

## [MS-WSDS-Diff]:

# WS-Enumeration: Directory Services Protocol Extensions

---

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation (“this documentation”) for protocols, file formats, data portability, computer languages, and standards support. Additionally, overview documents cover inter-protocol relationships and interactions.
- **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 can make copies of it in order to develop implementations of the technologies that are described in this documentation and can distribute portions of it in your implementations that use these technologies or in your documentation as necessary to properly document the implementation. You can also distribute in your implementation, with or without modification, any schemas, IDLs, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications documentation.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that might cover your implementations of the technologies described in the Open Specifications documentation. Neither this notice nor Microsoft's delivery of this documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open Specifications document might be covered by the Microsoft [Open Specifications Promise](#) or the [Microsoft Community Promise](#). If you would prefer a written license, or if the technologies described in this documentation 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).
- **License Programs.** To see all of the protocols in scope under a specific license program and the associated patents, visit the [Patent Map](#).
- **Trademarks.** The names of companies and products contained in this documentation might 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, email addresses, logos, people, places, and events that are 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 as specifically described above, whether by implication, estoppel, or otherwise.

**Tools.** The Open Specifications documentation does 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 documents are intended for use in conjunction with publicly available standards specifications and network programming art and, as such, assume that the reader either is familiar with the aforementioned material or has immediate access to it.

**Support.** For questions and support, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com).

## Revision Summary

Date	Revision History	Revision Class	Comments
12/5/2008	0.1	Major	Initial Availability
1/16/2009	0.1.1	Editorial	Changed language and formatting in the technical content.
2/27/2009	0.1.2	Editorial	Changed language and formatting in the technical content.
4/10/2009	1.0	Major	Updated and revised the technical content.
5/22/2009	2.0	Major	Updated and revised the technical content.
7/2/2009	3.0	Major	Updated and revised the technical content.
8/14/2009	4.0	Major	Updated and revised the technical content.
9/25/2009	5.0	Major	Updated and revised the technical content.
11/6/2009	5.1	Minor	Clarified the meaning of the technical content.
12/18/2009	5.2	Minor	Clarified the meaning of the technical content.
1/29/2010	5.3	Minor	Clarified the meaning of the technical content.
3/12/2010	5.3.1	Editorial	Changed language and formatting in the technical content.
4/23/2010	6.0	Major	Updated and revised the technical content.
6/4/2010	6.0.1	Editorial	Changed language and formatting in the technical content.
7/16/2010	6.0.1	None	No changes to the meaning, language, or formatting of the technical content.
8/27/2010	6.0.1	None	No changes to the meaning, language, or formatting of the technical content.
10/8/2010	6.0.1	None	No changes to the meaning, language, or formatting of the technical content.
11/19/2010	6.1	Minor	Clarified the meaning of the technical content.
1/7/2011	6.1	None	No changes to the meaning, language, or formatting of the technical content.
2/11/2011	6.1	None	No changes to the meaning, language, or formatting of the technical content.
3/25/2011	6.1	None	No changes to the meaning, language, or formatting of the technical content.
5/6/2011	6.1	None	No changes to the meaning, language, or formatting of the technical content.
6/17/2011	6.2	Minor	Clarified the meaning of the technical content.
9/23/2011	6.2	None	No changes to the meaning, language, or formatting of the technical content.
12/16/2011	7.0	Major	Updated and revised the technical content.
3/30/2012	7.0	None	No changes to the meaning, language, or formatting of the

<b>Date</b>	<b>Revision History</b>	<b>Revision Class</b>	<b>Comments</b>
			technical content.
7/12/2012	7.0	None	No changes to the meaning, language, or formatting of the technical content.
10/25/2012	7.0	None	No changes to the meaning, language, or formatting of the technical content.
1/31/2013	7.0	None	No changes to the meaning, language, or formatting of the technical content.
8/8/2013	8.0	Major	Updated and revised the technical content.
11/14/2013	8.0	None	No changes to the meaning, language, or formatting of the technical content.
2/13/2014	8.0	None	No changes to the meaning, language, or formatting of the technical content.
5/15/2014	8.0	None	No changes to the meaning, language, or formatting of the technical content.
6/30/2015	9.0	Major	Significantly changed the technical content.
10/16/2015	9.0	None	No changes to the meaning, language, or formatting of the technical content.
7/14/2016	9.0	None	No changes to the meaning, language, or formatting of the technical content.
3/16/2017	10.0	Major	Significantly changed the technical content.
6/1/2017	11.0	Major	Significantly changed the technical content.
<u>9/15/2017</u>	<u>12.0</u>	<u>Major</u>	<u>Significantly changed the technical content.</u>

# Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>6</b>
1.1	Glossary .....	6
1.2	References .....	7
1.2.1	Normative References .....	7
1.2.2	Informative References .....	8
1.3	Overview .....	9
1.4	Relationship to Other Protocols .....	9
1.5	Prerequisites/Preconditions .....	9
1.6	Applicability Statement .....	9
1.7	Versioning and Capability Negotiation .....	10
1.8	Vendor-Extensible Fields .....	10
1.9	Standards Assignments.....	10
<b>2</b>	<b>Messages.....</b>	<b>11</b>
2.1	Transport .....	11
2.2	Common Message Syntax .....	11
2.2.1	Namespaces .....	11
2.2.2	Messages.....	11
2.2.3	Elements .....	12
2.2.4	Complex Types.....	12
2.2.5	Simple Types .....	12
2.2.6	Attributes .....	12
2.2.7	Groups .....	12
2.2.8	Attribute Groups.....	12
2.3	Directory Service Schema Elements .....	12
<b>3</b>	<b>Protocol Details .....</b>	<b>13</b>
3.1	Enumeration Server Details .....	13
3.1.1	Abstract Data Model.....	13
3.1.2	Timers .....	14
3.1.3	Initialization.....	14
3.1.4	Message Processing Events and Sequencing Rules .....	14
3.1.4.1	wsen:Enumerate.....	15
3.1.4.1.1	Elements .....	16
3.1.4.1.1.1	adlq:LdapQuery .....	16
3.1.4.1.1.1.1	adlq:filter .....	17
3.1.4.1.1.1.2	adlq:BaseObject .....	17
3.1.4.1.1.1.3	adlq:Scope .....	17
3.1.4.1.1.2	ad:Selection .....	18
3.1.4.1.1.2.1	ad:SelectionProperty.....	18
3.1.4.1.1.3	ad:Sorting .....	19
3.1.4.1.1.3.1	ad:SortingProperty .....	19
3.1.4.1.2	Attributes .....	20
3.1.4.1.2.1	ad:Selection/@Dialect .....	20
3.1.4.1.2.2	ad:Sorting/@Dialect .....	20
3.1.4.1.2.3	ad:Sorting/ad:SortingProperty/@Ascending .....	20
3.1.4.1.3	SOAP Faults.....	20
3.1.4.1.3.1	ad:EnumerationContextLimitExceeded .....	20
3.1.4.1.3.2	ad:UnsupportedSelectOrSortDialectFault.....	21
3.1.4.1.3.3	ad:InvalidPropertyFault .....	21
3.1.4.1.3.4	ad:InvalidSortKey .....	22
3.1.4.1.3.5	wsen:CannotProcessFilter .....	22
3.1.4.1.3.6	wsa2004:EndPointUnavailable .....	22
3.1.4.2	wsen:Pull .....	23
3.1.4.2.1	SOAP Faults.....	24

3.1.4.2.1.1	ad:MaxCharsNotSupported.....	24
3.1.4.2.1.2	wsen:InvalidEnumerationContext.....	24
3.1.4.2.1.3	wsa2004:DestinationUnreachable .....	25
3.1.4.2.1.4	wsa2004:EndpointUnavailable .....	25
3.1.4.2.1.5	ad:MaxTimeExceedsLimit .....	25
3.1.4.3	wsen:Renew.....	26
3.1.4.3.1	SOAP faults .....	26
3.1.4.3.1.1	wsen:InvalidEnumerationContext.....	26
3.1.4.3.1.2	wsa2004:EndpointUnavailable .....	26
3.1.4.4	wsen:GetStatus .....	27
3.1.4.4.1	SOAP Faults.....	27
3.1.4.4.1.1	wsen:InvalidEnumerationContext.....	27
3.1.4.4.1.2	wsa2004:EndpointUnavailable .....	27
3.1.4.5	wsen:Release .....	28
3.1.4.5.1	SOAP Faults.....	28
3.1.4.5.1.1	wsa2004:EndpointUnavailable .....	28
3.1.5	Timer Events.....	28
3.1.6	Other Local Events.....	28
<b>4</b>	<b>Protocol Examples.....</b>	<b>29</b>
4.1	WS-Enumerate Directory Services Extension "Enumerate" Request Example .....	29
4.2	WS-Enumerate Directory Services Extension "Enumerate" Response Example .....	30
4.3	WS-Enumerate Directory Services Extension "Pull" Request Example.....	30
4.4	WS-Enumerate Directory Services Extension "Pull" Response Example.....	31
4.5	WS-Enumerate Directory Services Extension "FaultDetail" Example.....	32
<b>5</b>	<b>Security.....</b>	<b>34</b>
5.1	Security Considerations for Implementers .....	34
5.2	Index of Security Parameters .....	34
<b>6</b>	<b>Appendix A: WSDL.....</b>	<b>35</b>
<b>7</b>	<b>Appendix B: Schema.....</b>	<b>36</b>
<b>8</b>	<b>Appendix C: Product Behavior .....</b>	<b>39</b>
<b>9</b>	<b>Change Tracking.....</b>	<b>42</b>
<b>10</b>	<b>Index.....</b>	<b>43</b>

# 1 Introduction

The WS-Enumeration Directory Services Protocol Extensions are a set of extensions to the Web Services Enumeration (WS-Enumeration) [WSENUM] protocol for facilitating SOAP-based search operations against directory servers. This protocol makes it easy for client applications that currently use non-Web services protocols, such as Lightweight Directory Access Protocol (LDAP) version 3 [RFC2251], to instead use Web service protocols for such operations.

The extensions to the SOAP-based Enumeration protocol specify dialect for expressing the search filter for an enumeration. It also provide a means of requesting and receiving selected fragments of resultant objects in the context of a specific enumeration and an additional set of SOAP faults for various WS-Enumeration [WSENUM] operations.

Sections 1.5, 1.8, 1.9, 2, and 3 of this specification are normative. All other sections and examples in this specification are informative.

## 1.1 Glossary

This document uses the following terms:

**Active Directory Web Services (ADWS):** Provides a web service interface to Active Directory Domain Services (AD DS) and Active Directory Lightweight Directory Services (AD LDS) instances.

**constructed attribute:** An attribute whose values are computed from normal attributes (for read) and/or have effects on the values of normal attributes (for write).

**default attribute:** An attribute of an object that is not a constructed attribute.

**directory attribute:** An identifier for a single-valued or multi-valued data element that is associated with a directory object.

**directory object:** A Lightweight Directory Access Protocol (LDAP) object, as specified in [RFC2251], that is a specialization of an object.

**endpoint:** In the context of a web service, a network target to which a SOAP message can be addressed. See [WSADDR].

**enumeration context:** A session context that represents a specific traversal through a logical sequence of XML element information items using the Pull operation defined in WS-Enumeration specification. See [WSENUM].

**globally unique identifier (GUID):** A term used interchangeably with universally unique identifier (UUID) in Microsoft protocol technical documents (TDs). Interchanging the usage of these terms does not imply or require a specific algorithm or mechanism to generate the value. Specifically, the use of this term does not imply or require that the algorithms described in [RFC4122] or [C706] must be used for generating the GUID. See also universally unique identifier (UUID).

**Lightweight Directory Access Protocol (LDAP):** The primary access protocol for Active Directory. Lightweight Directory Access Protocol (LDAP) is an industry-standard protocol, established by the Internet Engineering Task Force (IETF), which allows users to query and update information in a directory service (DS), as described in [MS-ADTS]. The Lightweight Directory Access Protocol can be either version 2 [RFC1777] or version 3 [RFC3377].

**object reference property:** In Active Directory Web Services, this is the property that uniquely identifies a directory object. It can be expressed as either a GUID or as a distinguished name.

**requestor:** The client application that is requesting the specific objects from the Web Service.

**schema:** The set of attributes and object classes that govern the creation and update of objects.

**security principal:** A unique entity that is identifiable through cryptographic means by at least one key. It frequently corresponds to a human user, but also can be a service that offers a resource to other security principals. Also referred to as principal.

**session:** An authenticated communication channel between the client and server correlating a group of messages into a conversation.

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

**SOAP message:** An XML document consisting of a mandatory SOAP envelope, an optional SOAP header, and a mandatory SOAP body. See [SOAP1.2-1/2007] section 5 for more information.

**Uniform Resource Identifier (URI):** A string that identifies a resource. The URI is an addressing mechanism defined in Internet Engineering Task Force (IETF) Uniform Resource Identifier (URI): Generic Syntax [RFC3986].

**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 port type:** A named set of logically-related, abstract Web Services Description Language (WSDL) operations and messages.

**XML:** The Extensible Markup Language, as described in [XML1.0].

**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 Schema (XSD):** A language that defines the elements, attributes, namespaces, and data types for XML documents as defined by [XMLSCHEMA1/2] and [W3C-XSD] standards. An XML schema uses XML syntax for its language.

**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. We will assist you in finding the relevant information.

- [MS-ADDM] Microsoft Corporation, "Active Directory Web Services: Data Model and Common Elements".
- [MS-ADTS] Microsoft Corporation, "Active Directory Technical Specification".
- [MS-WSDS] Microsoft Corporation, "WS-Enumeration: Directory Services Protocol Extensions".
- [MS-WPELD] Microsoft Corporation, "WS-Transfer and WS-Enumeration Protocol Extension for Lightweight Directory Access Protocol v3 Controls".
- [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>
- [RFC2251] Wahl, M., Howes, T., and Kille, S., "Lightweight Directory Access Protocol (v3)", RFC 2251, December 1997, <http://www.ietf.org/rfc/rfc2251.txt>
- [RFC2254] Howes, T., "The String Representation of LDAP Search Filters", RFC 2254, December 1997, <http://www.ietf.org/rfc/rfc2254.txt>
- [SOAP1.1] Box, D., Ehnebuske, D., Kakivaya, G., et al., "Simple Object Access Protocol (SOAP) 1.1", W3C Note, May 2000, <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>
- [SOAP1.2-1/2003] Gudgin, M., Hadley, M., Mendelsohn, N., et al., "SOAP Version 1.2 Part 1: Messaging Framework", W3C Recommendation, June 2003, <http://www.w3.org/TR/2003/REC-soap12-part1-20030624>
- [WSAddressing] Box, D., et al., "Web Services Addressing (WS-Addressing)", August 2004, <http://www.w3.org/Submission/ws-addressing/>
- [WSADDR] Gudgin, M., Hadley, M., and Rogers, T., "Web Services Addressing (WS-Addressing) 1.0", W3C Recommendation, May 2006, <http://www.w3.org/2005/08/addressing>
- [WSASB] Gudgin, M., Hadley, M., and Rogers, T., Eds., "Web Services Addressing 1.0 - SOAP Binding", W3C Recommendation, May 2006, <http://www.w3.org/TR/2006/REC-ws-addr-soap-20060509/>
- [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>
- [WSENUM] Alexander, J., Box, D., Cabrera, L.F., et al., "Web Services Enumeration (WS-Enumeration)", March 2006, <http://www.w3.org/Submission/2006/SUBM-WS-Enumeration-20060315/>
- [XMLNS-2ED] World Wide Web Consortium, "Namespaces in XML 1.0 (Second Edition)", August 2006, <http://www.w3.org/TR/2006/REC-xml-names-20060816/>
- [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

- [MC-NMF] Microsoft Corporation, ".NET Message Framing Protocol".
- [MSFT-RSAT] Microsoft Corporation, "Remote Server Administration Tools (RSAT) for Windows operating systems", <https://support.microsoft.com/en-us/kb/2693643>



### 1.3 Overview

The WS-Enumeration [WSENUM] specification describes how query operations performed against the directory server in the form of SOAP messages are initiated using the Enumerate operation. The Enumerate operation creates a new enumeration context for subsequent traversal/retrieval of result items by means of the Pull operation. [MS-WSDS] specifies the query filter language allowed for such enumeration operations.

WS-Enumeration [WSENUM] does not provide an explicit means to specify which fragments of the object to return for a certain enumeration in the Enumerate request. Since directory services like Active Directory Web Services (ADWS) use the filter expression to just specify which objects to return, and not what portions of those objects to return, this protocol specifies extensions to the WS-Enumeration [WSENUM] Enumerate operation, providing a means of requesting selected fragments out of the resultant objects. The extensions also include a way to specify which directory attribute the sorting of the resultant objects or their fragments is based on and whether or not the order is ascending.

The specification of WS-Enumeration [WSENUM] provides only a small set of SOAP faults for a directory server to return. This set is insufficient for many error conditions that a server could need to report to the client, which forces the server to use its own nonstandard fault codes. This specification extends the WS-Enumeration [WSENUM] set of faults by specifying additional SOAP faults that a server is permitted to return to the client to indicate that an error occurred while processing the request. This improves interoperability between clients and servers by providing a standardized set of errors that both sides of the communications session can understand. [MS-WSDS] specifies SOAP faults for the Enumerate and Pull operation defined by WS-Enumeration [WSENUM].

### 1.4 Relationship to Other Protocols

[MS-WSDS] is an extension to the WS-Enumeration [WSENUM] protocol built on top of SOAP [SOAP1.2-1/2003] as shown in the following layering diagram.

WS-Enumeration Directory Services Protocol Extensions	This extension
WS-Enumeration	Industry-standard
SOAP	Industry-standard
.....	

### 1.5 Prerequisites/Preconditions

This protocol extension does not assume any prerequisites or preconditions.

### 1.6 Applicability Statement

Use of the WS-Enumeration: Directory Services protocol extensions is suitable when searching XML representations of directory objects by means of WS-Enumeration and the granularity of resultant items is required to be lesser than the entire directory object's representation. These extensions cannot be used independently of WS-Enumeration [WSENUM], so they might not be applicable in applications that have already standardized on a protocol other than WS-Enumeration [WSENUM] for querying directory services.

The XPath 1.0-derived selection language, defined in [MS-ADDM] section 2.4, is used to specify the fragments requested out of the resultant items in the enumerate operation. This is suitable only when the data stored in a directory service could be represented as an XML document.

There is an implicit assumption that the directory service exposes semantics similar to that of a Lightweight Directory Access Protocol (LDAP) version 3 directory service [RFC2251] facilitating the use of LdapQuery language for the filter expression in the enumerate request.

## 1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

- **Supported Transports:** This protocol extension can be implemented using transports that support sending SOAP messages as described in section 2.1.
- **Protocol Versions:** This protocol extension is not versioned.
- **Capability Negotiation:** This protocol does not support capability negotiation.
- **Localization:** This protocol includes text strings in various SOAP faults. Localization considerations for such strings are specified in section 3.1.4.

## 1.8 Vendor-Extensible Fields

There are no vendor-extensible fields.

## 1.9 Standards Assignments

There are no standards assignments for this protocol extension.

## 2 Messages

### 2.1 Transport

[MS-WSDS] imposes no transport requirements or behaviors beyond those of the underlying WS-Enumeration [WSENUM] protocol. WS-Enumeration provides SOAP bindings for both SOAP 1.1 [SOAP1.1] and SOAP 1.2 [SOAP1.2-1/2003]. All messages MUST be formatted as specified by either SOAP 1.1 or SOAP 1.2.<1>

### 2.2 Common Message Syntax

This section contains common definitions that are used by this protocol. The syntax of the definitions uses an XML Schema, as defined in [XMLSCHEMA1] and [XMLSCHEMA2], and Web Services Description Language (WSDL), as defined in [WSDL].

#### 2.2.1 Namespaces

This specification defines and references various XML namespaces using the mechanisms specified in [XMLNS-2ED]. 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.

The prefixes and XML namespaces used in this specification include the following.

Prefix	Namespace URI	Reference
soapenv:	http://www.w3.org/2003/05/soap-envelope	[SOAP1.2-1/2003]
soapenv11:	http://schemas.xmlsoap.org/soap/envelope	[SOAP1.1]
wsa:	http://www.w3.org/2005/08/addressing	[WSADDR]
wsa2004	http://schemas.xmlsoap.org/ws/2004/08/addressing	[WSAddressing]
wsen:	http://schemas.xmlsoap.org/ws/2004/09/enumeration	[WSENUM]
ad:	http://schemas.microsoft.com/2008/1/ActiveDirectory	[MS-ADDM]
addata:	http://schemas.microsoft.com/2008/1/ActiveDirectory/Data	[MS-ADDM]
adlq:	http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/LdapQuery	The LdapQuery language URI, defined in section 3.1.4.1.1.1 of this specification
xsd:	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1]
xsi:	http://www.w3.org/2001/XMLSchema-instance	[XMLSCHEMA1]

#### 2.2.2 Messages

This specification does not define any new messages.

### **2.2.3 Elements**

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

### **2.2.4 Complex Types**

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

### **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.

## **2.3 Directory Service Schema Elements**

This specification does not make use of any directory service schema elements.

## 3 Protocol Details

The WS-Enumeration: Directory Services Protocol Extensions [MS-WSDS] extend how the WS-Enumeration [WSENUM] protocol operates between an application and a directory server. The requestor that is the client for the protocol sends a SOAP message containing a request of an Enumerate, Pull, Renew, GetStatus or Release operation to the server, and the server responds with a SOAP message, containing the response, or a SOAP fault, if an error occurred during server processing.<2>

The WS-Enumeration [WSENUM] specification of the simple SOAP-based protocol consists of multiple operations that are served by the WSDL port types: an Enumerate operation to begin the request, one or more Pull operations to retrieve the results, and a Release operation to terminate the query. In subsequent sections, extensions to some of these operations have been specified.

In the WS-Enumeration [WSENUM] specification, all the Enumerate operations are tied together by an enumeration context that represents a logical cursor through a sequence of data items. The enumeration context is just an identifier that the server returns in response to the WS-Enumeration [WSENUM] Enumerate operation and which the client is responsible for passing back while performing subsequent WS-Enumeration [WSENUM] operations. Thus, the requestor MUST initiate an Enumerate operation to create new enumeration contexts for subsequent traversal/retrieval of result items through Pull operation or performing other WS-Enumeration [WSENUM] operations. At the same time, WS-Enumeration [WSENUM] mandates that the state regarding the progress of the iteration MUST be maintained between requests by the server being enumerated or by the requestor.

Besides the requirements for WS-Enumeration [WSENUM], this protocol extension is simply a pass-through on the client side. 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.

**Note** The server implementation of the WS-Enumeration: Directory Services Protocol Extensions MAY<3> limit the maximum validity of an enumeration context. This limit, if implemented, applies across the Renew operation (section 3.1.4.3).

### 3.1 Enumeration Server Details

This section describes the server behavior of the [MS-WSDS] protocol extensions as they apply to the WS-Enumeration [WSENUM] WSDL port types. In WS-Enumeration, this WSDL port types is used to process five WSDL operations:

- Enumeration
- Pull
- Renew
- GetStatus
- Release

#### 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.

[MS-WSDS] operates on a collection of directory objects. The Enumerate operation specifies the search filter requesting for selected directory objects matching the filter or only fragments of those selected directory objects, whichever is required. The Pull operation retrieves the objects or fragments of those objects in the context of the previous Enumeration operation. The resultant items out of the Pull operation are a sequence of enumerated directory objects or their fragments. A directory object is a collection of one or more directory attributes. Each directory attribute is a collection of one or more directory attribute values. Directory attribute values are represented as XML elements.

For example, a user might be represented as a directory object. That directory object could have a directory attribute representing the user's given name, containing a single directory attribute value (for example, "John"). It could also have a directory attribute for the user's surname (for example, "Smith"). It might also have a directory attribute for the user's telephone numbers, which could contain multiple directory attribute values, one for each telephone number possessed by that user.

The directory object collection is represented as an XML document. The collection of directory objects containing the directory attributes, the contents of those attributes, and the representation of those directory objects (including their directory attributes and directory attribute values) as an XML document is implementation-defined.<4>

### 3.1.2 Timers

There are no timers in this protocol extension.

### 3.1.3 Initialization

When this protocol initializes, it MUST begin listening on endpoints for the Enumeration interface. The URIs for the endpoints, as well as the transport and security mechanisms to use, are implementation-dependent.<5>

### 3.1.4 Message Processing Events and Sequencing Rules

The following operations are defined by the WS-Enumeration [WSENUM] protocol. This protocol specifies extensions to the request, response, and SOAP faults associated with these operations.

Operation	Description
wsen:Enumerate	Creates a context mapped to the query filter specified for search.
wsen:Pull	Pulls the resultant objects in the context of a specific enumeration.
wsen:Renew	Renews the expiration time of the specified enumeration context.
wsen:GetStatus	Gets the expiration time of the specified enumeration context.
wsen:Release	Releases the specified enumeration context.

WS-Enumeration [WSENUM] section 3.6 states that if the server terminates an enumeration unexpectedly, it should send an EnumerationEnd SOAP message to the endpoint reference indicated when the enumeration was created.<6>

The subsequent sections also document the SOAP faults specified for use by servers that implement the [MS-WSDS] protocol extensions for specific operations. These faults SHOULD be used by servers while processing a WS-Enumeration [WSENUM] message to indicate to the client that a server-side error has occurred. Many of the faults are adopted from the WS-Enumeration [WSENUM] protocol, and some are from WS-Addressing [WSAddressing]; WSDS assigns these faults a specific meaning within the context of a WS-Enumeration operation that uses the WSDS extensions.

Server implementations are permitted to return additional faults beyond those described as follows. Where applicable, make use of the documented faults in order to maximize interoperability.

Some of the SOAP faults documented as follows specify English-language text in their fault reason. Server implementations are permitted to localize this text to other languages. Such localizations SHOULD maintain, to the extent possible, the same meaning as the English text supplied in this document.

All SOAP faults defined in this document MUST be sent as described in [WSASB] section 6.

In the tables describing faults in the later sections, the following apply.

- [Code] is the SOAP fault code.
- [Subcode] is the SOAP fault subcode.
- [Action] is the SOAP action URI for the fault.
- [Reason] is a human-readable explanation of the error.
- [Details] is an illustrative example of the fault detail. If not present, WSDS does not specify a fault detail for the fault.

### 3.1.4.1 wsen:Enumerate

This section defines the directory services extensions to the Enumerate request and response message defined in WS-Enumeration [WSENUM] section 3.1.

The Instance header defined in [MS-ADDM] section 2.5.1 MUST be attached by the requestor to a WS-Enumeration [WSENUM] Enumerate request message containing an <ad:instance> payload to inform the server about the directory instance targeted. See section 4.1 for an example of how the header is attached in an Enumerate request.

For the /soapenv:Envelope/ soapenv:Body/ wsen:Enumerate/ wsen:Filter/ element defined in WS-Enumeration [WSENUM] section 3.1, the filter expression language supported by [MS-WSDS] is "LdapQuery" defined in section 3.1.4.1.1 and identified by the following URI:

<http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/LdapQuery>

The URI stated previously MUST be the value of the /soapenv:Envelope/ soapenv:Body/ wsen:Enumerate/ wsen:Filter/ @Dialect attribute specified in WS-Enumeration [WSENUM] section 3.1.

The wsen:Filter element is optional. If the wsen:Filter element is absent in the request (that is, no query filter has been provided in the request), the server SHOULD use the following Ldap query filter by default.

```
<wsen:Filter Dialect="http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/LdapQuery
">
  <adlq:LdapQuery>
    <adlq:Filter>
      (objectClass=*)
    </adlq:Filter>
    <adlq:BaseObject>
      <Default domain NC DN>
    </adlq:BaseObject>
    <adlq:Scope>
      subtree
    </adlq:Scope>
  </adlq:LdapQuery>
</wsen:Filter>
```

The WS-Enumeration [WSENUM] Enumerate request can specify the expiration time of the request as a specific time or duration type. If the wsen:Expires element specifying expiration time is absent in the request, meaning indefinite lifetime for the enumeration, it is up to the server to grant such enumeration.<7>

WS-Enumeration [WSENUM] also specifies for the Enumerate response that the expiration time can be either an absolute time or a duration but should be of the same type as the requested expiration.<8>

### 3.1.4.1.1 Elements

The following XML schema element definitions are specific to extensions of the enumerate operation defined in WS-Enumeration [WSENUM] section 3.1.

Element	Description
<adlq:LdapQuery>	Specifies the format for the LdapQuery query element.
<ad:Selection>	Specifies the fragments for the Selection element.
<ad:Sorting>	Specifies the Sorting element.

#### 3.1.4.1.1.1 adlq:LdapQuery

This section specify the format of optional LDAP query element which is used to define the optional /soapenv:Envelope/soapenv:Body/wsen:Enumerate/wsen:Filter element of the Enumerate request described in WS-Enumeration [WSENUM] section 3.1.

Following is the XML Schema [XMLSCHEMA1] definition of the LdapQuery element.

```
<xsd:element name = "LdapQuery">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="Filter" type="xsd:string" minOccurs="1"
        maxOccurs = "1" />
      <xsd:element name="BaseObject" type="xsd:string"
        minOccurs="1" maxOccurs="1" />
      <xsd:element name="Scope" type="xsd:string" minOccurs="1"
        maxOccurs="1" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

For example, the following XML shows the contents of LdapQuery element.

```
<adlq:LdapQuery>
  <adlq:Filter>
    <RFC 2254 filter string>
  </adlq:Filter>
  <adlq:BaseObject>
    <object Reference Property, a.k.a. GUID, or object DN>
  </adlq:BaseObject>
  <adlq:Scope>
    (base | onelevel | subtree)
  </adlq:Scope>
```



```
</adlq:LdapQuery>
```

If the `adlq:LdapQuery` element is absent in the request (that is, no query filter has been provided in the request), the server SHOULD use the following Ldap query filter by default:

```
<wsen:Filter Dialect="http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/LdapQuery">
  <adlq:LdapQuery>
    <adlq:Filter>
      (objectClass=*)
    </adlq:Filter>
    <adlq:BaseObject>
      <Default domain NC DN>
    </adlq:BaseObject>
    <adlq:Scope>
      subtree
    </adlq:Scope>
  </adlq:LdapQuery>
</wsen:Filter>
```

#### 3.1.4.1.1.1.1 adlq:filter

The requestor MUST specify the LDAP search filter strings as defined in [RFC2254] as the query string for the required filter element `wsen:Enumerate/wsens:Filter/adlq:LdapQuery/adlq:filter` in the Enumerate request.

```
<adlq:Filter>
  <RFC 2254 filter string>
</adlq:Filter>
```

#### 3.1.4.1.1.1.2 adlq:BaseObject

The requestor MUST specify the object reference property (GUID or distinguished name of the object) as defined in [MS-ADDM] section 2.3.3.1 under the required BaseObject element `wsen:Enumerate/wsens:Filter/adlq:LdapQuery/adlq:BaseObject` in the Enumerate request.

```
<adlq:BaseObject>
  <object Reference Property, a.k.a. GUID, or object DN>
</adlq:BaseObject>
```

#### 3.1.4.1.1.1.3 adlq:Scope

The requestor MUST specify "base", "onelevel", or "subtree" under the required scope element `wsen:Enumerate/wsens:Filter/adlq:LdapQuery/adlq:Scope` in the Enumerate request. This element specifies the scope for the directory search that is to be performed.

```
<adlq:Scope>
  (base | onelevel | subtree)
</adlq:Scope>
```

Scope	Explanation
base	Limits the search scope to the base object. The result contains a maximum of one object.
onelevel	Limits the search scope to the immediate child objects of the base object, excluding the base object.
subtree	Limits the search scope to the whole subtree, including the base object and all its child objects.

### 3.1.4.1.1.2 ad:Selection

The optional element /soapenv:Envelope/soapenv:Body/wsen:Enumerate/ad:Selection specifies which fragments (SelectionProperties) (that is, what portions (directory attributes)) of the object to return for a certain enumeration. The selection element is specified using a selection language that is derived from XPath 1.0. If specified, the Selection element MUST contain at least one SelectionProperty (see section 3.1.4.1.1.2.1).

If the Selection element is absent in the request, all default attributes of the directory object SHOULD be returned.<9>

Following is the XML Schema [XMLSCHEMA1] definition of the Selection element.

```
<xsd:element name="Selection">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="SelectionProperty" type="xsd:string" minOccurs="1"
        maxOccurs="unbounded"/>
    </xsd:sequence>
    <xsd:attribute name="Dialect"
      fixed="http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/XPath-Level-1"
      use="required"/>
  </xsd:complexType>
</xsd:element>
```

For example, the following XML shows the contents of the Selection element.

```
<ad:Selection Dialect="http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/XPath-
Level-1">
  <ad:SelectionProperty>
    <XPath Level 1>
  </ad:SelectionProperty>
</ad:Selection>
```

The value of /soapenv:Envelope/ soapenv:Body/wsen:Enumerate/ad:Selection/@Dialect is specified in section 3.1.4.1.2.1.

#### 3.1.4.1.1.2.1 ad:SelectionProperty

The required element

/soapenv:Envelope/soapenv:Body/wsen:Enumerate/ad:Selection/ad:SelectionProperty specifies which directory attribute is to be returned for the resultant objects evaluating to true for that particular enumeration. The Requestor MUST use XPath 1.0-derived Selection Language, defined in [MS-ADDM] section 2.4, to specify this element. The language is indicated by the Dialect attribute, defined in section 3.1.4.1.2.1, on its parent node ad:Selection, defined in section 3.1.4.1.1.2.

For doing range retrieval of a multi-valued directory attribute, the extension to [WSENUM] specified in [MS-ADDM] section 2.7 SHOULD be used with this XML element.

Selection property <ad:all> SHOULD be specified to indicate that the Pull operation SHOULD return all the default attributes of the resultant directory objects.<10>

Following is the XML Schema [XMLSCHEMA1] definition of the SelectionProperty element.

```
<xsd:element name="SelectionProperty" type="xsd:string" minOccurs="1"
maxOccurs = "unbounded"/>
```

### 3.1.4.1.1.3 ad:Sorting

The optional element /soapenv:Envelope/soapenv:Body/wsen:Enumerate/ad:Sorting determines which attribute the sorting of resultant items in the response message for that specific enumeration would depend on. The Sorting element is optional and, if specified, MUST contain exactly one SortingProperty (see section 3.1.4.1.1.3.1). An optional Ascending attribute (see section 3.1.4.1.2.2) is used to specify ascending or descending sort order. If the Sorting element is not specified, the results are not sorted.

Following is the XML Schema [XMLSCHEMA1] definition of the Sorting element:

```
<xsd:element name = "Sorting">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="SortingProperty" minOccurs="1"
maxOccurs = "1">
        <xsd:complexType>
          <xsd:simpleContent>
            <xsd:extension base="xsd:string">
              <xsd:attribute name="Ascending" use="optional" default="true"
type="xsd:boolean"/>
            </xsd:extension>
          </xsd:simpleContent>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
    <xsd:attribute name="Dialect"
fixed=http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/XPath-Level-1
use="required"/>
  </xsd:complexType>
</xsd:element>
```

For example, the following XML shows the contents of the Sorting element:

```
<ad:Sorting Dialect="http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/XPath-Level-1">
  <ad:SortingProperty Ascending=(true|false)>
    <XPath Level>
  </ad:SortingProperty>
</ad:Sorting>
```

The value of /s:Envelope/s:Body/wsen:Enumerate/ad:Sorting/@Dialect is specified in section 3.1.4.1.2.2.

#### 3.1.4.1.1.3.1 ad:SortingProperty

The required element

/soapenv:Envelope/soapenv:Body/wsen:Enumerate/ad:Sorting/ad:SortingProperty specifies the sorting directory attribute using XPath 1.0-derived Selection Language, defined in [MS-ADDM] section 2.4, and the sort order using the Ascending attribute (see section 3.1.4.1.2.3). If the Ascending attribute is absent, the default sorting order is ascending.

The following XML shows the contents of the SortingProperty element:

```
<ad:SortingProperty Ascending=(true|false)>
  <XPath Level 1>
</ad:SortingProperty>
```

### 3.1.4.1.2 Attributes

The following XML schema attribute definitions are specific to these Enumerate request extensions.

Attribute	Description
<ad:Selection/@Dialect>	Specifies the Selection property.
<ad:Sorting/@Dialect>	Specifies the Sorting property.
<ad:Sorting/ad:SortingProperty/@Ascending>	Specifies the ascending Sorting property.

#### 3.1.4.1.2.1 ad:Selection/@Dialect

The XPath 1.0-derived Selection Language, defined in [MS-ADDM] section 2.4, is used to specify selection properties. This derived language is identified by the following URI:

<http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/XPath-Level-1>

The URI stated previously MUST be the value of the /soapenv:Envelope/ soapenv:Body/ wsen:Enumerate / \* /ad:Selection/@Dialect required attribute.

#### 3.1.4.1.2.2 ad:Sorting/@Dialect

The XPath 1.0-derived Selection Language, defined in [MS-ADDM] section 2.4, is used to specify the sorting property. This derived language is identified by the following URI:

<http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/XPath-Level-1>

The URI stated previously MUST be the value of the /soapenv:Envelope/soapenv:Body/ wsen:Enumerate/ \* /ad:Sorting/ @Dialect required attribute.

#### 3.1.4.1.2.3 ad:Sorting/ad:SortingProperty/@Ascending

The value for the optional /soapenv:Envelope/ soapenv:Body/ wsen:Enumerate/ \* / ad:Sorting/ ad:SortingProperty/@Ascending attribute MUST be of type Boolean. If true, the sorting order of the resultant items based on the Sorting Property is ascending. If false, it is descending.

### 3.1.4.1.3 SOAP Faults

This section defines the extensions to SOAP fault messages for the Enumerate operation defined in WS-Enumeration [WSENUM] section 3.1.

#### 3.1.4.1.3.1 ad:EnumerationContextLimitExceeded

Server implementations SHOULD reject the Enumerate request and return the following fault when there is more than a specific, server-imposed limit number<11> of Enumeration contexts open either for the user or in total.

Implementations MAY<12> supply a fault detail of their choosing.

SOAP Element	Value
[Code]	soapenv:Sender
[Subcode]	ad:EnumerationContextLimitExceeded
[Action]	http://schemas.microsoft.com/2008/1/ActiveDirectory/Data/fault
[Reason]	Too many enumeration contexts open.
[Details]	Implementation-defined and MAY be empty.

### 3.1.4.1.3.2 ad:UnsupportedSelectOrSortDialectFault

Server implementations SHOULD reject the Enumerate request and return the following fault when it does not support the dialect specified for Selection or sorting properties defined in sections 3.1.4.1.2.2 and 3.1.4.1.2.3. The Server SHOULD provide the dialect supported as part of the fault detail shown in the following table.

SOAP Element	Value
[Code]	soapenv:Sender
[Subcode]	ad:UnsupportedSelectOrSortDialectFault
[Action]	http://schemas.microsoft.com/2008/1/ActiveDirectory/Data/fault
[Reason]	Specified dialect for Selection properties (or Sorting property) is not supported.
[Details]	<pre>&lt;soapenv:Detail&gt; &lt;ad:SupportedSelectOrSortDialect&gt; http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/XPath-Level-1 &lt;/ad:SupportedSelectOrSortDialect&gt; &lt;/soapenv:Detail&gt;</pre>

### 3.1.4.1.3.3 ad:InvalidPropertyFault

Server implementations SHOULD reject the Enumerate request and return the following fault when any selection or sorting property specified in the Enumerate request is not a valid directory attribute. Fault detail SHOULD contain the invalid selection or sorting property present in the Enumerate request.

SOAP Element	Value
[Code]	soapenv:Sender
[Subcode]	ad:InvalidPropertyFault
[Action]	http://schemas.microsoft.com/2008/1/ActiveDirectory/Data/fault
[Reason]	Sorting or selection property is invalid.
[Details]	<pre>&lt;soapenv:Detail&gt; &lt;ad:EnumerateFault&gt;</pre>

SOAP Element	Value
	<pre>&lt;ad:Error&gt;Selection or Sort property's value is not a valid directory attribute. (OR Selection or Sort property's syntax is not valid with respect to the dialect.)&lt;/ad:Error&gt; &lt;ad:ShortError&gt;InvalidPropertyValueDetail(OR InvalidPropertySyntaxDetail)&lt;/ad:ShortError&gt; &lt;ad:InvalidProperty&gt;Invalid property&lt;/ad:InvalidProperty&gt; &lt;/ad:EnumerateFault&gt; &lt;/soapenv:Detail&gt;</pre>

#### 3.1.4.1.3.4 ad:InvalidSortKey

If the sorting property defined in section 3.1.4.1.1.3.1 is one of the non-LDAP directory attributes including the synthetic (implementation-specific) directory attributes, <13> "<ad:all>", or if more than one sort key is specified, the server SHOULD reject the Enumerate request and return the following fault.

Implementations MAY<14> supply a fault detail of their own choosing.

SOAP Element	Value
[Code]	soapenv:Sender
[Subcode]	ad:InvalidSortKey
[Action]	http://schemas.microsoft.com/2008/1/ActiveDirectory/Data/fault
[Reason]	Invalid sorting property.
[Details]	Implementation-defined and MAY be empty.

#### 3.1.4.1.3.5 wsen:CannotProcessFilter

When the wsen:filter or adlq:LdapQuery element is absent in the enumerate request, the server uses a default Ldap query filter for search as defined in section 3.1.4.1 and 3.1.4.1.1.1. However, if the server encounters an error while using the default Ldap query filter, it SHOULD return the following fault.

Implementations MAY<15> supply a default detail of their own choosing.

SOAP Element	Value
[Code]	soapenv:Sender
[Subcode]	wsen:CannotProcessFilter
[Action]	http://schemas.xmlsoap.org/ws/2004/09/enumeration/fault
[Reason]	Invalid query language expression.
[Details]	Implementation-defined and MAY be empty.

#### 3.1.4.1.3.6 wsa2004:EndPointUnavailable

The server SHOULD notify a requestor of other application-level faults by generating a wsa2004:EndPointUnavailable SOAP fault defined in the WS-Addressing [WSAddressing] protocol.

wsa2004:EndPointUnavailable SOAP fault SHOULD be generated by the server when it encounters any other application, directory, XML serialization or deserialization, schema validation, or unknown error that is not specific to faults stated in the previous sections.<16>

Implementations MAY<17> supply a fault detail of their own choosing.

SOAP Element	Value
[Code]	soapenv:Receiver
[Subcode]	wsa2004:EndPointUnavailable
[Action]	http://schemas.xmlsoap.org/ws/2004/08/addressing/fault
[Reason]	Endpoint unavailable.
[Details]	Implementation-defined and MAY be empty.

### 3.1.4.2 wsen:Pull

This section defines the directory services extensions to the Pull response message defined in WS-Enumeration [WSENUM] section 3.2.

The WS-Enumeration [WSENUM] Pull specification as defined in WS-Enumeration [WSENUM] section 3.2 leaves the contents of the wsen:PullResponse/wsen:Items element open-ended for a Pull response. The data returned by the server via the wsen:Items element (defined in WS-Enumeration [WSENUM] section 3.2) is structured in accord with the XML data model specified in [MS-ADDM] section 2.3.2, except that only those attributes that were requested in the wsen:Enumerate/ad:Selection (see section 3.1.4.1.1.2) element are returned. Also, in addition to requested attributes through the wsen:Selection element in the Enumerate request, the server always returns <ad:objectReferenceProperty> synthetic attribute (as specified in [MS-ADDM] section 2.3.3.1) in the Pull response.

The fragments or portions of the resultant objects evaluating to be true for the specific enumeration query are returned as children nodes with the most specific structural object class ([MS-ADTS] section 3.1.1.1.4) being the root element. The server SHOULD specify the root element as "addata:top" in the Pull response if the LDAP display name of the most specific structural object class for the result fragment is not available as specified in [MS-ADDM] section 2.3.2. This sequence of result fragments forms children nodes to the <wsen:Items> element.

The following XML shows the contents of the Items element.

```
<wsen:Items>
  <addata:user>
    <ad:objectReferenceProperty>
      <ad:value xsi:type="xsd:string">
        13195F20-8F96-46ed-BFF2-7891817FFEB8
      </ad:value>
    </ad:objectReferenceProperty>
    <addata:givenName>
      <ad:value xsi:type="xsd:string">John</ad:value>
    </addata:givenName>
  </addata:user>
  <addata:user>
    <ad:objectReferenceProperty>
```

```

    <ad:value xsi:type= "xsd:string">
      ED18488A-3042-4e12-B7D1-69A059F80BC1
    </ad:value>
  </ad:objectReferenceProperty>
  <addata:givenName>
    <ad:value xsi:type= "xsd:string">Robert</ad:value>
  </addata:givenName>
</addata:user>
</wsen:Items>

```

The enumeration context is provided in the Pull request in order to fetch the resultant objects mapped to the query associated with it. Servers SHOULD reject the Pull request and return the ad:MaxCharsNotSupported fault, defined in section 3.1.4.2.1.1, if the optional element wsen:MaxCharacters (as defined in [WSENUM] section 3.2) is present.<18>

The optional element wsen:MaxTime indicates the maximum amount of time the initiator is willing to allow the server to assemble the Pull response as specified in [WSENUM] section 3.2. When wsen:MaxTime is absent in a Pull request message, the server SHOULD use a timeout to limit the time spent processing the Pull request. The server SHOULD also restrict the maximum value of wsen:MaxTime to less than this timeout value, and any attempt to set wsen:MaxTime to a value greater than this timeout value SHOULD return the ad:MaxTimeExceedsLimit fault defined in section 3.1.4.2.1.5.

An optional ad:controls element can also be passed to the pull request, according to the requirements in [MS-WSPELD] section 2.2.3.1.

### 3.1.4.2.1 SOAP Faults

This section defines the extensions to SOAP fault messages for the Pull operation defined in WS-Enumeration [WSENUM] section 3.2.

#### 3.1.4.2.1.1 ad:MaxCharsNotSupported

Server implementations SHOULD reject the Pull request and return the following fault when it does not support the wsen:MaxCharacters element specified in the Pull request.

Implementations MAY<19> supply a fault detail of their choosing.

SOAP Element	Value
[Code]	soapenv:Sender
[Subcode]	ad:MaxCharsNotSupported
[Action]	http://schemas.microsoft.com/2008/1/ActiveDirectory/Data/fault
[Reason]	MaxChars specified in the request.
[Details]	Implementation-defined and MAY be empty.

#### 3.1.4.2.1.2 wsen:InvalidEnumerationContext

The server SHOULD return the wsen:InvalidEnumerationContext fault defined by WS-Enumeration [WSENUM] when the enumeration context specified in the respective Pull request belongs to a different security principal or session.

Implementations MAY<20> supply a fault detail of their choosing.



SOAP Element	Value
[Code]	soapenv:Sender
[Subcode]	wsen:InvalidEnumerationContext
[Action]	http://schemas.xmlsoap.org/ws/2004/09/enumeration/fault
[Reason]	Invalid enumeration context specified in the request.
[Details]	Implementation-defined and MAY be empty.

### 3.1.4.2.1.3 wsa2004:DestinationUnreachable

The server SHOULD return the wsa2004:DestinationUnreachable fault defined in the WS-Addressing [WSAddressing] protocol when the object requested through the corresponding Enumerate request of this Pull operation does not exist in the directory.

Implementations MAY<21> supply a fault detail of their choosing.

SOAP Element	Value
[Code]	soapenv:Sender<22>
[Subcode]	wsa2004:DestinationUnreachable
[Action]	http://schemas.xmlsoap.org/ws/2004/08/addressing/fault
[Reason]	The failed operation was attempted on a nonexistent directory object.
[Details]	Implementation-defined and MAY be empty.

### 3.1.4.2.1.4 wsa2004:EndpointUnavailable

The server SHOULD notify a requestor of other application-level faults by generating a wsa2004:EndPointUnavailable SOAP fault defined in the WS-Addressing [WSAddressing] protocol.

wsa2004:EndPointUnavailable SOAP fault SHOULD be generated by the server when it encounters any other application, directory, XML serialization or deserialization, schema validation, or unknown error that is not specific to faults stated in the previous sections.

Implementations MAY<23> supply a fault detail of their choosing.

SOAP Element	Value
[Code]	soapenv:Receiver
[Subcode]	wsa2004:EndpointUnavailable
[Action]	http://schemas.xmlsoap.org/ws/2004/08/addressing/fault
[Reason]	Endpoint unavailable.
[Details]	Implementation-defined and MAY be empty.

### 3.1.4.2.1.5 ad:MaxTimeExceedsLimit

Server implementations SHOULD reject the Pull request and return the following fault when the value of wsen:MaxTime is greater than the default value mentioned in section 3.1.4.2.

Implementations MAY<24> supply a fault detail of their choosing.

SOAP Element	Value
[Code]	soapenv:Sender
[Subcode]	ad:MaxTimeExceedsLimit
[Action]	http://schemas.microsoft.com/2008/1/ActiveDirectory/Data/fault
[Reason]	MaxTime exceeds the limit.
[Details]	Implementation-defined and MAY be empty.

### 3.1.4.3 wsen:Renew

This section defines only the extensions to SOAP fault messages for the Renew operation defined in WS-Enumeration [WSENUM] section 3.3.

#### 3.1.4.3.1 SOAP faults

The server SHOULD reject the Renew request on application-level errors, client-side request errors, or directory errors; and return an appropriate fault from below.

##### 3.1.4.3.1.1 wsen:InvalidEnumerationContext

The server SHOULD return the wsen:InvalidEnumerationContext fault defined by WS-Enumeration [WSENUM] when the enumeration context specified in the respective Renew request belongs to a different security principal or session.

Implementations MAY<25> supply a fault detail of their choosing.

SOAP Element	Value
[Code]	soapenv:Sender
[Subcode]	wsen:InvalidEnumerationContext
[Action]	http://schemas.xmlsoap.org/ws/2004/09/enumeration/fault
[Reason]	Invalid enumeration context specified in the request.
[Details]	Implementation-defined and MAY be empty.

##### 3.1.4.3.1.2 wsa2004:EndpointUnavailable

The server SHOULD notify a requestor of other application-level faults by generating a wsa2004:EndPointUnavailable SOAP fault defined in the WS-Addressing [WSAddressing] protocol.

wsa2004:EndPointUnavailable SOAP fault SHOULD be generated by the server when it encounters any other application, directory, XML serialization or deserialization, schema validation, or unknown error that is not specific to faults stated in the previous sections.

Implementations MAY<26> supply a fault detail of their choosing.

SOAP Element	Value
[Code]	soapenv:Receiver
[Subcode]	wsa2004:EndpointUnavailable
[Action]	http://schemas.xmlsoap.org/ws/2004/08/addressing/fault
[Reason]	Endpoint unavailable.
[Details]	Implementation-defined and MAY be empty.

### 3.1.4.4 wsen:GetStatus

This section defines only the extensions to SOAP fault messages for the GetStatus operation defined in WS-Enumeration [WSENUM] section 3.4.

#### 3.1.4.4.1 SOAP Faults

The server SHOULD reject the GetStatus request on application-level errors, client-side request errors or directory errors, and return an appropriate fault from below.

##### 3.1.4.4.1.1 wsen:InvalidEnumerationContext

The server SHOULD return the wsen:InvalidEnumerationContext fault defined by WS-Enumeration [WSENUM] when the enumeration context specified in the respective Pull request belongs to a different security principal or session.

Implementations MAY<27> supply a fault detail of their choosing.

SOAP Element	Value
[Code]	soapenv:Sender
[Subcode]	wsen:InvalidEnumerationContext
[Action]	http://schemas.xmlsoap.org/ws/2004/09/enumeration/fault
[Reason]	Invalid enumeration context specified in the request.
[Details]	Implementation-defined and MAY be empty.

##### 3.1.4.4.1.2 wsa2004:EndpointUnavailable

The server SHOULD notify a requestor of other application-level faults by generating a wsa2004:EndPointUnavailable SOAP fault defined in the WS-Addressing [WSAddressing] protocol.

wsa2004:EndPointUnavailable SOAP fault SHOULD be generated by the server when it encounters any other application, directory, XML serialization or deserialization, schema validation, or unknown error that is not specific to faults stated in the previous sections.

Implementations MAY<28> supply a fault detail of their choosing.

SOAP Element	Value
[Code]	soapenv:Receiver
[Subcode]	wsa2004:EndpointUnavailable

SOAP Element	Value
[Action]	http://schemas.xmlsoap.org/ws/2004/08/addressing/fault
[Reason]	Endpoint unavailable.
[Details]	Implementation-defined and MAY be empty.

### 3.1.4.5 wsen:Release

This section defines only the extensions to SOAP fault messages for the Release operation defined in WS-Enumeration [WSENUM] section 3.5.

#### 3.1.4.5.1 SOAP Faults

The server SHOULD reject the Release request and return the following fault on application-level errors, client-side request errors, or directory errors encountered during the Release operation.

##### 3.1.4.5.1.1 wsa2004:EndpointUnavailable

The server SHOULD notify a requestor of other application-level faults by generating a wsa2004:EndPointUnavailable SOAP fault defined in the WS-Addressing [WSAddressing] protocol.

wsa2004:EndPointUnavailable SOAP fault SHOULD be generated by the server when it encounters any other application, directory, XML serialization or deserialization, schema validation, or unknown error that is not specific to faults stated in the previous sections.

Implementations MAY<29> supply a fault detail of their choosing.

SOAP Element	Value
[Code]	soapenv:Receiver
[Subcode]	wsa2004:EndpointUnavailable
[Action]	http://schemas.xmlsoap.org/ws/2004/08/addressing/fault
[Reason]	Endpoint unavailable.
[Details]	Implementation-defined and MAY be empty.

### 3.1.5 Timer Events

None.

### 3.1.6 Other Local Events

None.

## 4 Protocol Examples

This section contains examples of the WS-Enumeration extensions defined in the [MS-WSDS] protocol. For illustrative purposes, these examples have been shown in the context of Active Directory Web Services behavior.

### 4.1 WS-Enumerate Directory Services Extension "Enumerate" Request Example

This example demonstrates a [MS-WSDS] Enumerate operation. In this SOAP request message, the requestor is specifying the LDAP query filter for search and directory attributes it requests out of the resultant objects under the selection element. The requestor is also asking to base the sorting of the resultant fragments in the response on the LDAP givenName attribute. The directory instance on which this operation is being performed is identified by its string-valued instance header ldap:389.

```
<soapenv:Envelope
  xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope"
  xmlns:wsa="http://www.w3.org/2005/08/addressing">
  <soapenv:Header>
    <wsa:Action soapenv:mustUnderstand="1">
      http://schemas.xmlsoap.org/ws/2004/09/enumeration/Enumerate
    </wsa:Action>
    <instance xmlns="http://schemas.microsoft.com/2008/1/ActiveDirectory">
      ldap:389
    </instance>
    <wsa:MessageID>
      urn:uuid:e36457ff-d0f1-4c85-abe6-6cdf4bd511e9
    </wsa:MessageID>
    <wsa:ReplyTo>
      <wsa:Address>
        http://www.w3.org/2005/08/addressing/anonymous
      </wsa:Address>
    </wsa:ReplyTo>
    <wsa:To
  soapenv:mustUnderstand="1">net.tcp://server01.fabrikam.com:9389/ActiveDirectoryWebServices/Wi
  ndows/Enumeration</wsa:To>
  </soapenv:Header>
  <soapenv:Body>
    <wsen:Enumerate
      xmlns:wsen="http://schemas.xmlsoap.org/ws/2004/09/enumeration"

      xmlns:adlq="http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/LdapQuery"
      xmlns:addata="http://schemas.microsoft.com/2008/1/ActiveDirectory/Data"
      xmlns:ad="http://schemas.microsoft.com/2008/1/ActiveDirectory"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:xsd="http://www.w3.org/2001/XMLSchema">
      <wsen:Filter
        Dialect="http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/LdapQuery">
        <adlq:LdapQuery>
          <adlq:Filter>(objectclass=user)</adlq:Filter>
          <adlq:BaseObject>cc36a2a7-79a2-4d96-b1c2-
          31c30493b801</adlq:BaseObject>
          <adlq:Scope>subtree</adlq:Scope>
        </adlq:LdapQuery>
        </wsen:Filter>
        <ad:Selection
          Dialect="http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/XPath-Level-1">
          <ad:SelectionProperty>
            ad:container-hierarchy-parent
          </ad:SelectionProperty>
          <ad:SelectionProperty>
            addata:relativeDistinguishedName
          </ad:SelectionProperty>
          <ad:SelectionProperty>
            addata:givenName
```

```

        </ad:SelectionProperty>
    </ad:Selection>
    <ad:Sorting
Dialect="http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/XPath-Level-1">
        <ad:SortingProperty Ascending="true">
            addata:givenName
        </ad:SortingProperty>
    </ad:Sorting>
</wsen:Enumerate>
</soapenv:Body>
</soapenv:Envelope>

```

## 4.2 WS-Enumerate Directory Services Extension "Enumerate" Response Example

This example demonstrates an [MS-WSDS] Enumerate response to the Enumerate request in the previous example. In this SOAP response message, the expiration time and enumeration context associated with the operation are provided. Since the expiration time was absent in the request, the server assigned the maximum expiration time supported.

```

<soapenv:Envelope xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope"
xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <soapenv:Header>
        <wsa:Action
soapenv:mustUnderstand="1">http://schemas.xmlsoap.org/ws/2004/09/enumeration/EnumerateResponse
        </wsa:Action>
        <wsa:RelatesTo>
            urn:uuid:e36457ff-d0f1-4c85-abe6-6cdf4bd511e9
        </wsa:RelatesTo>
        <wsa:To soapenv:mustUnderstand="1">http://www.w3.org/2005/08/addressing/anonymous
        </wsa:To>
    </soapenv:Header>
    <soapenv:Body>
        <wsen:EnumerateResponse
            xmlns:wsen="http://schemas.xmlsoap.org/ws/2004/09/enumeration"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            xmlns:xsd="http://www.w3.org/2001/XMLSchema">
            <wsen:Expires>2008-08-18T22:07:41.0109506Z</wsen:Expires>
            <wsen:EnumerationContext>cda3e08b-cec1-42bb-8245-7cb6235a24b8
            </wsen:EnumerationContext>
        </wsen:EnumerateResponse>
    </soapenv:Body>
</soapenv:Envelope>

```

## 4.3 WS-Enumerate Directory Services Extension "Pull" Request Example

This example demonstrates a [MS-WSDS] Pull request. In this SOAP request message, the requestor is specifying the enumeration context in the form of GUID and the number of elements (wsen:maxElements) it is requesting out of the resultant objects at a time in the Pull response.

```

<soapenv:Envelope
    xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope"
    xmlns:wsa="http://www.w3.org/2005/08/addressing">
    <soapenv:Header>
        <wsa:Action
soapenv:mustUnderstand="1">http://schemas.xmlsoap.org/ws/2004/09/enumeration/Pull</wsa:Action
    >
        <wsa:To soapenv:mustUnderstand="1">
            net.tcp://server01.fabrikam.com:9389/ActiveDirectoryWebServices/Windows/Enumeration
        </wsa:To>
        <wsa:MessageID>

```

```

        urn:uuid:b22747a9-ca15-41de-8c91-5a51bd88669c
      </wsa:MessageID>
      <wsa:ReplyTo>
        <wsa:Address>
          http://www.w3.org/2005/08/addressing/anonymous
        </wsa:Address>
      </wsa:ReplyTo>
    </soapenv:Header>
    <soapenv:Body>
      <wsen:Pull xmlns:wsen="http://schemas.xmlsoap.org/ws/2004/09/enumeration">
        <wsen:EnumerationContext>cda3e08b-cec1-42bb-8245-7cb6235a24b8</wsen:EnumerationContext>
        <wsen:MaxTime>PT10S</wsen:MaxTime>
        <wsen:MaxElements>2</wsen:MaxElements>
      </wsen:Pull>
    </soapenv:Body>
  </soapenv:Envelope>

```

#### 4.4 WS-Enumerate Directory Services Extension "Pull" Response Example

This example demonstrates a [MS-WSDS] Pull response to the request specified in section 4.2 in context of the Enumeration defined in section 4.1.

```

<soapenv:Envelope
  xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope"
  xmlns:wsa="http://www.w3.org/2005/08/addressing">
  <soapenv:Header>
    <wsa:Action soapenv:mustUnderstand="1">
      http://schemas.xmlsoap.org/ws/2004/09/enumeration/PullResponse
    </wsa:Action>
    <wsa:RelatesTo>urn:uuid:b22747a9-ca15-41de-8c91-5a51bd88669c</wsa:RelatesTo>
    <wsa:To soapenv:mustUnderstand="1">
      http://www.w3.org/2005/08/addressing/anonymous
    </wsa:To>
  </soapenv:Header>
  <soapenv:Body>
    <wsen:PullResponse
      xmlns:wsen="http://schemas.xmlsoap.org/ws/2004/09/enumeration"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xmlns:xsd="http://www.w3.org/2001/XMLSchema"
      xmlns:ad="http://schemas.microsoft.com/2008/1/ActiveDirectory"
      xmlns:addata="http://schemas.microsoft.com/2008/1/ActiveDirectory/Data">
      <wsen:EnumerationContext>
        cda3e08b-cec1-42bb-8245-7cb6235a24b8
      </wsen:EnumerationContext>
      <wsen:Items>
        <addata:user>
          <ad:objectReferenceProperty>
            <ad:value xsi:type="xsd:string">
              373e1409-cf88-41dc-b8ea-bdd27d54e073
            </ad:value>
          </ad:objectReferenceProperty>
          <ad:container-hierarchy-parent>
            <ad:value xsi:type="xsd:string">
              41816238-95ca-48d9-9a99-3bd9ae9e0e42
            </ad:value>
          </ad:container-hierarchy-parent>
          <ad:relativeDistinguishedName>
            <ad:value xsi:type="xsd:string">CN=TestUser1</ad:value>
          </ad:relativeDistinguishedName>
          <addata:givenName LdapSyntax="UnicodeString">
            <ad:value xsi:type="xsd:string">John</ad:value>
          </addata:givenName>
        </addata:user>
        <addata:user>
          <ad:objectReferenceProperty>
            <ad:value xsi:type="xsd:string">

```

```

        51d67624-d52d-421d-a0d6-1dc350abd009
    </ad:value>
</ad:objectReferenceProperty>
<ad:container-hierarchy-parent>
    <ad:value xsi:type="xsd:string">
        41816238-95ca-48d9-9a99-3bd9ae9e0e42
    </ad:value>
</ad:container-hierarchy-parent>
<ad:relativeDistinguishedName>
    <ad:value xsi:type="xsd:string">CN=TestUser2</ad:value>
</ad:relativeDistinguishedName>
<addata:givenName LdapSyntax="UnicodeString">
    <ad:value xsi:type="xsd:string">Robert</ad:value>
</addata:givenName>
</addata:user>
</wsen:Items>
</wsen:PullResponse>
</soapenv:Body>
</soapenv:Envelope>

```

#### 4.5 WS-Enumerate Directory Services Extension "FaultDetail" Example

The following example shows a WS-Enumeration specific fault response message returned by Active Directory Web Services with detail element (specified in [MS-ADDM] section 2.6) to an Enumerate request containing an invalid attribute as a sorting property.

```

<soapenv:Envelope
  xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope"
  xmlns:wsa="http://www.w3.org/2005/08/addressing">
  <soapenv:Header>
    <wsa:Action soapenv:mustUnderstand="1">
      http://schemas.microsoft.com/2008/1/ActiveDirectory/Data/fault
    </wsa:Action>
    <wsa:RelatesTo>
      urn:uuid:50cfcace-0035-403b-88bf-1355f19eec65
    </wsa:RelatesTo>
    <wsa:To soapenv:mustUnderstand="1">
      http://www.w3.org/2005/08/addressing/anonymous
    </wsa:To>
  </soapenv:Header>
  <soapenv:Body>
    <soapenv:Fault>
      <soapenv:Code>
        <soapenv:Value>soapenv:Sender</soapenv:Value>
        <soapenv:Subcode>
          <soapenv:Value xmlns:a="http://schemas.microsoft.com/2008/1/ActiveDirectory">
            a:InvalidPropertyFault
          </soapenv:Value>
        </soapenv:Subcode>
      </soapenv:Code>
      <soapenv:Reason>
        <soapenv:Text xml:lang="en-US">Sorting or Selection Property is
invalid.</soapenv:Text>
      </soapenv:Reason>
      <soapenv:Detail>
        <EnumerateFault
          xmlns="http://schemas.microsoft.com/2008/1/ActiveDirectory"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:xsd="http://www.w3.org/2001/XMLSchema">
          <Error>Selection or Sort property's value is not a valid Active Directory
attribute.</Error>
          <ShortError>InvalidPropertyValueDetail</ShortError>
          <InvalidProperty>
            addata:Invalid_Entry
          </InvalidProperty>
        </EnumerateFault>
      </soapenv:Detail>
    </soapenv:Fault>
  </soapenv:Body>
</soapenv:Envelope>

```



```
    </EnumerateFault>  
  </soapenv:Detail>  
</soapenv:Fault>  
</soapenv:Body>  
</soapenv:Envelope>
```

## **5 Security**

### **5.1 Security Considerations for Implementers**

There are no known additional security considerations for these protocol extensions, but implementers have to consider the security implications of the data that they expose by way of these extensions. When exposing access to directory objects that can contain sensitive information, server implementers are encouraged to use transport mechanisms that support encryption and integrity-verification of the messages. Server implementers are also encouraged to enforce access controls prior to performing any operation against a directory object.

### **5.2 Index of Security Parameters**

This protocol extension has no security parameters.

## 6 Appendix A: WSDL

The [MS-WSDS] extensions to the WS-Enumeration [WSENUM] protocol do not define a WSDL of their own, nor do they extend the [WSENUM] WSDL. The schema of extension elements defined by the [MS-WSDS] protocol is specified in Appendix B: Schema. For a server to implement this protocol, it uses the full WSDL definition of the [WSENUM] protocol, while substituting the schema definitions in Appendix B: Schema for similarly named XML elements, attributes, complex types, and so on, shown in the [WSENUM] WSDL.

## 7 Appendix B: Schema

This section provides the additional schema elements for the extensions defined by the [MS-WSDS] protocol. For clarity, elements of the WS-Enumeration [WSENUM] schema that are not extended by this protocol do not appear in the extended schema. To obtain the fully-extended [WSENUM] schema, the implementer must include nonextended XML elements, attributes, complex types, and so on, from the [WSENUM] schema.

```
<!--[MS-WSDS] filter extension elements schema-->

<xsd:schema
  xmlns:adlq="http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/LdapQuery"
    attributeFormDefault="unqualified"
    elementFormDefault="qualified"

  targetNamespace="http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/LdapQuery"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">

  <xsd:element name = "LdapQuery">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="Filter" type="xsd:string"
          minOccurs="1" maxOccurs="1" />
        <xsd:element name="BaseObject" type="xsd:string"
          minOccurs="1" maxOccurs="1" />
        <xsd:element name="Scope" type="xsd:string"
          minOccurs="1" maxOccurs="1" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>

<!--[MS-WSDS] selection and sorting extension elements schema-->

<xsd:schema xmlns:ad="http://schemas.microsoft.com/2008/1/ActiveDirectory"
  xmlns:addata="http://schemas.microsoft.com/2008/1/ActiveDirectory/Data"
  attributeFormDefault="unqualified"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/2008/1/ActiveDirectory"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">

  <xsd:element name ="Selection">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="SelectionProperty" type="xsd:string" minOccurs="1"
          maxOccurs="unbounded" />
      </xsd:sequence>
      <xsd:attribute name="Dialect"
        fixed="http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/XPath-
Level-1" use="required"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name ="Sorting">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="SortingProperty" minOccurs="1"
          maxOccurs="1">
          <xsd:complexType>
            <xsd:simpleContent>
              <xsd:extension base="xsd:string">
                <xsd:attribute name="Ascending" use="optional"
                  type="xsd:boolean" default="true"/>
              </xsd:extension>
            </xsd:simpleContent>
          </xsd:complexType>
        </xsd:element>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

```

        </xsd:element>
    </xsd:sequence>
    <xsd:attribute name="Dialect"
fixed="http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/XPath-Level-1"
use="required"/>
    </xsd:complexType>
</xsd:element>
</xsd:schema>

<!--Extended [WSENUM] Schema-->
<xsd:schema
    targetNamespace="http://schemas.xmlsoap.org/ws/2004/09/enumeration"
    xmlns:wsen="http://schemas.xmlsoap.org/ws/2004/09/enumeration"
    xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:adlq="http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/LdapQuery"
    xmlns:ad="http://schemas.microsoft.com/2008/1/ActiveDirectory"
    xmlns:adata="http://schemas.microsoft.com/2008/1/ActiveDirectory/Data"
    elementFormDefault="qualified"
    blockDefault="#all">

    <xsd:import namespace="http://schemas.xmlsoap.org/ws/2004/08/addressing"
        schemaLocation="http://schemas.xmlsoap.org/ws/2004/08/addressing"/>

    <xsd:import
namespace="http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/LdapQuery"/>
    <xsd:import namespace="http://schemas.microsoft.com/2008/1/ActiveDirectory"/>

    <!--Extended filter type for Enumerate [WSENUM] request-->

    <xsd:complexType name="FilterType">
        <xsd:sequence>
            <xsd:element minOccurs="0" maxOccurs="1"
                ref="adlq:LdapQuery" />
        </xsd:sequence>
        <xsd:attribute name="Dialect"

fixed="http://schemas.microsoft.com/2008/1/ActiveDirectory/Dialect/LdapQuery"/>
    </xsd:complexType>

    <!--

    Other elements from [WSENUM] schema in section <!-- Types and global elements -->:
    PositiveDurationType (simpleType)
    NonNegativeDurationType (simpleType)
    ExpirationType (simpleType)
    EnumerationContextType (complexType)
    ItemListType (complexType)
    LanguageSpecificStringType (complexType)

-->

<!--Extended Enumerate [WSENUM] request-->
<xsd:element name="Enumerate">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="EndTo" type="wsa:EndpointReferenceType" minOccurs="0" />
            <xsd:element name="Expires" type="wsen:ExpirationType" minOccurs="0" />
            <xsd:element name="Filter" type="wsen:FilterType" minOccurs="0" maxOccurs="1" />
            <xsd:element ref="ad:Selection"
                minOccurs="0" maxOccurs="1" />
            <xsd:element ref="ad:Sorting"
                minOccurs="0" maxOccurs="1" />
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

<!--

    Other elements from [MS-WSPELD] schema in Appendix B:

```

```

ad:Controls (element)

-->

<!--Extended PULL [WSENUM] request-->
<xsd:element name="Pull">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="EnumerationContext"
        type="tns:EnumerationContextType" />
      <xsd:element name="MaxTime"
        type="tns:PositiveDurationType"
        minOccurs="0" />
      <xsd:element name="MaxElements"
        type="xsd:positiveInteger"
        minOccurs="0" />
      <xsd:element name="MaxCharacters"
        type="xsd:positiveInteger"
        minOccurs="0" />
      <xsd:element ref="ad:Controls"
        minOccurs="0"
        maxOccurs="1" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

<!--

Other elements from [WSENUM] schema, the following sections in [WSENUM]

Section <!-- Used for a fault response -->
  SupportedDialect (element)

Section <!-- Enumerate response -->
  EnumerateResponse (element)

Section <!-- Pull response -->
  PullResponse (element)

Section <!-- Renew request -->
  Renew (element)

Section <!-- Renew response -->
  RenewResponse (element)

Section <!-- GetStatus request -->
  GetStatus (element)

Section <!-- GetStatus response -->
  GetStatusResponse (element)

Section <!-- Release request -->
  Release (element)

Section <!-- EnumerationEnd message -->
  EnumerationEnd (element)
  EnumerationEndCodeType (simpleType)
  OpenEnumerationEndCodeType (simpleType)

-->

</xsd:schema>

```

## 8 Appendix C: 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~~updates to those products.

The terms "earlier" and "later", when used with a product version, refer to either all preceding versions or all subsequent versions, respectively. The term "through" refers to the inclusive range of versions. Applicable Microsoft products are listed chronologically in this section.

- Windows Server 2008 R2 operating system
- Windows Server 2012 operating system
- Windows Server 2012 R2 operating system
- Windows Server 2016 operating system
- Windows Server operating system

Exceptions, if any, are noted ~~below~~in this section. If ~~a~~an update version, service pack or ~~Quick-Fix Engineering (QFE)~~Knowledge Base (KB) number appears with ~~the~~a product ~~version~~name, the behavior changed in that ~~service pack or QFE update~~. The new behavior also applies to subsequent ~~service packs of the product~~updates 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.

<1> Section 2.1: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions use SOAP 1.2 [SOAP1.2-1/2003] over a NET.TCP [MC-NMF] transport binding. The transports used, as well as the authentication mechanisms supported and the endpoints exposed, are specified in [MS-ADDM] section 2.1.

<2> Section 3: The following products are applicable to WS-Enumeration: Directory Services Protocol Extensions:

- Active Directory Management Gateway Service contains the server implementation of WS-Enumeration: Directory Services Protocol Extensions.
- Remote Server Administration Tools (excluding Remote Server Administration Tools for Windows Vista operating system) contains the client implementation. For more information about Remote Server Administration Tools, see [MSFT-RSAT].
- Windows Server 2008 R2 and later have both the server and the client implementations.

Active Directory Management Gateway Service is available for Windows Server 2003 operating system with Service Pack 2 (SP2), Windows Server 2003 R2 operating system with Service Pack 2 (SP2), and Windows Server 2008 operating system.

<3> Section 3: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions limit the validity of an enumeration context to 30 minutes by default (defined as MaxEnumContextExpiration in the configuration settings).

<4> Section 3.1.1: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions use the mapping defined in [MS-ADDM] section 2.3.

<5> Section 3.1.3: All the endpoints of Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions including the Enumeration Interface specified in [MS-ADDM] section 2.1 listen on fixed TCP port 9389.

<6> Section 3.1.4: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions do not support sending EnumerationEnd SOAP message (defined in WS-Enumeration [WSENUM] section 3.6) to the endpoint reference, when the enumeration terminates unexpectedly.

<7> Section 3.1.4.1: If the expiration time is not provided in the enumerate request, Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions assign the default value of 5 minutes duration to the expiration time.

<8> Section 3.1.4.1: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions always return the expiration time in the enumerate response as an absolute time in Coordinated Universal Time (UTC) mode.

<9> Section 3.1.4.1.1.2: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions return all of the default attributes and all of the synthetic attributes (defined in [MS-ADDM] section 2.3.3) when there is no selection element present in the Enumerate request.

<10> Section 3.1.4.1.1.2.1: If selection property <ad:all> is specified in the Enumerate request, Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions return all of the default attributes and the synthetic attribute <ad:objectReferenceProperty> (defined in [MS-ADDM] section 2.3.3) in the SOAP response message for the Pull operation. In addition to <ad:all>, if certain constructed attributes are also specified as part of selection properties in the Enumerate request, then Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions also return the requested constructed attributes in the SOAP response message for the Pull operation.

<11> Section 3.1.4.1.3.1: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions allow no more than 5 enumeration contexts to be created per client-to-server session by default (defined as MaxEnumCtxsPerSession in the configuration settings).

They allow no more than 100 enumeration contexts in total for the entire set of client-to-server sessions open at a time by default (defined as MaxEnumCtxsTotal in the configuration settings).

<12> Section 3.1.4.1.3.1: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions use the fault detail defined in [MS-ADDM] section 2.6 for ad:EnumerationContextLimitExceeded fault.

<13> Section 3.1.4.1.3.4: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions have the following four synthetic attributes: <ad:container-hierarchy-parent>, <ad:distinguishedName>, <ad:relativeDistinguishedName>, <ad:objectReferenceProperty>. See [MS-ADDM] section 2.3.3 for their definitions.

<14> Section 3.1.4.1.3.4: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions use the fault detail defined in [MS-ADDM] section 2.6 for an ad:InvalidSortKey fault.

<15> Section 3.1.4.1.3.5: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions use an element named ad:PullFault of the complex type ad:, which is defined in [MS-ADDM] section 2.6, for the wsen:CannotProcessFilter fault.

<16> Section 3.1.4.1.3.6: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions perform validations on the adlq:LdapQuery element of a wsen:Enumerate request during the corresponding wsen:pull request processing. If an invalid adlq:LdapQuery element is received on a wsen:Enumerate request, a wsa2004:EndPointUnavailable fault is returned while processing the corresponding wsen:pull request.

<17> Section 3.1.4.1.3.6: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions use the element ad:FaultDetail, which is defined in [MS-ADDM] section 2.6, for wsa2004:EndPointUnavailable faults.



<18> Section 3.1.4.2: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions do not limit the number of characters in the Pull response by not supporting the wsen:MaxCharacters element in the client request.

<19> Section 3.1.4.2.1.1: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions use an element named ad:PullFault of the complex type ad:FaultDetailType, which is defined in [MS-ADDM] section 2.6, for an ad:MaxCharsNotSupported fault.

<20> Section 3.1.4.2.1.2: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions use the element ad:FaultDetail, which is defined in [MS-ADDM] section 2.6, for wsen:InvalidEnumerationContext faults.

<21> Section 3.1.4.2.1.3: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions use the element ad:FaultDetail, which is defined in [MS-ADDM] section 2.6, for wsa2004:DestinationUnreachable faults.

<22> Section 3.1.4.2.1.3: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions generate a receiver fault rather than a sender fault as specified in the section.

<23> Section 3.1.4.2.1.4: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions use the element ad:FaultDetail, which is defined in [MS-ADDM] section 2.6, for wsa2004:EndpointUnavailable faults.

<24> Section 3.1.4.2.1.5: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions use the element ad:FaultDetail, which is defined in [MS-ADDM] section 2.6, for ad:MaxTimeExceedsLimit faults.

<25> Section 3.1.4.3.1.1: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions use the element ad:FaultDetail, which is defined in [MS-ADDM] section 2.6, for wsen:InvalidEnumerationContext faults.

<26> Section 3.1.4.3.1.2: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions use the element ad:FaultDetail, which is defined in [MS-ADDM] section 2.6, for wsa2004:EndpointUnavailable faults.

<27> Section 3.1.4.4.1.1: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions use the element ad:FaultDetail, which is defined in [MS-ADDM] section 2.6, for wsen:InvalidEnumerationContext faults.

<28> Section 3.1.4.4.1.2: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions use the element ad:FaultDetail, which is defined in [MS-ADDM] section 2.6, for wsa2004:EndpointUnavailable faults.

<29> Section 3.1.4.5.1.1: Microsoft implementations of WS-Enumeration: Directory Services Protocol Extensions use the element ad:FaultDetail, which is defined in [MS-ADDM] section 2.6, for wsa2004:EndpointUnavailable faults.

## 9 Change Tracking

This section identifies changes that were made to this document since the last release. Changes are classified as Major, Minor, or None.

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.
- A document revision that captures changes to protocol functionality.

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 **None** means that no new technical changes were introduced. Minor editorial and formatting changes may have been made, but the relevant technical content is identical to the last released version.

The changes made to this document are listed in the following table. For more information, please contact dohelp@microsoft.com.

Section	Description	Revision class
<a href="#">3-Protocol-Details8</a> <a href="#">Appendix C: Product Behavior</a>	Added <del>information about which</del> Windows Server operating system to the list of applicable products <del>implement which protocol roles.</del>	Major

## 10 Index

### A

- Abstract data model 13
  - server 13
- Applicability 9
- Attribute groups 12
- Attributes 12

### C

- Capability negotiation 10
- Change tracking 42
- Complex types 12

### D

- Data model - abstract 13
  - server 13
- Directory service schema elements 12

### E

- Elements 12
- Elements - directory service schema 12
- Enumerate request example 29
- Enumerate response example 30
- Events
  - local - server 28
  - timer - server 28
- Examples
  - "Enumerate" request 29
  - "Enumerate" response 30
  - "FaultDetail" 32
  - "Pull" request example 30
  - "Pull" response 31
  - overview 29

### F

- FaultDetail example 32
- Fields - vendor-extensible 10

### G

- Glossary 6
- Groups 12

### I

- Implementer - security considerations 34
- Index of security parameters 34
- Informative references 8
- Initialization 14
  - server 14
- Introduction 6

### L

- Local events 28
  - server 28

## **M**

- Message processing
  - overview 14
  - server 14
  - wsen:Enumerate 15
  - wsen:GetStatus 27
  - wsen:Pull 23
  - wsen:Release 28
  - wsen:Renew 26

### Messages

- attribute groups 12
- attributes 12
- complex types 12
- directory service schema elements 12
- elements 12
- enumerated 11
- groups 12
- namespaces 11
- simple types 12
- syntax 11
  - attribute groups 12
  - attributes 12
  - complex types 12
  - elements 12
  - groups 12
  - messages 11
  - namespaces 11
  - overview 11
  - simple types 12
- transport 11

## **N**

- Namespaces 11
- Normative references 7

## **O**

- Operations
  - wsen:Enumerate 15
  - wsen:GetStatus 27
  - wsen:Pull 23
  - wsen:Release 28
  - wsen:Renew 26
- Overview (synopsis) 9

## **P**

- Parameters - security index 34
- Preconditions 9
- Prerequisites 9
- Product behavior 39
- Protocol Details
  - overview 13
- Pull request example 30
- Pull response example 31

## **R**

- References 7
  - informative 8
  - normative 7
- Relationship to other protocols 9

## S

Schema elements - directory service 12

Security

    implementer considerations 34

    parameter index 34

Sequencing rules

    overview 14

    server 14

    wsen:Enumerate 15

    wsen:GetStatus 27

    wsen:Pull 23

    wsen:Release 28

    wsen:Renew 26

Server

    abstract data model 13

    initialization 14

    local events 28

    message processing 14

        overview 14

        wsen:Enumerate 15

        wsen:GetStatus 27

        wsen:Pull 23

        wsen:Release 28

        wsen:Renew 26

    overview 13

    sequencing rules 14

        overview 14

        wsen:Enumerate 15

        wsen:GetStatus 27

        wsen:Pull 23

        wsen:Release 28

        wsen:Renew 26

    timer events 28

    timers 14

    wsen:Enumerate operation 15

    wsen:GetStatus operation 27

    wsen:Pull operation 23

    wsen:Release operation 28

    wsen:Renew operation 26

Simple types 12

Standards assignments 10

Syntax

    attribute groups 12

    attributes 12

    complex types 12

    elements 12

    groups 12

    messages 11

    messages - overview 11

    namespaces 11

    overview 11

    simple types 12

## T

Timer events 28

    server 28

Timers 14

    server 14

Tracking changes 42

Transport 11

Types

    complex 12

simple 12

## **V**

Vendor-extensible fields 10

Versioning 10

## **W**

WSDL 35

wsen:Enumerate

attributes

ad:Selection/@Dialect 20

ad:Sorting/@Dialect 20

ad:Sorting/ad:SortingProperty/@Ascending 20

overview 20

elements

ad:Selection 18

ad:Sorting 19

ad:SortingProperty 19

adlq:LdapQuery 16

overview 16

overview 15

SOAP faults

ad:EnumerationContextLimitExceeded 20

ad:InvalidPropertyFault 21

ad:InvalidSortKey 22

ad:UnsupportedSelectOrSortDialectFault 21

overview 20

wsen:GetStatus 27

wsen:Pull

overview 23

SOAP faults

ad:MaxCharsNotSupported 24

overview 24

wsen:Release 28

wsen:Renew 26