .....	33
3.1.4.5.3.2	ArrayOfString .....	34
3.1.4.5.4	Simple Types .....	34
3.1.4.5.5	Attributes .....	34
3.1.4.5.6	Groups .....	34
3.1.4.5.7	Attribute Groups .....	34
3.1.5	Timer Events .....	34
3.1.6	Other Local Events .....	34
<b>4</b>	<b>Protocol Examples .....</b>	<b>35</b>
4.1	Starting a Session and Expanding to a Second Level .....	35
4.2	Retrieving a List of Named Actions and Expanding to a Third Level .....	36

4.3	Retrieving Properties .....	39
<b>5</b>	<b>Security .....</b>	<b>40</b>
5.1	Security Considerations for Implementers .....	40
5.2	Index of Security Parameters .....	40
<b>6</b>	<b>Appendix A: Full WSDL .....</b>	<b>41</b>
<b>7</b>	<b>Appendix B: Product Behavior .....</b>	<b>47</b>
<b>8</b>	<b>Change Tracking.....</b>	<b>48</b>
<b>9</b>	<b>Index.....</b>	<b>50</b>

Preliminary

# 1 Introduction

The PerformancePoint Services Decomposition Tree Protocol is used by a protocol client to do contribution analysis on data known to the protocol server.

Sections 1.8, 2, and 3 of this specification are normative and can contain the terms MAY, SHOULD, MUST, MUST NOT, and SHOULD NOT as defined in [\[RFC2119\]](#). Sections 1.5 and 1.9 are also normative but do not contain those terms. All other sections and examples in this specification are informative.

## 1.1 Glossary

The following terms are specific to this document:

**data source:** A database, web service, disk, file, or other collection of information from which data is queried or submitted. Supported data sources vary based on application and data provider.

**Hypertext Transfer Protocol (HTTP):** An application-level protocol for distributed, collaborative, hypermedia information systems (text, graphic images, sound, video, and other multimedia files) on the World Wide Web.

**Hypertext Transfer Protocol Secure (HTTPS):** An extension of HTTP that securely encrypts and decrypts web page requests. In some older protocols, "Hypertext Transfer Protocol over Secure Sockets Layer" is still used (Secure Sockets Layer has been deprecated). For more information, see [\[SSL3\]](#) and [\[RFC5246\]](#).

**level:** A relative position in a hierarchy of data. A level is frequently used when describing how to navigate a hierarchy in an Online Analytical Processing (OLAP) database or a PivotTable report.

**list:** A container within a SharePoint site that stores list items. A list has a customizable schema that is composed of one or more fields.

**measure:** In a cube, a set of values that are typically numeric and are based on a column (2) in the fact table of the cube. Measures are the central values that are aggregated and analyzed.

**node:** A location in a diagram that can have links to other locations.

**Secure Sockets Layer (SSL):** A security protocol that supports confidentiality and integrity of messages in client and server applications that communicate over open networks. SSL uses two keys to encrypt data—a public key known to everyone and a private or secret key known only to the recipient of the message. SSL supports server and, optionally, client authentication (2) using X.509 certificates (2). For more information, see [\[X509\]](#). The SSL protocol is precursor to Transport Layer Security (TLS). The TLS version 1.0 specification is based on SSL version 3.0 [\[SSL3\]](#).

**server:** A computer on which the remote procedure call (RPC) server is executing.

**SOAP:** A lightweight protocol for exchanging structured information in a decentralized, distributed environment. **SOAP** uses XML technologies to define an extensible messaging framework, which provides a message construct that can be exchanged over a variety of underlying protocols. The framework has been designed to be independent of any particular programming model and other implementation-specific semantics. SOAP 1.2 supersedes SOAP 1.1. See [\[SOAP1.2-1/2003\]](#).

**SOAP action:** The HTTP request header field used to indicate the intent of the **SOAP** request, using a URI value. See [\[SOAP1.1\]](#) section 6.1.1 for more information.

**SOAP body:** A container for the payload data being delivered by a SOAP message to its recipient. See [\[SOAP1.2-1/2007\]](#) section 5.3 for more information.



**SOAP fault:** A container for error and status information within a SOAP message. See [SOAP1.2-1/2007] section 5.4 for more information.

**Web Services Description Language (WSDL):** An XML format for describing network services as a set of endpoints that operate on messages that contain either document-oriented or procedure-oriented information. The operations and messages are described abstractly and are bound to a concrete network protocol and message format in order to define an endpoint. Related concrete endpoints are combined into abstract endpoints, which describe a network service. WSDL is extensible, which allows the description of endpoints and their messages regardless of the message formats or network protocols that are used.

**WSDL message:** An abstract, typed definition of the data that is communicated during a **WSDL operation** [WSDL]. Also, an element that describes the data being exchanged between web service providers and clients.

**WSDL operation:** A single action or function of a web service. The execution of a WSDL operation typically requires the exchange of messages between the service requestor and the service provider.

**XML namespace:** A collection of names that is used to identify elements, types, and attributes in XML documents identified in a URI reference [RFC3986]. A combination of XML namespace and local name allows XML documents to use elements, types, and attributes that have the same names but come from different sources. For more information, see [XMLNS-2ED].

**XML namespace prefix:** An abbreviated form of an **XML namespace**, as described in [XML].

**XML schema:** A description of a type of XML document that is typically expressed in terms of constraints on the structure and content of documents of that type, in addition to the basic syntax constraints that are imposed by XML itself. An XML schema provides a view of a document type at a relatively high level of abstraction.

**MAY, SHOULD, MUST, SHOULD NOT, MUST NOT:** These terms (in all caps) are used as defined in [RFC2119]. All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

## 1.2 References

Links to a document in the Microsoft Open Specifications library point to the correct section in the most recently published version of the referenced document. However, because individual documents in the library are not updated at the same time, the section numbers in the documents may not match. You can confirm the correct section numbering by checking the [Errata](#).

### 1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com). We will assist you in finding the relevant information.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://www.rfc-editor.org/rfc/rfc2119.txt>

[RFC2616] Fielding, R., Gettys, J., Mogul, J., et al., "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999, <http://www.rfc-editor.org/rfc/rfc2616.txt>

[SOAP1.1] Box, D., Ehnebuske, D., Kakivaya, G., et al., "Simple Object Access Protocol (SOAP) 1.1", May 2000, <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>

[SOAP1.2/1] Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J., and Nielsen, H.F., "SOAP Version 1.2 Part 1: Messaging Framework", W3C Recommendation, June 2003, <http://www.w3.org/TR/2003/REC-soap12-part1-20030624>

[SOAP1.2/2] Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J., and Nielsen, H.F., "SOAP Version 1.2 Part 2: Adjuncts", W3C Recommendation, June 2003, <http://www.w3.org/TR/2003/REC-soap12-part2-20030624>

[WSDL] Christensen, E., Curbera, F., Meredith, G., and Weerawarana, S., "Web Services Description Language (WSDL) 1.1", W3C Note, March 2001, <http://www.w3.org/TR/2001/NOTE-wsdl-20010315>

[XMLNS] Bray, T., Hollander, D., Layman, A., et al., Eds., "Namespaces in XML 1.0 (Third Edition)", W3C Recommendation, December 2009, <http://www.w3.org/TR/2009/REC-xml-names-20091208/>

[XMLSCHEMA1] Thompson, H., Beech, D., Maloney, M., and Mendelsohn, N., Eds., "XML Schema Part 1: Structures", W3C Recommendation, May 2001, <http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/>

[XMLSCHEMA2] Biron, P.V., Ed. and Malhotra, A., Ed., "XML Schema Part 2: Datatypes", W3C Recommendation, May 2001, <http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/>

### 1.2.2 Informative References

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000, <http://www.rfc-editor.org/rfc/rfc2818.txt>

### 1.3 Overview

This protocol allows a protocol client to do contribution analysis on a data value using a **data source** or data source context known to the protocol server.

The protocol allows a protocol client to retrieve grouped, named actions that apply to a specific data value, or named properties and values associated with that **node**, and then using action name and context retrieved in a previous call, retrieve additional values that contribute to the value.

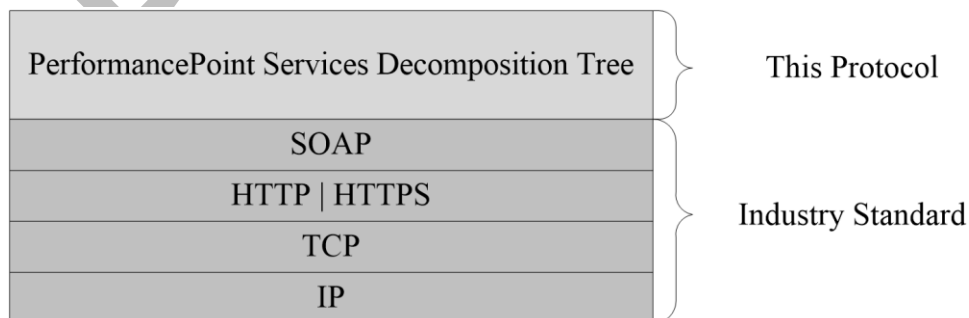
The protocol server returns all data points or some data points, and provides or denies summary data for those data points not returned.

A typical scenario for using this protocol is to display an interactive contribution analysis visualization.

### 1.4 Relationship to Other Protocols

This protocol uses the **SOAP** message protocol for formatting request and response messages, as described in [SOAP1.1], [SOAP1.2/1] and [SOAP1.2/2]. It transmits those messages by using **HTTP**, as described in [RFC2616], or **Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)**, as described in [RFC2818].

The following diagram shows the underlying messaging and transport stack used by the protocol:



**Figure 1: This protocol in Relation to Other Protocols**

## 1.5 Prerequisites/Preconditions

This protocol operates against a site that is identified by a URL that is known by protocol clients. The protocol server endpoint is formed by appending "/\_vti\_bin/PPS/PPSDecompRenderingService.asmx" to the URL of the site, for example:

http://www.example.com/Repository/\_vti\_bin/PPS/PPSDecompRenderingService.asmx.

This protocol assumes that authentication has been performed by the underlying protocols.

## 1.6 Applicability Statement

This protocol enables communication between a protocol server, which makes connections to a variety of data sources, and a protocol client.

## 1.7 Versioning and Capability Negotiation

This document specifies versioning issues in the following areas:

- **Supported Transports:** This protocol uses multiple transports with SOAP as specified in section Transport (see section 2.1)
- **Security and Authentication Methods:** The security and authentication methods supported by this protocol are specified in section Security (see section 5).
- **Localization:** This protocol includes text strings in various messages. Localization considerations for such strings are specified in the Common Message Syntax (Section [2.2](#)) and Message Processing Events and Sequencing Rules (Section [3.1.4](#)) sections.

## 1.8 Vendor-Extensible Fields

None.

## 1.9 Standards Assignments

None.

## 2 Messages

Occasionally, instances occur where this web service's WSDL specifies optionality (for example, `minOccurs="0"` and `maxOccurs="1"`) that conflicts with the stated behavior in this document, or the document doesn't contain the stated behavior. In such cases where the document and the WSDL conflict, the document takes precedence. In cases where the document doesn't specify optionality, the argument has to be present, even if the WSDL specifies that the argument is optional (for example, `minOccurs="0"`).

In cases where an argument is optional and the WSDL specifies both `"minOccurs=0"` and `"Nillable=true"`, treat the argument as only `"minOccurs=0"`.

### 2.1 Transport

Protocol servers **MUST** support SOAP over HTTP. Protocol servers **SHOULD** additionally support SOAP over HTTPS for securing communication with protocol clients.

Protocol messages **MUST** be formatted as specified either in [\[SOAP1.1\]](#), section 4 or in [\[SOAP1.2/1\]](#), section 5. Protocol server faults **MUST** be returned either using HTTP Status Codes as specified in [\[RFC2616\]](#), section 10 or using **SOAP faults** as specified either in [\[SOAP1.1\]](#), section 4.4 or in [\[SOAP1.2/1\]](#), section 5.4. The version of the SOAP fault returned **MUST** correspond to the version of SOAP used for the request **WSDL message**.

### 2.2 Common Message Syntax

This section contains common definitions that are used by this protocol. The syntax of the definitions uses **XML schema**, as specified in [\[XMLSCHEMA1\]](#) and [\[XMLSCHEMA2\]](#), and **WSDL**, as specified in [\[WSDL\]](#).

#### 2.2.1 Namespaces

This specification defines and references various **XML namespaces** using the mechanisms specified in [\[XMLNS\]](#). Although this specification associates a specific **XML namespace prefix** for each XML namespace that is used, the choice of any particular XML namespace prefix is implementation-specific and not significant for interoperability.

Prefix	Namespace URI	Reference
http	<a href="http://schemas.xmlsoap.org/wsd/http/">http://schemas.xmlsoap.org/wsd/http/</a>	<a href="#">[RFC2616]</a>
soap	<a href="http://schemas.xmlsoap.org/wsd/soap/">http://schemas.xmlsoap.org/wsd/soap/</a>	<a href="#">[SOAP1.1]</a>
soap12	<a href="http://schemas.xmlsoap.org/wsd/soap12/">http://schemas.xmlsoap.org/wsd/soap12/</a>	<a href="#">[SOAP1.2/1]</a> <a href="#">[SOAP1.2/2]</a>
tns	<a href="http://www.microsoft.com/performancepoint/scorecards">http://www.microsoft.com/performancepoint/scorecards</a>	
wSDL	<a href="http://schemas.xmlsoap.org/wsd/">http://schemas.xmlsoap.org/wsd/</a>	<a href="#">[WSDL]</a>
xs	<a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a>	<a href="#">[XMLSCHEMA1]</a> <a href="#">[XMLSCHEMA2]</a>

## 2.2.2 Messages

This specification does not define any common WSDL message definitions.

## 2.2.3 Elements

This specification does not define any common XML schema element definitions.

## 2.2.4 Complex Types

The following table summarizes the set of common XML schema complex type definitions defined by this specification. XML schema complex type definitions that are specific to a particular operation are described with the operation.

Complex type	Description
<b>ArrayOfDecompRenderResultLevel</b>	The <b>ArrayOfDecompRenderResultLevel</b> complex type specifies an array of <b>DecompRenderResultLevel</b> complex types (section <a href="#">2.2.4.3</a> ).
<b>ArrayOfDecompRenderResultNode</b>	The <b>ArrayOfDecompRenderResultNode</b> complex type specifies an array of <b>DecompRenderResultNode</b> complex types (section <a href="#">2.2.4.4</a> ).
<b>DecompRenderResultLevel</b>	The <b>DecompRenderResultLevel</b> complex type represents an ordered list of nodes and count of nodes, where total count includes nodes not returned.
<b>DecompRenderResultNode</b>	The <b>DecompRenderResultNode</b> complex type represents the details of a node.
<b>DecompRenderResultRecord</b>	The <b>DecompRenderResultRecord</b> complex type specifies details and results of a query.

### 2.2.4.1 ArrayOfDecompRenderResultLevel

**Namespace:** <http://www.microsoft.com/performancepoint/scorecards>

The **ArrayOfDecompRenderResultLevel** complex type specifies an array of **DecompRenderResultLevel** complex types (section [2.2.4.3](#)).

```
<xs:complexType name="ArrayOfDecompRenderResultLevel"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="DecompRenderResultLevel"
nillable="true" type="tns:DecompRenderResultLevel"/>
  </xs:sequence>
</xs:complexType>
```

**DecompRenderResultLevel:** The details about a **level**.

### 2.2.4.2 ArrayOfDecompRenderResultNode

**Namespace:** <http://www.microsoft.com/performancepoint/scorecards>

The **ArrayOfDecompRenderResultNode** complex type specifies an array of **DecompRenderResultNode** complex types (section [2.2.4.4](#)).

```

<xs:complexType name="ArrayOfDecompRenderResultNode"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="DecompRenderResultNode"
nillable="true" type="tns:DecompRenderResultNode"/>
  </xs:sequence>
</xs:complexType>

```

**DecompRenderResultNode:** The details about a node.

### 2.2.4.3 DecompRenderResultLevel

**Namespace:** http://www.microsoft.com/performancepoint/scorecards

The **DecompRenderResultLevel** complex type represents an ordered list of nodes and count of nodes, where total count includes nodes not returned.

```

<xs:complexType name="DecompRenderResultLevel" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="SortedBy" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="HierarchyName" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="TotalNodeCount" type="xs:int"/>
    <xs:element minOccurs="0" maxOccurs="1" name="UniqueName" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Caption" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Nodes"
type="tns:ArrayOfDecompRenderResultNode"/>
  </xs:sequence>
</xs:complexType>

```

**SortedBy:** Specifies the direction in which the resulting [ArrayOfDecompRenderResultNode](#) is sorted by [DecompRenderResultNode](#) Value. This element MUST be "ASCENDING" or "DESCENDING" (without quotes), or any case variation of those two values.

**HierarchyName:** If present, server MAY pass an empty string, this element MUST be ignored by client.

**TotalNodeCount:** Specifies the total number of nodes. This element MUST be equal to or greater than the number of [DecompRenderResultNode](#) elements returned in the [Nodes](#) element. This is not the actual number of [DecompRenderResultNode](#) element returned.

**UniqueName:** Uniquely identifies the level and is defined by the web service. This element MUST NOT be NULL or EMPTY.

**Caption:** Specifies the user-friendly name of the resulting level.

**Nodes:** Specifies a collection of [DecompRenderResultNode](#) elements.

### 2.2.4.4 DecompRenderResultNode

**Namespace:** http://www.microsoft.com/performancepoint/scorecards

The **DecompRenderResultNode** complex type represents the details of a node.

```

<xs:complexType name="DecompRenderResultNode" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="Value" type="xs:double"/>
    <xs:element minOccurs="0" maxOccurs="1" name="FormattedValue" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="HasChildren" type="xs:boolean"/>
    <xs:element minOccurs="0" maxOccurs="1" name="UniqueName" type="xs:string"/>
  </xs:sequence>
</xs:complexType>

```

```

<xs:element minOccurs="0" maxOccurs="1" name="Caption" type="xs:string"/>
<xs:element minOccurs="1" maxOccurs="1" name="BottomSumValue" type="xs:double"/>
<xs:element minOccurs="0" maxOccurs="1" name="BottomSumFormattedValue" type="xs:string"/>
<xs:element minOccurs="1" maxOccurs="1" name="TopSumValue" type="xs:double"/>
<xs:element minOccurs="0" maxOccurs="1" name="TopSumFormattedValue" type="xs:string"/>
</xs:sequence>
</xs:complexType>

```

**Value:** Specifies the raw numeric value of the node.

**FormattedValue:** Specifies the user-friendly value of the node.

**HasChildren:** The protocol server SHOULD return TRUE if the [AutoExpand](#) method called using this node's **UniqueName** will return a positive number of [DecompRenderResultNode](#) elements. Otherwise, the server MUST return FALSE.

**UniqueName:** Uniquely identifies the node and is defined by the web service. This element MUST NOT be NULL or EMPTY.

**Caption:** Specifies the user-friendly name of this node.

**BottomSumValue:** If the **MeasureIsAdditive** property of the associated [DecompRenderResultRecord](#) is TRUE, this element SHOULD specify the sum of all the **Values** of the [DecompRenderResultNodes](#) occurring after this node in the resulting [ArrayOfDecompRenderResultNode](#).

**BottomSumFormattedValue:** If the **MeasureIsAdditive** property of the associated [DecompRenderResultRecord](#) is TRUE, this element SHOULD specify the user-friendly value of the **BottomSumValue**.

**TopSumValue:** If the **MeasureIsAdditive** property of the associated [DecompRenderResultRecord](#) is TRUE, this element SHOULD specify the sum of all the **Values** of the [DecompRenderResultNodes](#) occurring before this node in the resulting [ArrayOfDecompRenderResultNode](#).

**TopSumFormattedValue:** If the **MeasureIsAdditive** property of the associated [DecompRenderResultRecord](#) is TRUE, this SHOULD specify the user-friendly value of the **TopSumValue**.

#### 2.2.4.5 DecompRenderResultRecord

**Namespace:** <http://www.microsoft.com/performancepoint/scorecards>

The **DecompRenderResultRecord** complex type specifies details and results of a query.

```

<xs:complexType name="DecompRenderResultRecord" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Levels"
type="tns:ArrayOfDecompRenderResultLevel"/>
    <xs:element minOccurs="0" maxOccurs="1" name="MeasureCaption" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="MeasureUniqueName" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="MeasureIsAdditive" type="xs:boolean"/>
    <xs:element minOccurs="0" maxOccurs="1" name="NavigationContext" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="ErrorMsg" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="ErrorDetails" type="xs:string"/>
  </xs:sequence>
</xs:complexType>

```

**Levels:** MUST specify an [ArrayOfDecompRenderResultLevel](#) containing no more than one [DecompRenderResultLevel](#) element.

**MeasureCaption:** Specifies the user-friendly name of the associated **measure**.

**MeasureUniqueName:** Uniquely identifies the measure and is defined by the web service. This element MUST NOT be NULL or EMPTY.

**MeasureIsAdditive:** If TRUE, the **TopSumValue**, **TopSumFormattedValue**, **BottomSumValue**, and **BottomSumFormattedValue** elements of all the [DecompRenderResultNode](#) element instances contained in the **Levels** element of this **DecompRenderResultRecord** MUST specify sum values. If FALSE, the protocol client MUST ignore the **TopSumValue**, **TopSumFormattedValue**, **BottomSumValue**, and **BottomSumFormattedValue** values.

**NavigationContext:** Specifies a session-specific value that specifies the current state of the protocol server.

**ErrorMsg:** Specifies an error message in the case of an application error on the protocol server.

**ErrorDetails:** Specifies the error details in the case of an application error on the protocol server.

### 2.2.5 Simple Types

This specification does not define any common XML schema simple type definitions.

### 2.2.6 Attributes

This specification does not define any common XML schema attribute definitions.

### 2.2.7 Groups

This specification does not define any common XML schema group definitions.

### 2.2.8 Attribute Groups

This specification does not define any common XML schema attribute group definitions.

### 2.2.9 Common Data Structures

This specification does not define any common XML schema data structures.



### 3 Protocol Details

This protocol allows protocol servers to perform implementation-specific localization of text in various messages. The localization of this text is an implementation-specific behavior of the protocol server and not significant for interoperability.

Occasionally, instances occur where this web service's WSDL specifies optionality (for example, `minOccurs="0"` and `maxOccurs="1"`) that conflicts with the stated behavior in this document, or the document doesn't contain the stated behavior. In such cases where the document and the WSDL conflict, the document takes precedence. In cases where the document doesn't specify optionality, the argument has to be present, even if the WSDL specifies that the argument is optional (for example, `minOccurs="0"`).

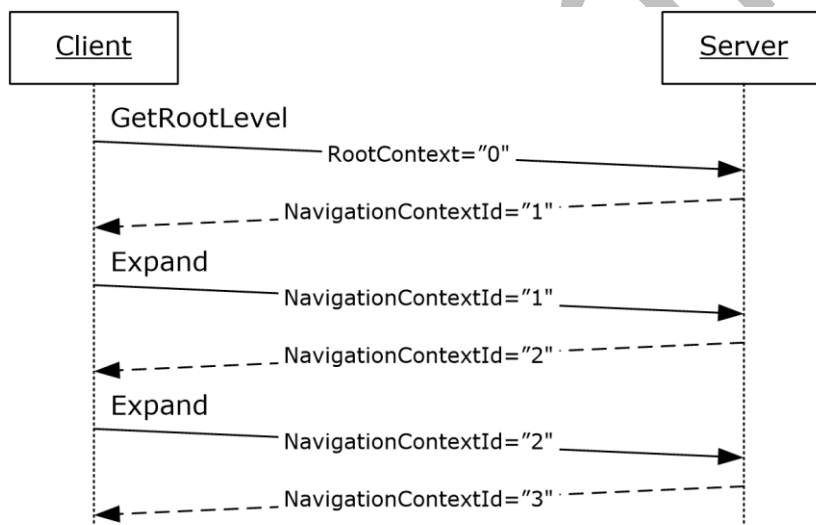
In cases where an argument is optional and the WSDL specifies both `"minOccurs=0"` and `"Nillable=true"`, treat the argument as only `"minOccurs=0"`.

The client side of this protocol is simply a pass-through. That is, no additional timers or other state is required on the client side of this protocol. Calls made by the higher-layer protocol or application are passed directly to the transport, and the results returned by the transport are passed directly back to the higher-layer protocol or application.

Except where specified, protocol clients SHOULD interpret HTTP Status Codes returned by the protocol server as specified in [RFC2616](#), section 10.

#### 3.1 Server Details

The following is a high-level sequence diagram illustrating the operation of the protocol.



**Figure 2: Protocol high level sequence diagram**

First, a protocol client establishes a session with a protocol server with the **GetRootLevel** operation (section [3.1.4.5](#)). The client passes in a *RootContextId* known to the server. The server responds with a *NavigationContextId* which represents a session state variable. Navigation operations, **AutoExpand** (section [3.1.4.1](#)) and **Expand** (section [3.1.4.3](#)), require the client to pass the session state to the server. The protocol server will respond to each navigation operation with a new session state variable.

### 3.1.1 Abstract Data Model

This section describes a conceptual model of possible data organization that an implementation maintains to participate in this protocol. The described organization is provided to facilitate the explanation of how the protocol behaves. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with that described in this document.

**Root Context:** The session state known to protocol server and protocol client prior to use of operation **GetRootLevel** (section [3.1.4.5](#)).

This state allows the protocol server to keep private information it uses to generate responses, for example connection strings to data sources.

**Navigation Context:** The session state representing an ordered sequence of navigation operations following use of operation **GetRootLevel**. Navigation operations are **AutoExpand** (section [3.1.4.1](#)) and **Expand** (section [3.1.4.3](#)).

This state allows the protocol server to track the ordered sequence of navigation operations performed. For example, for the following operations:

- **GetRootLevel**, *navigationContextId* "0", protocol server responds with *navigationContext* 1"
- **Expand**, *navigationContext* "1" and *itemUniqueName* "[Product].[52]", protocol server responds with *navigationContext* "2"
- **Expand**, *navigationContext* "2" and *itemUniqueName* "[Customer].[79]"

The protocol server references Decomposition Navigation Context identified by "2" (representing the preceding **Expand** operation and its parameters, and the preceding **GetRootLevel** operation and its parameters) to respond to the final **Expand** operation.

### 3.1.2 Timers

If a request exceeds a timeout period, the **server** MUST return a SOAP FAULT response.

### 3.1.3 Initialization

This protocol operates against a site that is identified by a URL that is known by protocol clients. The protocol server endpoint is formed by appending "/\_vti\_bin/PPS/PPSDecompRenderingService.asmx" to the URL of the site, for example:

[http://www.example.com/\\_vti\\_bin/PPS/PPSDecompRenderingService.asmx](http://www.example.com/_vti_bin/PPS/PPSDecompRenderingService.asmx).

A connection that uses the underlying protocol layers that are specified in section 1.4 MUST be established before using this protocol.

### 3.1.4 Message Processing Events and Sequencing Rules

This section specifies the protocol operations.

The following table summarizes the list of operations as defined by this specification.

Operation	Description
<b>AutoExpand</b>	The <b>AutoExpand</b> operation is used to get additional node data for a specific node, where named actions are not known. Also see <b>Expand</b> section <a href="#">3.1.4.3</a> .
<b>CreateNavigationList</b>	The <b>CreateNavigationList</b> operation returns a list of named actions for use with the <i>navigationItemUniqueName</i> element of the <b>Expand</b> operation (section 3.1.4.3).

Operation	Description
<b>Expand</b>	The <b>Expand</b> operation is used to get additional node data for a specific node, using a named action.
<b>GetMemberProperties</b>	The <b>GetMemberProperties</b> operation returns properties associated with a specified node.
<b>GetRootLevel</b>	The <b>GetRootLevel</b> operation is used to start a session. A protocol client MUST use this operation to retrieve the Navigation Context session state variable before using other operations.

### 3.1.4.1 AutoExpand

The **AutoExpand** operation is used to get additional node data for a specific node, where named actions are not known. Also see **Expand** section [3.1.4.3](#).

The following is the WSDL port type specification of the **AutoExpand WSDL operation**.

```
<wsdl:operation name="AutoExpand" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:AutoExpandSoapIn"/>
  <wsdl:output message="tns:AutoExpandSoapOut"/>
</wsdl:operation>
```

A session state variable is required to successfully complete this operation. As part of the **AutoExpandSoapIn** request message (section 3.1.4.1.1), **navigationContext** MUST contain a value obtained from a **DecompRenderResultRecord** result from a previous successful request to either a **GetRootLevel** (section 3.1.4.5), **AutoExpand** (section 3.1.4.1), or **Expand** (section 3.1.4.3) operation.

A session state variable, *Navigation Context*, is created in this operation. The *NavigationContext* value is returned as part of the **DecompRenderResultRecord** element. All requests using other protocol messages, such as the **AutoExpand**, **Expand**, **CreateNavigationList** (section 3.1.4.2) and **GetMemberProperties** (section 3.1.4.4), require such a session state variable.

In the event of an application error on the protocol server during the operation, the protocol server MUST return a SOAP FAULT response.

#### 3.1.4.1.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
<b>AutoExpandSoapIn</b>	The request WSDL message for the <b>AutoExpand</b> WSDL operation.
<b>AutoExpandSoapOut</b>	The response WSDL message for the <b>AutoExpand</b> WSDL operation.

##### 3.1.4.1.1.1 AutoExpandSoapIn

The request WSDL message for the **AutoExpand** WSDL operation.

The **SOAP action** value is:

The **SOAP body** contains the **AutoExpand** element.

### 3.1.4.1.1.2 AutoExpandSoapOut

The response WSDL message for the **AutoExpand** WSDL operation.

The SOAP body contains the **AutoExpandResponse** element.

### 3.1.4.1.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
<b>AutoExpand</b>	The input data for the <b>AutoExpand</b> WSDL operation.
<b>AutoExpandResponse</b>	The result data for the <b>AutoExpand</b> WSDL operation.

#### 3.1.4.1.2.1 AutoExpand

The **AutoExpand** element specifies the input data for the **AutoExpand** WSDL operation.

```
<xs:element name="AutoExpand" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="levelUniqueName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="itemUniqueName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="sortType" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="navigationContext" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

**levelUniqueName:** Uniquely identifies the level to expand. This identifier is defined by the web service and is the **DecompRenderResultLevel** unique name. It MUST NOT be NULL or EMPTY.

**itemUniqueName:** Uniquely identifies the node to expand. This identifier is defined by the web service and is the **DecompRenderResultNode** unique name. It MUST NOT be NULL or EMPTY.

**sortType:** Specifies how the result nodes are sorted. This element MUST be "ASCENDING" or "DESCENDING" (without quotes), and MUST NOT be case sensitive. The **DecompRenderResultNode** elements in **ExpandResponse** (section [3.1.4.3.2.2](#)) MUST be sorted according to the specified value. If empty, NULL, or not specified, the protocol server MUST use "DESCENDING".

**navigationContext:** A session identifier which defines the state of the client. MUST be the *NavigationContext* value from a **DecompRenderResultRecord** element obtained from a previous successful request to either a **GetRootLevel** (section 3.1.4.5), **AutoExpand** (section 3.1.4.1) or **Expand** (section 3.1.4.3) operation. This element MUST NOT be NULL and MUST NOT be empty.

#### 3.1.4.1.2.2 AutoExpandResponse

The **AutoExpandResponse** element specifies the result data for the **AutoExpand** WSDL operation.

```

<xs:element name="AutoExpandResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="AutoExpandResult"
type="tns:DecompRenderResultRecord"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

**AutoExpandResult:** A **DecompRenderResultRecord** element (section 2.2.4.5). In the event of an application error on the protocol server, this element will not occur.

### 3.1.4.1.3 Complex Types

None.

### 3.1.4.1.4 Simple Types

None.

### 3.1.4.1.5 Attributes

None.

### 3.1.4.1.6 Groups

None.

### 3.1.4.1.7 Attribute Groups

None.

### 3.1.4.2 CreateNavigationList

The **CreateNavigationList** operation returns a list of named actions for use with the *navigationItemUniqueName* element of the **Expand** operation (section 3.1.4.3).

The following is the WSDL port type specification of the **CreateNavigationList** WSDL operation.

```

<wsdl:operation name="CreateNavigationList" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:CreateNavigationListSoapIn"/>
  <wsdl:output message="tns:CreateNavigationListSoapOut"/>
</wsdl:operation>

```

Two tiers of information are returned. The first tier is a list of all the possible groups a node can expand to. Each group within that first tier SHOULD have a list of named actions.

A session state variable is required to successfully complete this operation. As part of the **CreateNavigationListSoapIn** request message (section 3.1.4.2.1.1), **navigationContext** MUST contain a value obtained from a **DecompRenderResultRecord** result from a previous successful request to either a **GetRootLevel** (section 3.1.4.5), **AutoExpand** (section 3.1.4.1), or **Expand** operation.

In the event of an application error on the protocol server during the operation, the protocol server MUST return a SOAP FAULT response.

### 3.1.4.2.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
<b>CreateNavigationListSoapIn</b>	The request WSDL message for the <b>CreateNavigationList</b> WSDL operation.
<b>CreateNavigationListSoapOut</b>	The response WSDL message for the <b>CreateNavigationList</b> WSDL operation.

#### 3.1.4.2.1.1 CreateNavigationListSoapIn

The request WSDL message for the **CreateNavigationList** WSDL operation.

The SOAP action value is:

```
http://www.microsoft.com/performancepoint/scorecards/CreateNavigationList
```

The SOAP body contains the **CreateNavigationList** element.

#### 3.1.4.2.1.2 CreateNavigationListSoapOut

The response WSDL message for the **CreateNavigationList** WSDL operation.

The SOAP body contains the **CreateNavigationListResponse** element.

#### 3.1.4.2.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
<b>CreateNavigationList</b>	The input data for the <b>CreateNavigationList</b> WSDL operation.
<b>CreateNavigationListResponse</b>	The result data for the <b>CreateNavigationList</b> WSDL operation.

#### 3.1.4.2.2.1 CreateNavigationList

The **CreateNavigationList** element specifies the input data for the **CreateNavigationList** WSDL operation.

```
<xs:element name="CreateNavigationList" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="levelUniqueName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="itemUniqueName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="navigationContext" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

**levelUniqueName:** Uniquely identifies the level to expand. This identifier is defined by the web service and is the **DecompRenderResultLevel** unique name. It MUST NOT be NULL or EMPTY.

**itemUniqueName:** Uniquely identifies the node to expand. This identifier is defined by the web service and is the **DecompRenderResultNode** unique name. It MUST NOT be NULL or EMPTY.

**navigationContext:** A session identifier which defines the state of the client. This element MUST be the *NavigationContext* value from a **DecompRenderResultRecord** element obtained from a previous successful request to either a **GetRootLevel** (section 3.1.4.5), **AutoExpand** (section 3.1.4.1) or **Expand** (section 3.1.4.3) operation. This element MUST NOT be NULL and MUST NOT be empty.

### 3.1.4.2.2 CreateNavigationListResponse

The **CreateNavigationListResponse** element specifies the result data for the **CreateNavigationList** WSDL operation.

```
<xs:element name="CreateNavigationListResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="CreateNavigationListResult"
        type="tns:DecompNavigationList"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

**CreateNavigationListResult:** A **DecompNavigationList** value (section 3.1.4.2.3.1). In the event of an application error on the protocol server, this value will not occur.

### 3.1.4.2.3 Complex Types

The following table summarizes the XML schema complex type definitions that are specific to this operation.

Complex type	Description
<b>ArrayOfDecompNavigationAction</b>	Specifies an array of <a href="#">DecompNavigationAction complex types</a> (section 3.1.4.2.3.5).
<b>ArrayOfDecompNavigationGroup</b>	Specifies an array of <a href="#">DecompNavigationGroup complex types</a> (section 3.1.4.2.3.3).
<b>DecompNavigationAction</b>	Contains details about a named action.
<b>DecompNavigationGroup</b>	Contains details about a group of named actions.
<b>DecompNavigationList</b>	Contains a collection of possible grouped, named actions.

#### 3.1.4.2.3.1 DecompNavigationList

**Namespace:** http://www.microsoft.com/performancepoint/scorecards

The **DecompNavigationList** complex type contains a collection of possible grouped, named actions.

```
<xs:complexType name="DecompNavigationList" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Groups"
      type="tns:ArrayOfDecompNavigationGroup"/>
  </xs:sequence>
</xs:complexType>
```

**Groups:** A collection of **DecompNavigationGroup** objects (section [3.1.4.2.3.3](#)).

### 3.1.4.2.3.2 ArrayOfDecompNavigationGroup

**Namespace:** http://www.microsoft.com/performancepoint/scorecards

The **ArrayOfDecompNavigationGroup** complex type specifies an array of **DecompNavigationGroup** complex types (section [3.1.4.2.3.3](#)).

```
<xs:complexType name="ArrayOfDecompNavigationGroup"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="DecompNavigationGroup"
nillable="true" type="tns:DecompNavigationGroup"/>
  </xs:sequence>
</xs:complexType>
```

**DecompNavigationGroup:** Contains details about a group of named actions.

### 3.1.4.2.3.3 DecompNavigationGroup

**Namespace:** http://www.microsoft.com/performancepoint/scorecards

The **DecompNavigationGroup** complex type contains details about a group of named actions.

```
<xs:complexType name="DecompNavigationGroup" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Caption" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Actions"
type="tns:ArrayOfDecompNavigationAction"/>
  </xs:sequence>
</xs:complexType>
```

**Name:** If present, the server MAY pass an empty string, this element MUST be ignored by the client.

**Caption:** MUST specify the user-friendly name of the group.

**Actions:** MUST specify a collection of **DecompNavigationAction** complex types (section [3.1.4.2.3.5](#)).

### 3.1.4.2.3.4 ArrayOfDecompNavigationAction

**Namespace:** http://www.microsoft.com/performancepoint/scorecards

The **ArrayOfDecompNavigationAction** complex type specifies an array of **DecompNavigationAction** complex types (section [3.1.4.2.3.5](#)).

```
<xs:complexType name="ArrayOfDecompNavigationAction"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="DecompNavigationAction"
nillable="true" type="tns:DecompNavigationAction"/>
  </xs:sequence>
</xs:complexType>
```

**DecompNavigationAction:** Contains details about a named action.



### 3.1.4.2.3.5 DecompNavigationAction

**Namespace:** http://www.microsoft.com/performancepoint/scorecards

The **DecompNavigationAction** complex type contains details about a named action.

```
<xs:complexType name="DecompNavigationAction" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Caption" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="ParentCaption" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="Enabled" type="xs:boolean"/>
  </xs:sequence>
</xs:complexType>
```

**Name:** MUST specify the unique name of the action.

**Caption:** MUST specify the user-friendly name of the action.

**ParentCaption:** Specifies a user-friendly categorization for this action.

**Enabled:** If FALSE, the protocol client MUST NOT pass this action's name as the value of the **navigationItemUniqueName** parameter of the **Expand** method (section 3.1.4.3). If TRUE, this action's name SHOULD be passed as the value of the **navigationItemUniqueName** parameter of the **Expand** method.

### 3.1.4.2.4 Simple Types

None.

### 3.1.4.2.5 Attributes

None.

### 3.1.4.2.6 Groups

None.

### 3.1.4.2.7 Attribute Groups

None.

### 3.1.4.3 Expand

The **Expand** operation is used to get additional node data for a specific node, using a named action.

The following is the WSDL port type specification of the **Expand** WSDL operation.

```
<wsdl:operation name="Expand" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:ExpandSoapIn"/>
  <wsdl:output message="tns:ExpandSoapOut"/>
</wsdl:operation>
```

A session state variable is required to successfully complete this operation. As part of the **ExpandSoapIn** request message (section 3.1.4.3.1.1), **navigationContext** MUST contain a value obtained from a **DecompRenderResultRecord** result from a previous successful request to either a **GetRootLevel** (section 3.1.4.5), **AutoExpand** (section 3.1.4.1), or **Expand** (section 3.1.4.3) operation.

A session state variable, *Navigation Context*, is created in this operation. The *NavigationContext* value is returned as part of the **DecompRenderResultRecord** element. All requests using other protocol messages, such as the **AutoExpand**, **Expand**, **CreateNavigationList** (section 3.1.4.2) and **GetMemberProperties** (section 3.1.4.4) operations, require such a session state variable.

In the event of an application error on the protocol server during the operation, the protocol server MUST return a SOAP FAULT response.

### 3.1.4.3.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
<b>ExpandSoapIn</b>	The request WSDL message for the <b>Expand</b> WSDL operation.
<b>ExpandSoapOut</b>	The response WSDL message for the <b>Expand</b> WSDL operation.

#### 3.1.4.3.1.1 ExpandSoapIn

The request WSDL message for the **Expand** WSDL operation.

The SOAP action value is:

```
http://www.microsoft.com/performancepoint/scorecards/Expand
```

The SOAP body contains the **Expand** element.

#### 3.1.4.3.1.2 ExpandSoapOut

The response WSDL message for the **Expand** WSDL operation.

The SOAP body contains the **ExpandResponse** element.

### 3.1.4.3.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
<b>Expand</b>	The input data for the <b>Expand</b> WSDL operation.
<b>ExpandResponse</b>	The result data for the <b>Expand</b> WSDL operation.

#### 3.1.4.3.2.1 Expand

The **Expand** element specifies the input data for the **Expand** WSDL operation.

```
<xs:element name="Expand" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="levelUniqueName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="itemUniqueName" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```

    <xs:element minOccurs="0" maxOccurs="1" name="navigationItemUniqueName"
type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="sortType" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="navigationContext" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
</xs:element>

```

**levelUniqueName:** Uniquely identifies the level to expand. This identifier is defined by the web service and is the **DecompRenderResultLevel** unique name. It MUST NOT be NULL or EMPTY.

**itemUniqueName:** Uniquely identifies the node to expand. This identifier is defined by the web service and is the **DecompRenderResultNode** unique name. It MUST NOT be NULL or EMPTY.

**navigationItemUniqueName:** MUST be the *Name* value from a **DecompNavigationAction** element obtained from a previous call to **CreateNavigationList** operation (section 3.1.4.2). This element MUST NOT be NULL and MUST NOT be empty.

**sortType:** Specifies how the result nodes are sorted. This element MUST be "ASCENDING" or "DESCENDING" (without quotes), and MUST NOT be case sensitive. The **DecompRenderResultNode** elements in the **ExpandResponse** element (section 3.1.4.3.2.2) MUST be sorted according to the specified value. If empty, NULL, or not specified, the protocol server MUST use "DESCENDING".

**navigationContext:** A session identifier which defines the state of the client. This element MUST be the *NavigationContext* value from a **DecompRenderResultRecord** element obtained from a previous successful request to either a **GetRootLevel** (section 3.1.4.5), **AutoExpand** (section 3.1.4.1), or **Expand** (section 3.1.4.3) operation. It MUST NOT be NULL and MUST NOT be empty.

### 3.1.4.3.2.2 ExpandResponse

The **ExpandResponse** element specifies the result data for the **Expand** WSDL operation.

```

<xs:element name="ExpandResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="ExpandResult"
type="tns:DecompRenderResultRecord"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

**ExpandResult:** A **DecompRenderResultRecord** value (section 2.2.4.5). In the event of an application error on the server, this value will not occur.

### 3.1.4.3.3 Complex Types

None.

### 3.1.4.3.4 Simple Types

None.

### 3.1.4.3.5 Attributes

None.

### 3.1.4.3.6 Groups

None.

### 3.1.4.3.7 Attribute Groups

None.

### 3.1.4.4 GetMemberProperties

The **GetMemberProperties** operation returns properties associated with a specified node.

The following is the WSDL port type specification of the **GetMemberProperties** WSDL operation.

```
<wsdl:operation name="GetMemberProperties" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">  
  <wsdl:input message="tns:GetMemberPropertiesSoapIn"/>  
  <wsdl:output message="tns:GetMemberPropertiesSoapOut"/>  
</wsdl:operation>
```

This operation returns a list of member properties associated with a node.

A session state variable is required to successfully complete this operation. As part of the **GetMemberPropertiesSoapIn** request message (section 3.1.4.4.1.1), **navigationContext** MUST contain a value obtained from a **DecompRenderResultRecord** result from a previous successful request to either a **GetRootLevel** (section 3.1.4.5), **AutoExpand** (section 3.1.4.1), or **Expand** (section 3.1.4.3) operation.

In the event of an application error on the protocol server during the operation, the protocol server MUST return a SOAP FAULT response.

#### 3.1.4.4.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
<b>GetMemberPropertiesSoapIn</b>	The request WSDL message for the <b>GetMemberProperties</b> WSDL operation.
<b>GetMemberPropertiesSoapOut</b>	The response WSDL message for the <b>GetMemberProperties</b> WSDL operation.

##### 3.1.4.4.1.1 GetMemberPropertiesSoapIn

The request WSDL message for the **GetMemberProperties** WSDL operation.

The SOAP action value is:

```
http://www.microsoft.com/performancepoint/scorecards/GetMemberProperties
```

The SOAP body contains the **GetMemberProperties** element.

##### 3.1.4.4.1.2 GetMemberPropertiesSoapOut

The response WSDL message for the **GetMemberProperties** WSDL operation.

The SOAP body contains the **GetMemberPropertiesResponse** element.

### 3.1.4.4.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
<b>GetMemberProperties</b>	The input data for the <b>GetMemberProperties</b> WSDL operation.
<b>GetMemberPropertiesResponse</b>	The result data for the <b>GetMemberProperties</b> WSDL operation.

#### 3.1.4.4.2.1 GetMemberProperties

The **GetMemberProperties** element specifies the input data for the **GetMemberProperties** WSDL operation.

```
<xs:element name="GetMemberProperties" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="navigationContext" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="itemUniqueName" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

**navigationContext:** A session identifier which defines the state of the client. This element MUST be the *NavigationContext* value from a **DecompRenderResultRecord** element obtained from a previous request to either a **GetRootLevel** (section 3.1.4.5), **AutoExpand** (section 3.1.4.1) or **Expand** (section 3.1.4.3) operation. It MUST NOT be NULL and MUST NOT be empty.

**itemUniqueName:** Uniquely identifies the node to expand. This identifier is defined by the web service and is the **DecompRenderResultNode** unique name. It MUST NOT be NULL or EMPTY.

#### 3.1.4.4.2.2 GetMemberPropertiesResponse

The **GetMemberPropertiesResponse** element specifies the result data for the **GetMemberProperties** WSDL operation.

```
<xs:element name="GetMemberPropertiesResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="GetMemberPropertiesResult"
type="tns:DecompMemberProperties"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

**GetMemberPropertiesResult:** A **DecompMemberProperties** value.

#### 3.1.4.4.3 Complex Types

The following table summarizes the XML schema complex type definitions that are specific to this operation.

Complex type	Description
<b>ArrayOfDecompMemberProperty</b>	Specifies an array of <a href="#">DecompMemberProperty complex types</a> (section 3.1.4.4.3.3).
<b>DecompMemberProperties</b>	Contains details about properties of the associated node.
<b>DecompMemberProperty</b>	Contains details about a property of the associated node.

### 3.1.4.4.3.1 DecompMemberProperties

**Namespace:** http://www.microsoft.com/performancepoint/scorecards

The **DecompMemberProperties** complex type contains details about properties of the associated node.

```
<xs:complexType name="DecompMemberProperties" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="MemberName" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Properties"
type="tns:ArrayOfDecompMemberProperty"/>
  </xs:sequence>
</xs:complexType>
```

**MemberName:** If present, the server MAY pass an empty string, this element MUST be ignored by the client.

**Properties:** Specifies a collection of **DecompMemberProperty** elements.

### 3.1.4.4.3.2 ArrayOfDecompMemberProperty

**Namespace:** http://www.microsoft.com/performancepoint/scorecards

The **ArrayOfDecompMemberProperty** complex type specifies an array of **DecompMemberProperty** complex types (section [3.1.4.4.3.1](#)).

```
<xs:complexType name="ArrayOfDecompMemberProperty"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="DecompMemberProperty"
nillable="true" type="tns:DecompMemberProperty"/>
  </xs:sequence>
</xs:complexType>
```

**DecompMemberProperty:** Contains details about a property.

### 3.1.4.4.3.3 DecompMemberProperty

**Namespace:** http://www.microsoft.com/performancepoint/scorecards

The **DecompMemberProperty** complex type contains details about a property of the associated node.

```
<xs:complexType name="DecompMemberProperty" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Caption" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

```
<xs:element minOccurs="0" maxOccurs="1" name="Value" type="xs:string"/>
</xs:sequence>
</xs:complexType>
```

**Name:** MUST be ignored.

**Caption:** Specifies the user-friendly name of the property.

**Value:** Specifies the value of the property.

#### 3.1.4.4.4 Simple Types

None.

#### 3.1.4.4.5 Attributes

None.

#### 3.1.4.4.6 Groups

None.

#### 3.1.4.4.7 Attribute Groups

None.

#### 3.1.4.5 GetRootLevel

The **GetRootLevel** operation is used to start a session. A protocol client MUST use this operation to retrieve the Navigation Context session state variable before using other operations.

The following is the WSDL port type specification of the **GetRootLevel** WSDL operation.

```
<wsdl:operation name="GetRootLevel" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:GetRootLevelSoapIn"/>
  <wsdl:output message="tns:GetRootLevelSoapOut"/>
</wsdl:operation>
```

This operation, if successful, will return a non-empty **DecompRenderResultRecord** complex type (section 2.2.4.5) that contains data for a specific node in the form of a **DecompRenderResultLevel** (section 2.2.4.3) and a **DecompRenderResultNode** (section 2.2.4.4).

A session state variable, *Navigation Context*, is created in this operation. The *NavigationContext* value is returned as part of the **DecompRenderResultRecord** element. All requests using other protocol messages, such as **AutoExpand** (section 3.1.4.1), **Expand** (section 3.1.4.3), **CreateNavigationList** (section 3.1.4.2) and **GetMemberProperties** (section 3.1.4.4), require such a session state variable.

In the event of an application error on the protocol server during the operation, the protocol server MUST return a SOAP FAULT response.

#### 3.1.4.5.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
<b>GetRootLevelSoapIn</b>	The request WSDL message for the <b>GetRootLevel</b> WSDL operation.
<b>GetRootLevelSoapOut</b>	The response WSDL message for the <b>GetRootLevel</b> WSDL operation.

### 3.1.4.5.1.1 GetRootLevelSoapIn

The request WSDL message for the **GetRootLevel** WSDL operation.

The SOAP action value is:

```
http://www.microsoft.com/performancepoint/scorecards/GetRootLevel
```

The SOAP body contains the **GetRootLevel** element.

### 3.1.4.5.1.2 GetRootLevelSoapOut

The response WSDL message for the **GetRootLevel** WSDL operation.

The SOAP body contains the **GetRootLevelResponse** element.

### 3.1.4.5.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
<b>GetRootLevel</b>	The input data for the <b>GetRootLevel</b> WSDL operation.
<b>GetRootLevelResponse</b>	The result data for the <b>GetRootLevel</b> WSDL operation.

### 3.1.4.5.2.1 GetRootLevel

The **GetRootLevel** element specifies the input data for the **GetRootLevel** WSDL operation.

```
<xs:element name="GetRootLevel" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="navigationType" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="navigationContextId" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="tupleRecord"
type="tns:DecompTupleRecord"/>
      <xs:element minOccurs="0" maxOccurs="1" name="viewState" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

**navigationType:** MUST be "olap" (without quotes.)

**navigationContextId:** MUST be a string identifying the Root Context session state known to the protocol server. This element MUST NOT be NULL and MUST NOT be empty. An implementation of this protocol, for example, could pass the identifier of the initial report containing the cell that is being analyzed as the value of this parameter.



**tupleRecord:** MUST be a **DecompTupleRecord** value (section 3.1.4.5.3.1). Specifies specific named filters to be applied to the response. It MUST NOT be NULL.

**viewState:** MAY be empty. If empty, this value is ignored. If not empty, this value is xml-formatted and is a set of name-value pairs, where the names MUST be a string type and unique, and its associated value MUST be a string type. For example:

```
<ViewState>
  <GridViewDataInputs />
  <TimeFilters />
</ViewState>
```

### 3.1.4.5.2.2 GetRootLevelResponse

The **GetRootLevelResponse** element specifies the result data for the **GetRootLevel** WSDL operation.

```
<xs:element name="GetRootLevelResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="GetRootLevelResult"
type="tns:DecompRenderResultRecord"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

**GetRootLevelResult:** A **DecompRenderResultRecord** value (section 2.2.4.5). In the event of an application error on the protocol server, this value will not occur.

### 3.1.4.5.3 Complex Types

The following table summarizes the XML schema complex type definitions that are specific to this operation.

Complex type	Description
<b>ArrayOfString</b>	Specifies an array of strings.
<b>DecompTupleRecord</b>	Names specific filters to be applied.

#### 3.1.4.5.3.1 DecompTupleRecord

**Namespace:** <http://www.microsoft.com/performancepoint/scorecards>

The **DecompTupleRecord** complex type names specific filters to be applied.

```
<xs:complexType name="DecompTupleRecord" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="RowSelections" type="tns:ArrayOfString"/>
    <xs:element minOccurs="0" maxOccurs="1" name="ColumnSelections"
type="tns:ArrayOfString"/>
    <xs:element minOccurs="0" maxOccurs="1" name="BackgroundMeasureName" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

**RowSelections:** MUST specify zero or more filter names. The names MUST be known to the protocol server. These named filters are applied to the response to scope it to a subset of results that match.

**ColumnSelections:** MUST specify zero or more filter names. The names MUST be known to the protocol server. These named filters are applied to the response to scope it to a subset of results that match.

**BackgroundMeasureName:** MAY be empty. Uniquely identifies a measure. This element MUST be known to the protocol server. Measures are the central values that are aggregated and analyzed.

### 3.1.4.5.3.2 ArrayOfString

**Namespace:** http://www.microsoft.com/performancepoint/scorecards

The **ArrayOfString** complex type specifies an array of strings.

```
<xs:complexType name="ArrayOfString" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="string" nillable="true"
      type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

**string:** This element MUST NOT be null and MUST conform to the XML data type string.

### 3.1.4.5.4 Simple Types

None.

### 3.1.4.5.5 Attributes

None.

### 3.1.4.5.6 Groups

None.

### 3.1.4.5.7 Attribute Groups

None.

## 3.1.5 Timer Events

None.

## 3.1.6 Other Local Events

None.

## 4 Protocol Examples

### 4.1 Starting a Session and Expanding to a Second Level

In this example, the protocol client starts a session with the protocol server, receives a node, then uses the default named action to retrieve additional nodes.

The named filter `[Product].[Product Categories].[All Products]` and measure name `[Measures].[Internet Order Count]` are already known to the protocol client from previous interactions with a data source being used by the implementation of the protocol server. The navigation context value `{b1150827-c56c-963f-a02c-b5a526402953}` is known to the protocol client from previous interactions with the implementation of the protocol server.

Using these values, the protocol client constructs the following **GetRootLevel** message:

```
<s:Envelope
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"><s:Body><GetRootLevel
xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.microsoft.com/performancepoint/scorecards"><navigationType>olap</
navigationType><navigationContextId>b1150827-c56c-963f-a02c-
b5a526402953</navigationContextId><tupleRecord><RowSelections><string>[Product].[P
roduct Categories].[All
Products]</string></RowSelections><ColumnSelections><string>[Measures].[Internet
Order
Count]</string></ColumnSelections></tupleRecord><viewState></viewState></GetRootLev
el></s:Body></s:Envelope>
```

The protocol server responds with the following:

```
<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"><soap:Body><GetRootLevelResponse
xmlns="http://www.microsoft.com/performancepoint/scorecards"><GetRootLevelResult><Levels><Dec
ompRenderResultLevel><SortedBy>DESCENDING</SortedBy><HierarchyName>[Product].[Product
Categories]</HierarchyName><TotalNodeCount>1</TotalNodeCount><UniqueName>[Product].[Product
Categories].[All]</UniqueName><Caption>(All)</Caption><Nodes><DecompRenderResultNode><Value
>27659</Value><FormattedValue>27,659</FormattedValue><HasChildren>true</HasChildren><UniqueName
>[Product].[Product Categories].[All Products]</UniqueName><Caption>All
Products</Caption><BottomSumValue>0</BottomSumValue><TopSumValue>0</TopSumValue></DecompRende
rResultNode></Nodes></DecompRenderResultLevel></Levels><MeasureCaption>Internet Order
Count</MeasureCaption><MeasureUniqueName>[Measures].[Internet Order
Count]</MeasureUniqueName><MeasureIsAdditive>false</MeasureIsAdditive><NavigationContext>{"Ty
pe":"olap","NavigationContext":"%7B430cc761-8437-7443-d7ea-aa397ef1b132%7D","Title":"Protocol
Sample
Report"}</NavigationContext></GetRootLevelResult></GetRootLevelResponse></soap:Body></soap:En
velope>
```

The protocol client renders this data for the user.

The protocol client constructs the following **AutoExpand** message using the values of the elements `DecompRenderResultLevel/UniqueName`, `DecompRenderResultLevel/Nodes[0]/DecompRenderResultNode/UniqueName`, and `DecompRenderResultLevel/NavigationContext` from the first response:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"><s:Body><AutoExpand
xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.microsoft.com/performancepoint/scorecards"><levelUniqueName>[Product].[Product
Categories].[All]</levelUniqueName><itemUniqueName>[Product].[Product Categories].[All
Products]</itemUniqueName><sortType>DESCENDING</sortType><navigationContext>{"Type":"olap","N
avigationContext":"%7B430cc761-8437-7443-d7ea-aa397ef1b132%7D","Title":"Protocol Sample
Report"}</navigationContext></AutoExpand></s:Body></s:Envelope>
```

The protocol server responds with the following:

```
<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"><soap:Body><AutoExpandResponse
xmlns="http://www.microsoft.com/performancepoint/scorecards"><AutoExpandResult><Levels><Decom
pRenderResultLevel><SortedBy>DESCENDING</SortedBy><HierarchyName>[Product].[Product
Categories]</HierarchyName><TotalNodeCount>3</TotalNodeCount><UniqueName>[Product].[Product
Categories].[Category]</UniqueName><Caption>Category</Caption><Nodes><DecompRenderResultNode>
<Value>18208</Value><FormattedValue>18,208</FormattedValue><HasChildren>true</HasChildren><Un
iqueName>[Product].[Product
Categories].[Category].&#amp;#39;4</UniqueName><Caption>Accessories</Caption><BottomSumValue>0</
BottomSumValue><TopSumValue>0</TopSumValue></DecompRenderResultNode><DecompRenderResultNode>
<Value>15205</Value><FormattedValue>15,205</FormattedValue><HasChildren>true</HasChildren><Uni
queName>[Product].[Product
Categories].[Category].&#amp;#39;1</UniqueName><Caption>Bikes</Caption><BottomSumValue>0</Bottom
SumValue><TopSumValue>0</TopSumValue></DecompRenderResultNode><DecompRenderResultNode><Value>
7461</Value><FormattedValue>7,461</FormattedValue><HasChildren>true</HasChildren><UniqueName>
[Product].[Product
Categories].[Category].&#amp;#39;3</UniqueName><Caption>Clothing</Caption><BottomSumValue>0</Bot
tomSumValue><TopSumValue>0</TopSumValue></DecompRenderResultNode></Nodes></DecompRenderResult
Level></Levels><MeasureCaption>Internet Order
Count</MeasureCaption><MeasureUniqueName>[Measures].[Internet Order
Count]</MeasureUniqueName><MeasureIsAdditive>>false</MeasureIsAdditive><NavigationContext>{"Ty
pe":"olap","NavigationContext":"%7B4ca709b5-7e78-9571-e7be-f6f89e3b0e6d%7D","Title":"Protocol
Sample
Report"}</NavigationContext></AutoExpandResult></AutoExpandResponse></soap:Body></soap:Envelo
pe>
```

The protocol client renders the response data for the user.

## 4.2 Retrieving a List of Named Actions and Expanding to a Third Level

In this scenario, the protocol client has previously started a session with the protocol server (Example 1.) It retrieves a list of named actions for one of the nodes from the second set of nodes retrieved in Example 1, then it retrieves a third set of nodes using a named action.

The protocol client constructs the following **CreateNavigationList** message using the values of the elements *DecompRenderResultLevel/UniqueName*, *DecompRenderResultLevel/Nodes[0]/DecompRenderResultNode/UniqueName*, and *DecompRenderResultLevel/NavigationContext* from the second protocol server response in Example 1:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"><s:Body><CreateNavigationList
xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.microsoft.com/performancepoint/scorecards"><levelUniqueName>[Product].[Product
Categories].[Category]</levelUniqueName><itemUniqueName>[Product].[Product
Categories].[Category].&#amp;#39;4</itemUniqueName><navigationContext>{"Type":"olap","Navigation
Context":"%7B4ca709b5-7e78-9571-e7be-f6f89e3b0e6d%7D","Title":"Protocol Sample
Report"}</navigationContext></CreateNavigationList></s:Body></s:Envelope>
```

The protocol server responds with the following:

```
<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"><soap:Body><CreateNavigationListResponse
xmlns="http://www.microsoft.com/performancepoint/scorecards"><CreateNavigationListResult><Gro
ups><DecompNavigationGroup><Name>[Product]</Name><Caption>Product</Caption><Actions><DecompNa
vigationAction><Name>[Product].[Product
Categories]. [Category]</Name><Caption>Category</Caption><ParentCaption>Categories</ParentCapt
ion><Enabled>>false</Enabled></DecompNavigationAction><DecompNavigationAction><Name>[Product].
[Product
Categories]. [Subcategory]</Name><Caption>Subcategory</Caption><ParentCaption>Categories</Pare
ntCaption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>[Prod
uct].[Product
Categories]. [Product]</Name><Caption>Product</Caption><ParentCaption>Categories</ParentCaptio
n><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>[Product].[Pr
oduct Model Lines]. [Product Line]</Name><Caption>Product Line</Caption><ParentCaption>Model
Lines</ParentCaption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction>
<Name>[Product].[Product Model
Lines]. [Model]</Name><Caption>Model</Caption><ParentCaption>Model
Lines</ParentCaption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction>
<Name>[Product].[Stock Level]. [Safety Stock Level]</Name><Caption>Safety Stock
Level</Caption><ParentCaption>Stock
Level</ParentCaption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction>
<Name>[Product].[Stock Level]. [Product]</Name><Caption>Product</Caption><ParentCaption>Stock
Level</ParentCaption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction>
<Name>[Product].[Class]. [Class]</Name><Caption>Class</Caption><Enabled>true</Enabled></Decomp
NavigationAction><DecompNavigationAction><Name>[Product].[Color]. [Color]</Name><Caption>Color
</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>[Prod
uct].[Days to Manufacture]. [Days to Manufacture]</Name><Caption>Days to
Manufacture</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction>
<Name>[Product].[Dealer Price]. [Dealer Price]</Name><Caption>Dealer
Price</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>
[Product].[End Date]. [End Date]</Name><Caption>End
Date</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>[
Product].[Large Photo]. [Large Photo]</Name><Caption>Large
Photo</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>
[Product].[List Price]. [List Price]</Name><Caption>List
Price</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>
[Product].[Model Name]. [Model Name]</Name><Caption>Model
Name</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>[
Product].[Reorder Point]. [Reorder Point]</Name><Caption>Reorder
Point</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>
[Product].[Size]. [Size]</Name><Caption>Size</Caption><Enabled>true</Enabled></DecompNavigatio
nAction><DecompNavigationAction><Name>[Product].[Size Range]. [Size Range]</Name><Caption>Size
Range</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>
[Product].[Standard Cost]. [Standard Cost]</Name><Caption>Standard
Cost</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>[
Product].[Start Date]. [Start Date]</Name><Caption>Start
Date</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>[
Product].[Status]. [Status]</Name><Caption>Status</Caption><Enabled>true</Enabled></DecompNavi
gationAction><DecompNavigationAction><Name>[Product].[Style]. [Style]</Name><Caption>Style</Ca
ption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>[Product]
.[Weight]. [Weight]</Name><Caption>Weight</Caption><Enabled>true</Enabled></DecompNavigationAc
tion></Actions></DecompNavigationGroup><DecompNavigationGroup><Name>[Promotion]</Name><Captio
n>Promotion</Caption><Actions><DecompNavigationAction><Name>[Promotion]. [Promotions]. [Categor
y]</Name><Caption>Category</Caption><ParentCaption>Promotions</ParentCaption><Enabled>true</E
nabled></DecompNavigationAction><DecompNavigationAction><Name>[Promotion]. [Promotions]. [Type]
</Name><Caption>Type</Caption><ParentCaption>Promotions</ParentCaption><Enabled>true</Enabled
></DecompNavigationAction><DecompNavigationAction><Name>[Promotion]. [Promotions]. [Promotion]
</Name><Caption>Promotion</Caption><ParentCaption>Promotions</ParentCaption><Enabled>true</Ena
bled></DecompNavigationAction><DecompNavigationAction><Name>[Promotion]. [Discount
Percent]. [Discount Percent]</Name><Caption>Discount
Percent</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>
[Promotion]. [End Date]. [End Date]</Name><Caption>End
Date</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>[
Promotion]. [Max Quantity]. [Max Quantity]</Name><Caption>Max
Quantity</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>
[Promotion]. [Min Quantity]. [Min Quantity]</Name><Caption>Min
```

```

Quantity</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>[Promotion].[Promotion Category].[Promotion Category]</Name><Caption>Category</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>[Promotion].[Promotion Type].[Promotion Type]</Name><Caption>Type</Caption><Enabled>true</Enabled></DecompNavigationAction><DecompNavigationAction><Name>[Promotion].[Start Date].[Start Date]</Name><Caption>StartDate</Caption><Enabled>true</Enabled></DecompNavigationAction></Actions></DecompNavigationGroup></Groups></CreateNavigationListResult></CreateNavigationListResponse></soap:Body></soap:Envelope>

```

The protocol client renders this data in a hierarchical user interface.

The user of the protocol client selects an action by its *DecompNavigationAction/Caption* value "Subcategory".

The protocol client constructs the following **Expand** message using the *DecompNavigationAction/Name* value of the action:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"><s:Body><Expand xmlns:i="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.microsoft.com/performancepoint/scorecards"><levelUniqueName>[Product].[Product Categories].[Category].&[4]</levelUniqueName><itemUniqueName>[Product].[Product Categories].[Subcategory]</navigationItemUniqueName><sortType>DESCENDING</sortType><navigationContext>{"Type": "olap", "NavigationContext": "%7B4ca709b5-7e78-9571-e7be-f6f89e3b0e6d%7D", "Title": "Protocol Sample Report"}</navigationContext></Expand></s:Body></s:Envelope>

```

The protocol server responds with the following:

```

<?xml version="1.0" encoding="utf-8"?><soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"><soap:Body><ExpandResponse xmlns="http://www.microsoft.com/performancepoint/scorecards"><ExpandResult><Levels><DecompRenderResultLevel><SortedBy>DESCENDING</SortedBy><HierarchyName>[Product].[Product Categories]</HierarchyName><TotalNodeCount>8</TotalNodeCount><UniqueName>[Product].[Product Categories].[Subcategory].&[37]</UniqueName><Caption>Tires and Tubes</Caption><BottomSumValue>0</BottomSumValue><TopSumValue>0</TopSumValue></DecompRenderResultNode><DecompRenderResultNode><Value>6440</Value><FormattedValue>6,440</FormattedValue><HasChildren>true</HasChildren><UniqueName>[Product].[Product Categories].[Subcategory].&[31]</UniqueName><Caption>Helmets</Caption><BottomSumValue>0</BottomSumValue><TopSumValue>0</TopSumValue></DecompRenderResultNode><DecompRenderResultNode><Value>4768</Value><FormattedValue>4,768</FormattedValue><HasChildren>true</HasChildren><UniqueName>[Product].[Product Categories].[Subcategory].&[28]</UniqueName><Caption>Bottles and Cages</Caption><BottomSumValue>0</BottomSumValue><TopSumValue>0</TopSumValue></DecompRenderResultNode><DecompRenderResultNode><Value>2121</Value><FormattedValue>2,121</FormattedValue><HasChildren>true</HasChildren><UniqueName>[Product].[Product Categories].[Subcategory].&[30]</UniqueName><Caption>Fenders</Caption><BottomSumValue>0</BottomSumValue><TopSumValue>0</TopSumValue></DecompRenderResultNode><DecompRenderResultNode><Value>908</Value><FormattedValue>908</FormattedValue><HasChildren>true</HasChildren><UniqueName>[Product].[Product Categories].[Subcategory].&[29]</UniqueName><Caption>Cleaners</Caption><BottomSumValue>0</BottomSumValue><TopSumValue>0</TopSumValue></DecompRenderResultNode><DecompRenderResultNode><Value>733</Value><FormattedValue>733</FormattedValue><HasChildren>true</HasChildren><UniqueName>[Product].[Product Categories].[Subcategory].&[32]</UniqueName><Caption>Hydration Packs</Caption><BottomSumValue>0</BottomSumValue><TopSumValue>0</TopSumValue></DecompRenderResultNode><DecompRenderResultNode><Value>328</Value><FormattedValue>328</FormattedValue><HasChildren>true</HasChildren><UniqueName>[Product].[Product Categories].[Subcategory].&[26]</UniqueName><Caption>Bike Racks</Caption><BottomSumValue>0</BottomSumValue><TopSumValue>0</TopSumValue></DecompRenderResultNode></Levels></ExpandResult></ExpandResponse></soap:Body></soap:Envelope>

```

```

sultNode><DecompRenderResultNode><Value>249</Value><FormattedValue>249</FormattedValue><HasChildren>true</HasChildren><UniqueName>[Product].[Product
Categories].[Subcategory].&[27]</UniqueName><Caption>Bike
Stands</Caption><BottomSumValue>0</BottomSumValue><TopSumValue>0</TopSumValue></DecompRenderR
esultNode></Nodes></DecompRenderResultLevel></Levels><MeasureCaption>Internet Order
Count</MeasureCaption><MeasureUniqueName>[Measures].[Internet Order
Count]</MeasureUniqueName><MeasureIsAdditive>>false</MeasureIsAdditive><NavigationContext>{"Type":
"olap","NavigationContext": "%7Bc10f24bc-3a2a-403a-c2e9-eab0313455d5%7D","Title": "Protocol
Sample
Report"}</NavigationContext></ExpandResult></ExpandResponse></soap:Body></soap:Envelope>

```

The protocol client renders the response data for the user.

### 4.3 Retrieving Properties

In this scenario, the protocol client has previously started a session with the protocol server (Example 1) and has retrieved an additional set of nodes (Example 2.) It retrieves a list of properties for the nodes retrieved in Example 2.

The protocol client constructs the following **GetMemberProperties** message, using the values of the elements *DecompRenderResultLevel/NavigationContext* and *DecompRenderResultLevel/Nodes[0]/DecompRenderResultNode/UniqueName* from the second protocol server response in Example 2:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"><s:Body><GetMemberProperties
xmlns:i="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.microsoft.com/performancepoint/scorecards"><navigationContext>{"Type": "olap
", "NavigationContext": "%7Bc10f24bc-3a2a-403a-c2e9-eab0313455d5%7D", "Title": "Protocol Sample
Report"}</navigationContext><itemUniqueName>[Product].[Product
Categories].[Subcategory].&[37]</itemUniqueName></GetMemberProperties></s:Body></s:Envelo
pe>

```

The protocol server responds with the following:

```

<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"><soap:Body><GetMemberPropertiesResponse
xmlns="http://www.microsoft.com/performancepoint/scorecards"><GetMemberPropertiesResult><Memb
erName>[Product].[Product
Categories].[Subcategory].&[37]</MemberName><Properties><DecompMemberProperty><Name>Categ
ory</Name><Caption>Category</Caption><Value>Accessories</Value></DecompMemberProperty></Prope
rties></GetMemberPropertiesResult></GetMemberPropertiesResponse></soap:Body></soap:Envelope>

```

The protocol client renders the response data for the user.

## 5 Security

### 5.1 Security Considerations for Implementers

It is possible to use **Secure Sockets Layer (SSL)** to securely implement this protocol. The protocol exposes data sources to potential tampering or data disclosure without SSL.

The security permissions required for all operations are inherited from the parent **list** item.

### 5.2 Index of Security Parameters

None.

Preliminary



## 6 Appendix A: Full WSDL

For ease of implementation, the full WSDL is provided in this appendix.

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:tns="http://www.microsoft.com/performancepoint/scorecards"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
targetNamespace="http://www.microsoft.com/performancepoint/scorecards"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types>
    <xs:schema elementFormDefault="qualified"
targetNamespace="http://www.microsoft.com/performancepoint/scorecards">
      <xs:element name="AutoExpand">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="levelUniqueName" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="itemUniqueName" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="sortType" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="navigationContext"
type="xs:string"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="AutoExpandResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="AutoExpandResult"
type="tns:DecompRenderResultRecord"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:complexType name="DecompRenderResultRecord">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="1" name="Levels"
type="tns:ArrayOfDecompRenderResultLevel"/>
          <xs:element minOccurs="0" maxOccurs="1" name="MeasureCaption" type="xs:string"/>
          <xs:element minOccurs="0" maxOccurs="1" name="MeasureUniqueName" type="xs:string"/>
          <xs:element minOccurs="1" maxOccurs="1" name="MeasureIsAdditive"
type="xs:boolean"/>
          <xs:element minOccurs="0" maxOccurs="1" name="NavigationContext" type="xs:string"/>
          <xs:element minOccurs="0" maxOccurs="1" name="ErrorMsg" type="xs:string"/>
          <xs:element minOccurs="0" maxOccurs="1" name="ErrorDetails" type="xs:string"/>
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="ArrayOfDecompRenderResultLevel">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="unbounded" name="DecompRenderResultLevel"
nillable="true" type="tns:DecompRenderResultLevel"/>
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="DecompRenderResultLevel">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="1" name="SortedBy" type="xs:string"/>
          <xs:element minOccurs="0" maxOccurs="1" name="HierarchyName" type="xs:string"/>
          <xs:element minOccurs="1" maxOccurs="1" name="TotalNodeCount" type="xs:int"/>
          <xs:element minOccurs="0" maxOccurs="1" name="UniqueName" type="xs:string"/>
          <xs:element minOccurs="0" maxOccurs="1" name="Caption" type="xs:string"/>
          <xs:element minOccurs="0" maxOccurs="1" name="Nodes"
type="tns:ArrayOfDecompRenderResultNode"/>
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="ArrayOfDecompRenderResultNode">
        <xs:sequence>
```

```

        <xs:element minOccurs="0" maxOccurs="unbounded" name="DecompRenderResultNode"
nillable="true" type="tns:DecompRenderResultNode"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="DecompRenderResultNode">
    <xs:sequence>
        <xs:element minOccurs="1" maxOccurs="1" name="Value" type="xs:double"/>
        <xs:element minOccurs="0" maxOccurs="1" name="FormattedValue" type="xs:string"/>
        <xs:element minOccurs="1" maxOccurs="1" name="HasChildren" type="xs:boolean"/>
        <xs:element minOccurs="0" maxOccurs="1" name="UniqueName" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="Caption" type="xs:string"/>
        <xs:element minOccurs="1" maxOccurs="1" name="BottomSumValue" type="xs:double"/>
        <xs:element minOccurs="0" maxOccurs="1" name="BottomSumFormattedValue"
type="xs:string"/>
        <xs:element minOccurs="1" maxOccurs="1" name="TopSumValue" type="xs:double"/>
        <xs:element minOccurs="0" maxOccurs="1" name="TopSumFormattedValue"
type="xs:string"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="Expand">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="levelUniqueName" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="itemUniqueName" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="navigationItemUniqueName"
type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="sortType" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="navigationContext"
type="xs:string"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="ExpandResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="ExpandResult"
type="tns:DecompRenderResultRecord"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="CreateNavigationList">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="levelUniqueName" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="itemUniqueName" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="navigationContext"
type="xs:string"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="CreateNavigationListResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="CreateNavigationListResult"
type="tns:DecompNavigationList"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:complexType name="DecompNavigationList">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Groups"
type="tns:ArrayOfDecompNavigationGroup"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfDecompNavigationGroup">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="DecompNavigationGroup"
nillable="true" type="tns:DecompNavigationGroup"/>
    </xs:sequence>
</xs:complexType>

```

```

    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="DecompNavigationGroup">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="Caption" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="Actions"
type="tns:ArrayOfDecompNavigationAction"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ArrayOfDecompNavigationAction">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="DecompNavigationAction"
nillable="true" type="tns:DecompNavigationAction"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="DecompNavigationAction">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="Caption" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="ParentCaption" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="Enabled" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="GetMemberProperties">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="navigationContext"
type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="itemUniqueName" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetMemberPropertiesResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="GetMemberPropertiesResult"
type="tns:DecompMemberProperties"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="DecompMemberProperties">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="MemberName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="Properties"
type="tns:ArrayOfDecompMemberProperty"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ArrayOfDecompMemberProperty">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="DecompMemberProperty"
nillable="true" type="tns:DecompMemberProperty"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="DecompMemberProperty">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="Caption" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="Value" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="GetRootLevel">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="navigationType" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="navigationContextId"
type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="tupleRecord"
type="tns:DecompTupleRecord"/>

```

```

        <xs:element minOccurs="0" maxOccurs="1" name="viewState" type="xs:string"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:complexType name="DecompTupleRecord">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="RowSelections"
type="tns:ArrayOfString"/>
        <xs:element minOccurs="0" maxOccurs="1" name="ColumnSelections"
type="tns:ArrayOfString"/>
        <xs:element minOccurs="0" maxOccurs="1" name="BackgroundMeasureName"
type="xs:string"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfString">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="string" nillable="true"
type="xs:string"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="GetRootLevelResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="GetRootLevelResult"
type="tns:DecompRenderResultRecord"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
</xs:schema>
</wsdl:types>
<wsdl:portType name="PPSDecompRenderingServiceSoap">
    <wsdl:operation name="AutoExpand">
        <wsdl:input message="tns:AutoExpandSoapIn"/>
        <wsdl:output message="tns:AutoExpandSoapOut"/>
    </wsdl:operation>
    <wsdl:operation name="Expand">
        <wsdl:input message="tns:ExpandSoapIn"/>
        <wsdl:output message="tns:ExpandSoapOut"/>
    </wsdl:operation>
    <wsdl:operation name="CreateNavigationList">
        <wsdl:input message="tns:CreateNavigationListSoapIn"/>
        <wsdl:output message="tns:CreateNavigationListSoapOut"/>
    </wsdl:operation>
    <wsdl:operation name="GetMemberProperties">
        <wsdl:input message="tns:GetMemberPropertiesSoapIn"/>
        <wsdl:output message="tns:GetMemberPropertiesSoapOut"/>
    </wsdl:operation>
    <wsdl:operation name="GetRootLevel">
        <wsdl:input message="tns:GetRootLevelSoapIn"/>
        <wsdl:output message="tns:GetRootLevelSoapOut"/>
    </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="PPSDecompRenderingServiceSoap"
type="tns:PPSDecompRenderingServiceSoap">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="AutoExpand">
        <soap:operation
soapAction="http://www.microsoft.com/performancepoint/scorecards/AutoExpand"
style="document"/>
        <wsdl:input>
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="Expand">
        <soap:operation
soapAction="http://www.microsoft.com/performancepoint/scorecards/Expand" style="document"/>

```

```

        <wsdl:input>
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="CreateNavigationList">
        <soap:operation
soapAction="http://www.microsoft.com/performancepoint/scorecards/CreateNavigationList"
style="document"/>
        <wsdl:input>
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetMemberProperties">
        <soap:operation
soapAction="http://www.microsoft.com/performancepoint/scorecards/GetMemberProperties"
style="document"/>
        <wsdl:input>
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetRootLevel">
        <soap:operation
soapAction="http://www.microsoft.com/performancepoint/scorecards/GetRootLevel"
style="document"/>
        <wsdl:input>
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
</wsdl:binding>
<wsdl:binding name="PPSDecompRenderingServiceSoap12"
type="tns:PPSDecompRenderingServiceSoap">
    <soap12:binding transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="AutoExpand">
        <soap12:operation
soapAction="http://www.microsoft.com/performancepoint/scorecards/AutoExpand"
style="document"/>
        <wsdl:input>
            <soap12:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="Expand">
        <soap12:operation
soapAction="http://www.microsoft.com/performancepoint/scorecards/Expand" style="document"/>
        <wsdl:input>
            <soap12:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="CreateNavigationList">
        <soap12:operation
soapAction="http://www.microsoft.com/performancepoint/scorecards/CreateNavigationList"
style="document"/>

```

```

    <wsdl:input>
      <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetMemberProperties">
    <soap12:operation
soapAction="http://www.microsoft.com/performancepoint/scorecards/GetMemberProperties"
style="document"/>
    <wsdl:input>
      <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetRootLevel">
    <soap12:operation
soapAction="http://www.microsoft.com/performancepoint/scorecards/GetRootLevel"
style="document"/>
    <wsdl:input>
      <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
</wsdl:binding>
<wsdl:message name="AutoExpandSoapIn">
  <wsdl:part name="parameters" element="tns:AutoExpand"/>
</wsdl:message>
<wsdl:message name="AutoExpandSoapOut">
  <wsdl:part name="parameters" element="tns:AutoExpandResponse"/>
</wsdl:message>
<wsdl:message name="CreateNavigationListSoapIn">
  <wsdl:part name="parameters" element="tns:CreateNavigationList"/>
</wsdl:message>
<wsdl:message name="CreateNavigationListSoapOut">
  <wsdl:part name="parameters" element="tns:CreateNavigationListResponse"/>
</wsdl:message>
<wsdl:message name="ExpandSoapIn">
  <wsdl:part name="parameters" element="tns:Expand"/>
</wsdl:message>
<wsdl:message name="ExpandSoapOut">
  <wsdl:part name="parameters" element="tns:ExpandResponse"/>
</wsdl:message>
<wsdl:message name="GetMemberPropertiesSoapIn">
  <wsdl:part name="parameters" element="tns:GetMemberProperties"/>
</wsdl:message>
<wsdl:message name="GetMemberPropertiesSoapOut">
  <wsdl:part name="parameters" element="tns:GetMemberPropertiesResponse"/>
</wsdl:message>
<wsdl:message name="GetRootLevelSoapIn">
  <wsdl:part name="parameters" element="tns:GetRootLevel"/>
</wsdl:message>
<wsdl:message name="GetRootLevelSoapOut">
  <wsdl:part name="parameters" element="tns:GetRootLevelResponse"/>
</wsdl:message>
</wsdl:definitions>

```

## 7 Appendix B: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs.

- Microsoft SharePoint Server 2010
- Microsoft SharePoint Server 2013
- Microsoft SharePoint Server 2016 Preview

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms SHOULD or SHOULD NOT implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that the product does not follow the prescription.

Preliminary

## 8 Change Tracking

This section identifies changes that were made to this document since the last release. Changes are classified as New, Major, Minor, Editorial, or No change.

The revision class **New** means that a new document is being released.

The revision class **Major** means that the technical content in the document was significantly revised. Major changes affect protocol interoperability or implementation. Examples of major changes are:

- A document revision that incorporates changes to interoperability requirements or functionality.
- The removal of a document from the documentation set.

The revision class **Minor** means that the meaning of the technical content was clarified. Minor changes do not affect protocol interoperability or implementation. Examples of minor changes are updates to clarify ambiguity at the sentence, paragraph, or table level.

The revision class **Editorial** means that the formatting in the technical content was changed. Editorial changes apply to grammatical, formatting, and style issues.

The revision class **No change** means that no new technical changes were introduced. Minor editorial and formatting changes may have been made, but the technical content of the document is identical to the last released version.

Major and minor changes can be described further using the following change types:

- New content added.
- Content updated.
- Content removed.
- New product behavior note added.
- Product behavior note updated.
- Product behavior note removed.
- New protocol syntax added.
- Protocol syntax updated.
- Protocol syntax removed.
- New content added due to protocol revision.
- Content updated due to protocol revision.
- Content removed due to protocol revision.
- New protocol syntax added due to protocol revision.
- Protocol syntax updated due to protocol revision.
- Protocol syntax removed due to protocol revision.
- Obsolete document removed.

Editorial changes are always classified with the change type **Editorially updated**.

Some important terms used in the change type descriptions are defined as follows:



- **Protocol syntax** refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.
- **Protocol revision** refers to changes made to a protocol that affect the bits that are sent over the wire.

The changes made to this document are listed in the following table. For more information, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com).

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
<a href="#">Z</a> Appendix B: Product Behavior	Updated list of supported products.	Y	Content updated due to protocol revision.

Preliminary

## 9 Index

### A

Abstract data model  
[server](#) 18  
[Applicability](#) 11  
[ArrayOfDecompRenderResultLevel complex type](#) 13  
[ArrayOfDecompRenderResultNode complex type](#) 13  
[Attribute groups](#) 16  
[Attributes](#) 16  
AutoExpand operation ([section 3.1.4](#) 18, [section 3.1.4.1](#) 19)

### C

[Capability negotiation](#) 11  
[Change tracking](#) 48  
Client  
[overview](#) 17  
[Common data structures](#) 16  
[Complex types](#) 13  
[ArrayOfDecompRenderResultLevel](#) 13  
[ArrayOfDecompRenderResultNode](#) 13  
[DecompRenderResultLevel](#) 14  
[DecompRenderResultNode](#) 14  
[DecompRenderResultRecord](#) 15  
CreateNavigationList operation ([section 3.1.4](#) 18, [section 3.1.4.2](#) 21)

### D

Data model - abstract  
[server](#) 18  
[DecompRenderResultLevel complex type](#) 14  
[DecompRenderResultNode complex type](#) 14  
[DecompRenderResultRecord complex type](#) 15

### E

Events  
[local - server](#) 34  
[timer - server](#) 34  
Examples  
[overview](#) 35  
[retrieving a list of named actions and expanding to a third level](#) 36  
[retrieving properties](#) 39  
[starting a session and expanding to a second level](#) 35  
Expand operation ([section 3.1.4](#) 18, [section 3.1.4.3](#) 25)

### F

[Fields - vendor-extensible](#) 11  
[Full WSDL](#) 41

### G

GetMemberProperties operation ([section 3.1.4](#) 18, [section 3.1.4.4](#) 28)

GetRootLevel operation ([section 3.1.4](#) 18, [section 3.1.4.5](#) 31)  
[Glossary](#) 8  
[Groups](#) 16

### I

[Implementer - security considerations](#) 40  
[Index of security parameters](#) 40  
[Informative references](#) 10  
Initialization  
[server](#) 18  
[Introduction](#) 8

### L

Local events  
[server](#) 34

### M

Message processing  
[server](#) 18  
Messages  
[ArrayOfDecompRenderResultLevel complex type](#) 13  
[ArrayOfDecompRenderResultNode complex type](#) 13  
[attribute groups](#) 16  
[attributes](#) 16  
[common data structures](#) 16  
[complex types](#) 13  
[DecompRenderResultLevel complex type](#) 14  
[DecompRenderResultNode complex type](#) 14  
[DecompRenderResultRecord complex type](#) 15  
[elements](#) 13  
[enumerated](#) 12  
[groups](#) 16  
[namespaces](#) 12  
[simple types](#) 16  
[syntax](#) 12  
[transport](#) 12

### N

[Namespaces](#) 12  
[Normative references](#) 9

### O

Operations  
[AutoExpand](#) 19  
[CreateNavigationList](#) 21  
[Expand](#) 25  
[GetMemberProperties](#) 28  
[GetRootLevel](#) 31  
[Overview \(synopsis\)](#) 10

### P

[Parameters - security index](#) 40  
[Preconditions](#) 11

[Prerequisites](#) 11  
[Product behavior](#) 47  
Protocol Details  
[overview](#) 17

## R

[References](#) 9  
[informative](#) 10  
[normative](#) 9  
[Relationship to other protocols](#) 10  
[Retrieving a list of named actions and expanding to a third level example](#) 36  
[Retrieving properties example](#) 39

## S

Security  
[implementer considerations](#) 40  
[parameter index](#) 40  
Sequencing rules  
[server](#) 18  
Server  
[abstract data model](#) 18  
[AutoExpand operation](#) 19  
[CreateNavigationList operation](#) 21  
[details](#) 17  
[Expand operation](#) 25  
[GetMemberProperties operation](#) 28  
[GetRootLevel operation](#) 31  
[initialization](#) 18  
[local events](#) 34  
[message processing](#) 18  
[overview](#) 17  
[sequencing rules](#) 18  
[timer events](#) 34  
[timers](#) 18  
[Simple types](#) 16  
[Standards assignments](#) 11  
[Starting a session and expanding to a second level example](#) 35  
Syntax  
[messages - overview](#) 12

## T

Timer events  
[server](#) 34  
Timers  
[server](#) 18  
[Tracking changes](#) 48  
[Transport](#) 12  
Types  
[complex](#) 13  
[simple](#) 16

## V

[Vendor-extensible fields](#) 11  
[Versioning](#) 11

## W

[WSDL](#) 41