

[MS-CPSWS]:

SharePoint Claim Provider Web Service Protocol

Intellectual Property Rights Notice for Open Specifications Documentation

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

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

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

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

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

Preliminary

Revision Summary

Date	Revision History	Revision Class	Comments
7/13/2009	0.1	Major	Initial Availability
8/28/2009	0.2	Editorial	Revised and edited the technical content
11/6/2009	0.3	Editorial	Revised and edited the technical content
2/19/2010	1.0	Major	Updated and revised the technical content
3/31/2010	1.01	Editorial	Revised and edited the technical content
4/30/2010	1.02	Editorial	Revised and edited the technical content
6/7/2010	1.03	Editorial	Revised and edited the technical content
6/29/2010	1.04	Editorial	Changed language and formatting in the technical content.
7/23/2010	1.05	Minor	Clarified the meaning of the technical content.
9/27/2010	1.05	No Change	No changes to the meaning, language, or formatting of the technical content.
11/15/2010	1.05	No Change	No changes to the meaning, language, or formatting of the technical content.
12/17/2010	1.05	No Change	No changes to the meaning, language, or formatting of the technical content.
3/18/2011	1.05	No Change	No changes to the meaning, language, or formatting of the technical content.
1/20/2012	2.0	Major	Significantly changed the technical content.
4/11/2012	2.0	No Change	No changes to the meaning, language, or formatting of the technical content.
7/16/2012	2.0	No Change	No changes to the meaning, language, or formatting of the technical content.
9/12/2012	2.0	No Change	No changes to the meaning, language, or formatting of the technical content.
10/8/2012	2.1	Minor	Clarified the meaning of the technical content.
2/11/2013	2.1	No Change	No changes to the meaning, language, or formatting of the technical content.
7/30/2013	2.1	No Change	No changes to the meaning, language, or formatting of the technical content.
11/18/2013	2.1	No Change	No changes to the meaning, language, or formatting of the technical content.
2/10/2014	2.1	No Change	No changes to the meaning, language, or formatting of the technical content.
4/30/2014	3.0	Major	Significantly changed the technical content.
7/31/2014	4.0	Major	Significantly changed the technical content.
10/30/2014	4.0	No Change	No changes to the meaning, language, or formatting of the technical content.
3/16/2015	5.0	Major	Significantly changed the technical content.

Table of Contents

1	Introduction	9
1.1	Glossary	9
1.2	References	11
1.2.1	Normative References	11
1.2.2	Informative References	11
1.3	Protocol Overview (Synopsis)	12
1.4	Relationship to Other Protocols	12
1.5	Prerequisites/Preconditions	12
1.6	Applicability Statement	13
1.7	Versioning and Capability Negotiation	13
1.8	Vendor-Extensible Fields	13
1.9	Standards Assignments.....	13
2	Messages.....	14
2.1	Transport.....	14
2.2	Common Message Syntax	14
2.2.1	Namespaces	14
2.2.2	Messages.....	14
2.2.3	Elements	14
2.2.4	Complex Types.....	15
2.2.4.1	ArrayOfAnyType.....	15
2.2.4.2	ArrayOfPair	15
2.2.4.3	ArrayOfPickerEntity	16
2.2.4.4	ArrayOfSPPProviderHierarchyNode.....	16
2.2.4.5	ArrayOfSPPProviderHierarchyTree.....	16
2.2.4.6	ArrayOfSPSchemaElement	17
2.2.4.7	ArrayOfString	17
2.2.4.8	Pair	17
2.2.4.9	PickerEntity	17
2.2.4.10	SPClaim	18
2.2.4.11	SPPProviderHierarchyElement	19
2.2.4.12	SPPProviderHierarchyNode.....	20
2.2.4.13	SPPProviderHierarchyTree.....	20
2.2.4.14	SPPProviderSchema	20
2.2.4.15	SPSchemaElement	21
2.2.5	Simple Types	21
2.2.5.1	SPPPrincipalType	21
2.2.5.2	SPSchemaElementType	22
2.2.6	Attributes	22
2.2.7	Groups	22
2.2.8	Attribute Groups.....	23
3	Protocol Details.....	24
3.1	Server Details.....	24
3.1.1	Abstract Data Model.....	24
3.1.2	Timers	25
3.1.3	Initialization.....	25
3.1.4	Message Processing Events and Sequencing Rules	25
3.1.4.1	ClaimTypes	26
3.1.4.1.1	Messages	26
3.1.4.1.1.1	IClaimProviderWebService_ClaimTypes_InputMessage	26
3.1.4.1.1.2	IClaimProviderWebService_ClaimTypes_OutputMessage	27
3.1.4.1.2	Elements.....	27
3.1.4.1.2.1	ClaimTypes	27

3.1.4.1.2.2	ClaimTypesResponse	27
3.1.4.1.3	Complex Types	27
3.1.4.1.4	Simple Types	27
3.1.4.1.5	Attributes	27
3.1.4.1.6	Groups.....	28
3.1.4.1.7	Attribute Groups.....	28
3.1.4.2	ClaimValueTypes.....	28
3.1.4.2.1	Messages	28
3.1.4.2.1.1	IClaimProviderWebService_ClaimValueTypes_InputMessage.....	29
3.1.4.2.1.2	IClaimProviderWebService_ClaimValueTypes_OutputMessage	29
3.1.4.2.2	Elements	29
3.1.4.2.2.1	ClaimValueTypes	29
3.1.4.2.2.2	ClaimValueTypesResponse	29
3.1.4.2.3	Complex Types	30
3.1.4.2.4	Simple Types	30
3.1.4.2.5	Attributes	30
3.1.4.2.6	Groups.....	30
3.1.4.2.7	Attribute Groups.....	30
3.1.4.3	EntityTypes	30
3.1.4.3.1	Messages	30
3.1.4.3.1.1	IClaimProviderWebService_EntityTypes_InputMessage	31
3.1.4.3.1.2	IClaimProviderWebService_EntityTypes_OutputMessage	31
3.1.4.3.2	Elements	31
3.1.4.3.2.1	EntityTypes	31
3.1.4.3.2.2	EntityTypesResponse.....	31
3.1.4.3.3	Complex Types	32
3.1.4.3.4	Simple Types	32
3.1.4.3.5	Attributes	32
3.1.4.3.6	Groups.....	32
3.1.4.3.7	Attribute Groups.....	32
3.1.4.4	GetHierarchy	32
3.1.4.4.1	Messages	33
3.1.4.4.1.1	IClaimProviderWebService_GetHierarchy_InputMessage	33
3.1.4.4.1.2	IClaimProviderWebService_GetHierarchy_OutputMessage	33
3.1.4.4.2	Elements.....	33
3.1.4.4.2.1	GetHierarchy	33
3.1.4.4.2.2	GetHierarchyResponse.....	34
3.1.4.4.3	Complex Types	34
3.1.4.4.4	Simple Types	34
3.1.4.4.5	Attributes	34
3.1.4.4.6	Groups.....	34
3.1.4.4.7	Attribute Groups.....	35
3.1.4.5	GetHierarchyAll.....	35
3.1.4.5.1	Messages	35
3.1.4.5.1.1	IClaimProviderWebService_GetHierarchyAll_InputMessage.....	36
3.1.4.5.1.2	IClaimProviderWebService_GetHierarchyAll_OutputMessage	36
3.1.4.5.2	Elements	36
3.1.4.5.2.1	GetHierarchyAll	36
3.1.4.5.2.2	GetHierarchyAllResponse	37
3.1.4.5.3	Complex Types	37
3.1.4.5.4	Simple Types	37
3.1.4.5.5	Attributes	37
3.1.4.5.6	Groups.....	37
3.1.4.5.7	Attribute Groups.....	37
3.1.4.6	HierarchyProviderSchema	37
3.1.4.6.1	Messages	38
3.1.4.6.1.1	IClaimProviderWebService_HierarchyProviderSchema_InputMessage	38

3.1.4.6.1.2	IClaimProviderWebService_HierarchyProviderSchema_OutputMessage	38
3.1.4.6.2	Elements	38
3.1.4.6.2.1	HierarchyProviderSchema	39
3.1.4.6.2.2	HierarchyProviderSchemaResponse	39
3.1.4.6.3	Complex Types	39
3.1.4.6.4	Simple Types	39
3.1.4.6.5	Attributes	39
3.1.4.6.6	Groups	39
3.1.4.6.7	Attribute Groups	39
3.1.4.7	ProviderSchemas	39
3.1.4.7.1	Messages	40
3.1.4.7.1.1	IClaimProviderWebService_ProviderSchemas_InputMessage	40
3.1.4.7.1.2	IClaimProviderWebService_ProviderSchemas_OutputMessage	40
3.1.4.7.2	Elements	40
3.1.4.7.2.1	ProviderSchemas	41
3.1.4.7.2.2	ProviderSchemasResponse	41
3.1.4.7.3	Complex Types	41
3.1.4.7.3.1	ArrayOfSPPProviderSchema	41
3.1.4.7.4	Simple Types	42
3.1.4.7.5	Attributes	42
3.1.4.7.6	Groups	42
3.1.4.7.7	Attribute Groups	42
3.1.4.8	Resolve	42
3.1.4.8.1	Messages	42
3.1.4.8.1.1	IClaimProviderWebService_Resolve_InputMessage	43
3.1.4.8.1.2	IClaimProviderWebService_Resolve_OutputMessage	43
3.1.4.8.2	Elements	43
3.1.4.8.2.1	Resolve	43
3.1.4.8.2.2	ResolveResponse	44
3.1.4.8.3	Complex Types	44
3.1.4.8.4	Simple Types	44
3.1.4.8.5	Attributes	44
3.1.4.8.6	Groups	44
3.1.4.8.7	Attribute Groups	44
3.1.4.9	ResolveClaim	44
3.1.4.9.1	Messages	45
3.1.4.9.1.1	IClaimProviderWebService_ResolveClaim_InputMessage	45
3.1.4.9.1.2	IClaimProviderWebService_ResolveClaim_OutputMessage	45
3.1.4.9.2	Elements	45
3.1.4.9.2.1	ResolveClaim	45
3.1.4.9.2.2	ResolveClaimResponse	46
3.1.4.9.3	Complex Types	46
3.1.4.9.4	Simple Types	46
3.1.4.9.5	Attributes	46
3.1.4.9.6	Groups	46
3.1.4.9.7	Attribute Groups	46
3.1.4.10	ResolveMultiple	47
3.1.4.10.1	Messages	47
3.1.4.10.1.1	IClaimProviderWebService_ResolveMultiple_InputMessage	47
3.1.4.10.1.2	IClaimProviderWebService_ResolveMultiple_OutputMessage	47
3.1.4.10.2	Elements	48
3.1.4.10.2.1	ResolveMultiple	48
3.1.4.10.2.2	ResolveMultipleResponse	48
3.1.4.10.3	Complex Types	49
3.1.4.10.4	Simple Types	49
3.1.4.10.5	Attributes	49

3.1.4.10.6	Groups	49
3.1.4.10.7	Attribute Groups	49
3.1.4.11	ResolveMultipleClaim	49
3.1.4.11.1	Messages	49
3.1.4.11.1.1	IClaimProviderWebService_ResolveMultipleClaim_InputMessage	50
3.1.4.11.1.2	IClaimProviderWebService_ResolveMultipleClaim_OutputMessage	50
3.1.4.11.2	Elements	50
3.1.4.11.2.1	ResolveMultipleClaim	50
3.1.4.11.2.2	ResolveMultipleClaimResponse	51
3.1.4.11.3	Complex Types	51
3.1.4.11.3.1	ArrayOfSPClaim	51
3.1.4.11.4	Simple Types	51
3.1.4.11.5	Attributes	52
3.1.4.11.6	Groups	52
3.1.4.11.7	Attribute Groups	52
3.1.4.12	Search	52
3.1.4.12.1	Messages	52
3.1.4.12.1.1	IClaimProviderWebService_Search_InputMessage	53
3.1.4.12.1.2	IClaimProviderWebService_Search_OutputMessage	53
3.1.4.12.2	Elements	53
3.1.4.12.2.1	Search	53
3.1.4.12.2.2	SearchResponse	54
3.1.4.12.3	Complex Types	54
3.1.4.12.3.1	ArrayOfSPPProviderSearchArguments	54
3.1.4.12.3.2	SPPProviderSearchArguments	54
3.1.4.12.4	Simple Types	55
3.1.4.12.5	Attributes	55
3.1.4.12.6	Groups	55
3.1.4.12.7	Attribute Groups	55
3.1.4.13	SearchAll	55
3.1.4.13.1	Messages	56
3.1.4.13.1.1	IClaimProviderWebService_SearchAll_InputMessage	56
3.1.4.13.1.2	IClaimProviderWebService_SearchAll_OutputMessage	56
3.1.4.13.2	Elements	56
3.1.4.13.2.1	SearchAll	56
3.1.4.13.2.2	SearchAllResponse	57
3.1.4.13.3	Complex Types	57
3.1.4.13.4	Simple Types	57
3.1.4.13.5	Attributes	57
3.1.4.13.6	Groups	57
3.1.4.13.7	Attribute Groups	57
3.1.5	Timer Events	58
3.1.6	Other Local Events	58
4	Protocol Examples	59
4.1	GetHierarchyAll	59
4.1.1	Retrieve Claim Provider Hierarchy Trees from All Available Claim Providers	59
4.1.2	Retrieve Claim Provider Hierarchy Trees from One of the Claim Providers	60
4.2	SearchAll	61
4.3	Search	63
4.4	GetHierarchy	64
4.5	EntityTypes	65
4.6	ClaimTypes	65
4.7	ClaimValueTypes	66
4.8	Resolve	67
4.9	ResolveMultiple	68

5	Security	71
5.1	Security Considerations for Implementers	71
5.2	Index of Security Parameters	71
6	Appendix A: Full WSDL	72
7	Appendix B: Full XML Schema	78
8	Appendix C: Product Behavior	84
9	Change Tracking	85
10	Index	87

Preliminary

1 Introduction

This document specifies the SharePoint Claim Provider Web Service Protocol. This protocol enables a protocol client to access claims, which are a set of statements one subject makes about itself or another subject. Claims are returned by the claims providers that are available on a protocol server. A typical scenario for this protocol is a client application that allows users to search and resolve users and groups from a set of directory services and external identity providers.

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

1.1 Glossary

The following terms are specific to this document:

claim: (1) A set of operations that are performed on a workflow task to specify the user who owns it.

(2) A statement that one subject makes about itself or another subject. For example, the statement can be about a name, identity, key, group, privilege, or capability. Claims have a provider that issues them, and they are given one or more values. They are also defined by a claim value type and, possibly, associated metadata.

claim type: A statement that is part of a **claim (2)** and provides context for a claim value. It represents the type of claim and is typically a **Uniform Resource Identifier (URI)**. Examples include FirstName and Role.

claim value: A string that represents the value of a statement in a **claim (2)**. It specifies what is being asserted by a claim.

claim-based authentication mode: A set of operations that is used to establish trust relationships between **claims providers** and relying party applications. It involves the exchange of identifying certificates (1) that make it possible for a relying party to trust the content of a **claim (2)** that is issued by a claims provider.

claims provider: A software component or service that can be used to issue a **claim (2)** during sign-in operations and to display, resolve, and provide search capabilities for claims in a card selector.

claims provider schema: A schema that is used to specify which fields can be returned as metadata for a **claim (2)** that is issued by a specific claims provider.

directory service (DS): A service that stores and organizes information about a computer network's users and network shares, and that allows network administrators to manage users' access to the shares. See also Active Directory.

group: A named collection of users who share similar access permissions or roles.

hierarchy provider: A code structure that defines a hierarchy.

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

Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS): An extension of **HTTP** that securely encrypts and decrypts webpage requests.

login name: A string that is used to identify a user or entity to an operating system, directory service, or distributed system. For example, in Windows-integrated authentication, a login name uses the form "DOMAIN\username".

site: A group of related pages and data within a SharePoint site collection. The structure and content of a site is based on a site definition. Also referred to as SharePoint site and web site.

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

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

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

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

Status-Code: A 3-digit integer result code in an HTTP response message, as described in [\[RFC2616\]](#).

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 [\[RFC3986\]](#).

Uniform Resource Locator (URL): A string of characters in a standardized format that identifies a document or resource on the World Wide Web. The format is as specified in [\[RFC1738\]](#).

web application: (1) A container in a configuration database that stores administrative settings and entry-point **URLs** for site collections.

(2) A software application that uses **HTTP** as its core communication protocol and delivers information to the user by using web-based languages such as HTML and XML.

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

WSDL message: An abstract, typed definition of the data that is communicated during a **WSDL operation**, as described in [\[WSDL\]](#).

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

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

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

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

XML schema definition (XSD): The World Wide Web Consortium (W3C) standard language that is used in defining XML schemas. Schemas are useful for enforcing structure and constraining the types of data that can be used validly within other XML documents. XML schema definition refers to the fully specified and currently recommended standard for use in authoring **XML schemas**.

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

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.

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

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

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

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

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

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

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

1.2.2 Informative References

[RFC1738] Berners-Lee, T., Masinter, L., and McCahill, M., Eds., "Uniform Resource Locators (URL)", RFC 1738, December 1994, <http://www.ietf.org/rfc/rfc1738.txt>

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

[RFC3986] Berners-Lee, T., Fielding, R., and Masinter, L., "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005, <http://www.ietf.org/rfc/rfc3986.txt>

[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>

[SOAP1.2-1/2007] Gudgin, M., Hadley, M., Mendelsohn, N., et al., "SOAP Version 1.2 Part 1: Messaging Framework (Second Edition)", W3C Recommendation 27, April 2007, <http://www.w3.org/TR/2007/REC-soap12-part1-20070427/>

[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/>

[XML] World Wide Web Consortium, "Extensible Markup Language (XML) 1.0 (Fourth Edition)", W3C Recommendation 16 August 2006, edited in place 29 September 2006, <http://www.w3.org/TR/2006/REC-xml-20060816/>

1.3 Protocol Overview (Synopsis)

This protocol enables a protocol client to access **claims (2)** returned by the **claims providers** available for a specific **Web application (1)**. A typical scenario for this protocol is a client application that allows users to browse, search, and resolve people and groups from a set of **directory services (DS)** and external identity providers.

This protocol allows a protocol client to send a search query of claims providers and obtain information necessary to store a reference to each claim.

Claims providers can also return metadata relevant to each claim and return claims in hierarchical trees for better presentation by the protocol client.

1.4 Relationship to Other Protocols

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

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

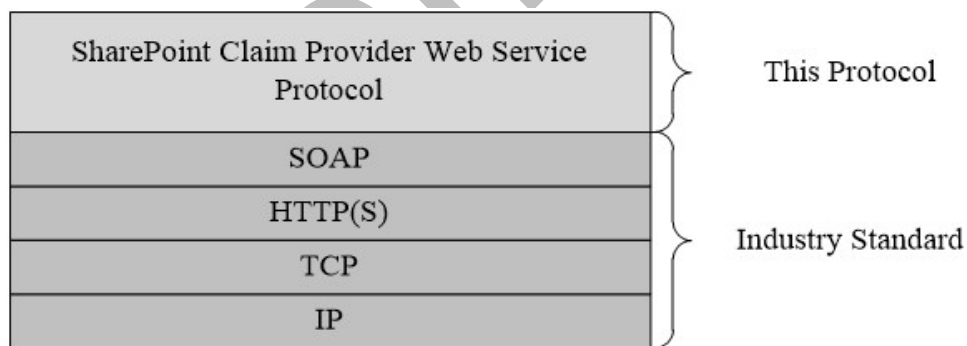


Figure 1: This protocol in relation to other protocols

1.5 Prerequisites/Preconditions

This protocol operates against a site that is identified by a **URL** that is known by protocol clients. When using HTTP transport, the protocol server endpoint is formed by appending `"/_vti_bin/SPClaimProviderWebService.svc"` to the URL of the site, for example: `http://www.contoso.com/_vti_bin/SPClaimProviderWebService.svc`.

When using HTTPS transport, the protocol server endpoint is formed by appending "/_vti_bin/SPClaimProviderWebService.https.svc" to the URL of the site, for example: https://www.contoso.com/_vti_bin/SPClaimProviderWebService.https.svc.

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

1.6 Applicability Statement

This protocol is applicable when a client application allows users to browse, search, and resolve people and groups from a set of directory services (DS) and external identity providers through claims providers.

This protocol does not specify how the claims providers interface with the directory services (DS) or identity providers or how that data is stored.

1.7 Versioning and Capability Negotiation

None.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

In the following sections, the schema definition might differ from the processing rules imposed by the protocol. The **WSDL** in this specification matches the WSDL that shipped with the product and provides a base description of the schema. The text that introduces the WSDL might specify differences that reflect actual Microsoft product behavior. For example, the schema definition might allow for an element to be **empty**, **null**, or **not present** but the behavior of the protocol as specified restricts the same elements to being **non-empty**, **not null**, and **present**.

2.1 Transport

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

Protocol messages **MUST** be formatted as specified in [\[SOAP1.1\]](#), section 4. Protocol server faults **MUST** be returned using HTTP **Status-Codes** as specified in [\[RFC2616\]](#), section 10 or using **SOAP faults** as specified in [\[SOAP1.1\]](#), section 4.4.

2.2 Common Message Syntax

This section contains common structures used by this protocol. The syntax of the structures uses **XML schema definition (XSD)**, as specified in [\[XMLSCHEMA1\]](#) and [\[XMLSCHEMA2\]](#), and Web Services Description Language (WSDL), as specified in [\[WSDL\]](#).

2.2.1 Namespaces

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

Prefix	Namespace URI	Reference
soap	http://schemas.xmlsoap.org/wsdl/soap/	[SOAP1.1]
tns	http://schemas.microsoft.com/sharepoint/claims/	
tns1	http://schemas.microsoft.com/sharepoint/claims/Imports	
wsaw	http://www.w3.org/2006/05/addressing/wsdl	
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL]
xs	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1] [XMLSCHEMA2]

2.2.2 Messages

This specification does not define any common **WSDL message** definitions.

2.2.3 Elements

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

2.2.4 Complex Types

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

Complex type	Description
ArrayOfAnyType	An array of elements of any type.
ArrayOfPair	An array of elements of type Pair (section 2.2.4.7).
ArrayOfPickerEntity	An array of elements of type PickerEntity (section 2.2.4.8).
ArrayOfSPPProviderHierarchyNode	An array of elements of type SPPProviderHierarchyNode (section 2.2.4.12).
ArrayOfSPPProviderHierarchyTree	An array of elements of type SPPProviderHierarchyTree (section 2.2.4.10).
ArrayOfSPSchemaElement	An array of SPSchemaElement (section 2.2.4.12) elements.
ArrayOfString	An array of elements of type string.
Pair	A collection of two named elements of any type in a sequence.
PickerEntity	An object containing basic information about a picker entity.
SPClaim	The claim (2) associated with a login name .
SPPProviderHierarchyElement	Defines the base type for SPPProviderHierarchyNode and SPPProviderHierarchyTree .
SPPProviderHierarchyNode	A claims provider hierarchy node.
SPPProviderHierarchyTree	A claims provider hierarchy tree.
SPPProviderSchema	The user interface display characteristics of a claims provider.
SPSchemaElement	The user interface display characteristics of a field in a picker entity.

2.2.4.1 ArrayOfAnyType

Namespace: <http://schemas.microsoft.com/sharepoint/claims/>

An array of elements of any type.

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

anyType: An anyType value as specified in [\[XMLSCHEMA1\]](#).

2.2.4.2 ArrayOfPair

Namespace: <http://schemas.microsoft.com/sharepoint/claims/>

An array of elements of type **Pair** (section [2.2.4.7](#)).

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

```
</xs:complexType>
```

Pair: A collection of two elements.

2.2.4.3 ArrayOfPickerEntity

Namespace: <http://schemas.microsoft.com/sharepoint/claims/>

An array of elements of type **PickerEntity** (section [2.2.4.8](#)).

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

PickerEntity: An element containing information about a picker entity.

2.2.4.4 ArrayOfSPPProviderHierarchyNode

Namespace: <http://schemas.microsoft.com/sharepoint/claims/>

An array of elements of type **SPPProviderHierarchyNode** (section [2.2.4.12](#)).

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

SPPProviderHierarchyNode: An element containing information about a provider hierarchy node.

2.2.4.5 ArrayOfSPPProviderHierarchyTree

Namespace: <http://schemas.microsoft.com/sharepoint/claims/>

An array of elements of type **SPPProviderHierarchyTree** (section [2.2.4.10](#)).

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

SPPProviderHierarchyTree: An element containing information about a claims provider hierarchy tree.

2.2.4.6 ArrayOfSPSchemaElement

Namespace: http://schemas.microsoft.com/sharepoint/claims/

An array of **SPSchemaElement** (section [2.2.4.12](#)) elements.

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

SPSchemaElement: The user interface display characteristics of a field in a login name.

2.2.4.7 ArrayOfString

Namespace: http://schemas.microsoft.com/sharepoint/claims/

An array of elements of type string.

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

string: A string as specified in [\[XMLSCHEMA1\]](#).

2.2.4.8 Pair

Namespace: http://schemas.microsoft.com/sharepoint/claims/

A collection of two named elements of any type in a sequence.

```
<xs:complexType name="Pair" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="First"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Second"/>
  </xs:sequence>
</xs:complexType>
```

First: The element at the first position.

Second: The element at the second position.

2.2.4.9 PickerEntity

Namespace: http://schemas.microsoft.com/sharepoint/claims/

An object containing basic information about a picker entity.

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

```

    <xs:element minOccurs="0" maxOccurs="1" name="DisplayText" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="IsResolved" type="xs:boolean"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Description" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="EntityType" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="EntityGroupName" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="HierarchyIdentifier"/>
    <xs:element minOccurs="0" maxOccurs="1" name="EntityDataElements"
type="tns:ArrayOfPair"/>
    <xs:element minOccurs="0" maxOccurs="1" name="MultipleMatches"
type="tns:ArrayOfAnyType"/>
    <xs:element minOccurs="0" maxOccurs="1" name="ProviderName" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="ProviderDisplayName" type="xs:string"/>
  </xs:sequence>
</xs:complexType>

```

Key: The name that uniquely identifies the picker entity.

DisplayText: The name to be used in the user interface.

IsResolved: A Boolean value that indicates if the picker entity has been successfully resolved to a unique match.

Description: The detailed description for the picker entity.

EntityType: The type for the picker entity.

EntityGroupName: The group name of the picker entity.

HierarchyIdentifier: The identifier of the picker entity in the hierarchy.

EntityDataElements: The list of name-value pairs representing the picker entity data.

MultipleMatches: The list of matched picker entities when there is more than one match.

ProviderName: The name of a claims provider.

ProviderDisplayName: The human-readable display name of a claims provider.

2.2.4.10 SPClaim

Namespace: <http://schemas.microsoft.com/sharepoint/claims/>

The claim (2) associated with a login name.

```

<xs:complexType name="SPClaim" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="ClaimType" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Value" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="ValueType" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="OriginalIssuer" type="xs:string"/>
  </xs:sequence>
</xs:complexType>

```

ClaimType: The **claim type**.

Value: The **claim value**.

ValueType: The type of the claim value [<1>](#).

OriginalIssuer: The original issuer of the claim (2). The original issuer string MUST begin with one of the following strings representing type of the issuer followed by an optional valid identifier of the issuer. The original issuer type and identifier MUST be separated by a single colon (:) character.

Prefix	Comment
Windows	Issuer is of type Windows.
SecurityTokenService	Issuer is of type Security Token Service.
Forms	Issuer is of type Forms. The protocol client MUST specify valid identifier of the issuer.
TrustedProvider	Issuer is of type Trusted Provider. The protocol client MUST specify valid identifier of the issuer.
ClaimProvider	Issuer is of type Claims Provider . The protocol client MUST specify valid identifier of the issuer.

A valid identifier MUST not begin or end with spaces, MUST not contain any control character or character that can be used as XML markup, and MUST not contain any of the following characters: colon (:), comma (,) or semicolon (;).

2.2.4.11 SPPProviderHierarchyElement

Namespace: http://schemas.microsoft.com/sharepoint/claims/

Defines the base type for **SPPProviderHierarchyNode** and **SPPProviderHierarchyTree**.

```
<xs:complexType name="SPPProviderHierarchyElement"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Nm" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="ProviderName" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="HierarchyNodeID" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="IsLeaf" type="xs:boolean"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Children"
type="tns:ArrayOfSPPProviderHierarchyNode"/>
    <xs:element minOccurs="0" maxOccurs="1" name="EntityData"
type="tns:ArrayOfPickerEntity"/>
    <xs:element minOccurs="1" maxOccurs="1" name="Count" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
```

Nm: The human-readable display name.

ProviderName: The name of the claims provider.

HierarchyNodeID: The unique identifier for the element. This value is empty for the hierarchy provider and the root element of the hierarchy tree.

IsLeaf: Whether or not the element is a leaf element. This value is true if and only if the element has no subelements. Otherwise, it is false.

Children: The list of immediate child elements.

EntityData: The **PickerEntity** objects associated with this element.

Count: The number of **PickerEntity** objects in this element and all child elements, calculated recursively. <2>

2.2.4.12 SPPProviderHierarchyNode

Namespace: http://schemas.microsoft.com/sharepoint/claims/

A claims provider hierarchy node.

```
<xs:complexType name="SPPProviderHierarchyNode" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexContent mixed="false">
    <xs:extension base="tns:SPPProviderHierarchyElement"/>
  </xs:complexContent>
</xs:complexType>
```

2.2.4.13 SPPProviderHierarchyTree

Namespace: http://schemas.microsoft.com/sharepoint/claims/

A claims provider hierarchy tree.

```
<xs:complexType name="SPPProviderHierarchyTree" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexContent mixed="false">
    <xs:extension base="tns:SPPProviderHierarchyElement">
      <xs:sequence>
        <xs:element minOccurs="1" maxOccurs="1" name="IsRoot" type="xs:boolean"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

IsRoot: A Boolean value that indicates whether the node root of this tree is also root of the claims provider's complete hierarchy tree.

2.2.4.14 SPPProviderSchema

Namespace: http://schemas.microsoft.com/sharepoint/claims/

The user interface display characteristics of a claims provider.

```
<xs:complexType name="SPPProviderSchema" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="DisplayName" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="ProviderName" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="ProviderSchema"
type="tns:ArrayOfSPSchemaElement"/>
    <xs:element minOccurs="1" maxOccurs="1" name="SupportsHierarchy" type="xs:boolean"/>
  </xs:sequence>
</xs:complexType>
```

DisplayName: The name to be used in user interfaces.

ProviderName: The name of a claims provider.

ProviderSchema: The list of schema elements that comprise the claims provider schema.

SupportsHierarchy: A Boolean value that indicates whether the claims provider supports hierarchy.

2.2.4.15 SPSchemaElement

Namespace: http://schemas.microsoft.com/sharepoint/claims/

The user interface display characteristics of a field in a picker entity.

```
<xs:complexType name="SPSchemaElement" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="DisplayName" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="Type" type="tns:SPSchemaElementType"/>
  </xs:sequence>
</xs:complexType>
```

Name: The name that uniquely identifies the field.

DisplayName: The name to be used for the field in user interfaces.

Type: The display type of a field in a picker entity.

2.2.5 Simple Types

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

Simple type	Description
SPPrincipalType	Contains the value that is used to specify the type(s) of picker entities included in the output claims provider hierarchy tree.
SPSchemaElementType	The display type of a field.

2.2.5.1 SPPrincipalType

Namespace: http://schemas.microsoft.com/sharepoint/claims/

Contains the value that is used to specify the type(s) of picker entities included in the output claims provider hierarchy tree.

```
<xs:simpleType name="SPPrincipalType" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:list>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="None"/>
        <xs:enumeration value="User"/>
        <xs:enumeration value="DistributionList"/>
        <xs:enumeration value="SecurityGroup"/>
        <xs:enumeration value="SharePointGroup"/>
        <xs:enumeration value="All"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:list>
</xs:simpleType>
```

```

    </xs:restriction>
  </xs:simpleType>
</xs:list>
</xs:simpleType>

```

The following table specifies the allowable values for the **SPPrincipalType** simple type.

Value	Meaning
None	No picker entity is included in the output claims provider hierarchy tree.
User	Only picker entities of type user are included in the output claims provider hierarchy tree.
DistributionList	Only picker entities of type Distribution List are included in the output claims provider hierarchy tree.
SecurityGroup	Picker entities of type Security Group and all other types except User and Distribution List are included in the output claims provider hierarchy tree.
SharePointGroup	Picker entities of the site group are included in the output claims provider hierarchy tree.
All	All picker entities are included in the output claims provider hierarchy tree.

2.2.5.2 SPSchemaElementType

Namespace: http://schemas.microsoft.com/sharepoint/claims/

The display type of a field.

```

<xs:simpleType name="SPSchemaElementType" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:restriction base="xs:string">
    <xs:enumeration value="None"/>
    <xs:enumeration value="TableViewOnly"/>
    <xs:enumeration value="DetailViewOnly"/>
    <xs:enumeration value="Both"/>
  </xs:restriction>
</xs:simpleType>

```

The following table specifies the allowable values for the **SPSchemaElementType** simple type.

Value	Meaning
None	The field is not intended to be displayed in the user interface.
TableViewOnly	The field to be displayed in the Table-View only.
DetailViewOnly	The field to be displayed in the Detailed-View only.
Both	The field can be displayed either in Table-View or Detailed-View.

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.

Preliminary

3 Protocol Details

In the following sections, the schema definition might differ from the processing rules imposed by the protocol. The WSDL in this specification matches the WSDL that shipped with the product and provides a base description of the schema. The text that introduces the WSDL might specify differences that reflect actual Microsoft product behavior. For example, the schema definition might allow for an element to be **empty**, **null**, or **not present** but the behavior of the protocol as specified restricts the same elements to being **non-empty**, **not null**, and **present**.

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

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

This protocol allows protocol servers to notify protocol clients of application-level faults using SOAP faults. Except where specified, these SOAP faults are not significant for interoperability, and protocol clients can interpret them in an implementation-specific manner.

This protocol allows protocol servers to perform implementation-specific authorization checks and notify protocol clients of authorization faults either using HTTP Status-Codes or using SOAP faults as specified previously in this section.

3.1 Server Details

All operations consist of a basic request-response pair and the protocol server treats each request as an independent transaction that is unrelated to any previous request.

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.

The protocol server maintains the following data:

- A list of Web applications (2) in **claim-based authentication mode**.
- A list of claims providers available to the protocol client.

Each claims provider maintains the following data:

- A list of claims (2) that the protocol client can search or resolve against. This protocol does not enforce any implementation on how the list of claims is maintained or the search and resolve operations determined by each claims provider.
- A claims provider hierarchy tree: The tree structure representing the hierarchy of claims as determined by the claim provider. For example, an organization tree could contain a top-level "Contoso" node, a "Sales" child node with 23 claims and an "Engineering" node with 18 claims.
- A **claims provider schema**.

Each claim (2) maintains the following data:

- A claim value.
- A claim type.
- A claim value type.

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Message Processing Events and Sequencing Rules

There are no special sequencing rules for the operations defined in this specification.

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

Operation	Description
ClaimTypes	This operation is used to retrieve a list of all possible claim types from a list of claims providers.
ClaimValueTypes	This operation is used to retrieve a list of claim value types from a list of claims providers.
EntityTypes	This operation is used to retrieve a list of all possible picker entity types from a list of claims providers.
GetHierarchy	This operation is used to retrieve a claims provider hierarchy tree from a claims provider.
GetHierarchyAll	This operation is used to retrieve a list of claims provider hierarchy trees from a list of claims providers.
HierarchyProviderSchema	This operation is used to retrieve schema for the current hierarchy provider .
ProviderSchemas	This operation is used to retrieve a list of claims provider schemas from a list of claims providers.
Resolve	This operation is used to resolve an input string to picker entities using a list of claims providers.
ResolveClaim	This operation is used to resolve an input claim (2) to picker entities using a list of claims providers.
ResolveMultiple	This operation is used to resolve a list of strings to picker entities using a list of claims providers.
ResolveMultipleClaim	This operation is used to resolve a list of claims to picker entities using a list of claims providers.
Search	This operation is used to perform a search for entities on a list of claims providers. The input to this operation is a search string and the result of this operation is a list of claims provider hierarchy trees that contain the matched entities. If a hierarchy provider is available, the protocol server MUST also retrieve the hierarchy provider hierarchy tree and MAY move results returned by claims providers to hierarchy provider hierarchy tree.
SearchAll	This operation is used to perform a search for entities on a list of claims providers. The result of this operation is a list of claims provider hierarchy trees. If a hierarchy provider is available, the protocol server MAY move entities returned by claims providers to hierarchy provider hierarchy tree.

3.1.4.1 ClaimTypes

This operation is used to retrieve a list of all possible claim types from a list of claims providers.

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

```
<wsdl:operation name="ClaimTypes" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimTypes"
    message="tns:IClaimProviderWebService_ClaimTypes_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimTypesResponse"
    message="tns:IClaimProviderWebService_ClaimTypes_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_ClaimTypes_InputMessage** (section [3.1.4.1.1.1](#)) request WSDL message and the protocol server responds with an **IClaimProviderWebService_ClaimTypes_OutputMessage** (section [3.1.4.1.1.2](#)) response message.

The protocol server MUST retrieve all known basic claim types. In addition, the protocol server MUST retrieve claim types from claims providers that meet both of the following criteria:

- The claims providers are associated with the Web application (1) specified in the input message.
- The claims providers are listed in the provider names in the input message.

The protocol server will remove the duplicated claim types from the known basic claim types and the claim providers' claim types.

If the list of provider names is NULL, the protocol server MUST use all the available claims providers.

3.1.4.1.1 Messages

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

Message	Description
IClaimProviderWebService_ClaimTypes_InputMessage	The request WSDL message for the ClaimTypes WSDL operation.
IClaimProviderWebService_ClaimTypes_OutputMessage	The response WSDL message for the ClaimTypes WSDL operation.

3.1.4.1.1.1 IClaimProviderWebService_ClaimTypes_InputMessage

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

The **SOAP action** value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimTypes
```

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

3.1.4.1.1.2 IClaimProviderWebService_ClaimTypes_OutputMessage

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

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

3.1.4.1.2 Elements

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

Element	Description
ClaimTypes	The input data for the ClaimTypes WSDL operation.
ClaimTypesResponse	The result data for the ClaimTypes WSDL operation.

3.1.4.1.2.1 ClaimTypes

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

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

providerNames: The list of claims provider names.

3.1.4.1.2.2 ClaimTypesResponse

The **ClaimTypesResponse** element specifies the result data for the **ClaimTypes** WSDL operation.

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

ClaimTypesResult: The list of claim types supported by the claims providers. The claim type should format as a URI.

3.1.4.1.3 Complex Types

None.

3.1.4.1.4 Simple Types

None.

3.1.4.1.5 Attributes

None.

3.1.4.1.6 Groups

None.

3.1.4.1.7 Attribute Groups

None.

3.1.4.2 ClaimValueTypes

This operation is used to retrieve a list of claim value types from a list of claims providers.

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

```
<wsdl:operation name="ClaimValueTypes" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimValueTypes" message="tns:IClaimProviderWebService_ClaimValueTypes_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimValueTypesResponse" message="tns:IClaimProviderWebService_ClaimValueTypes_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_ClaimValueTypes_InputMessage** (section [3.1.4.2.1.1](#)) request WSDL message and the protocol server responds with an **IClaimProviderWebService_ClaimValueTypes_OutputMessage** (section [3.1.4.2.1.2](#)) response message.

The protocol server MUST retrieve all known basic claim value types. In addition, the protocol server MUST retrieve claim value types from claims providers that meet both of the following criteria:

- The claims providers are associated with the Web application (1) specified in the input message.
- The claims providers are listed in the provider names in the input message.

The protocol server will remove the duplicated claim value types from the known basic claim value types and claims providers' claim value types.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.2.1 Messages

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

Message	Description
IClaimProviderWebService_ClaimValueTypes_InputMessage	The request WSDL message for the ClaimValueTypes WSDL operation.
IClaimProviderWebService_ClaimValueTypes_OutputMessage	The response WSDL message for the ClaimValueTypes WSDL operation.

3.1.4.2.1.1 IClaimProviderWebService_ClaimValueTypes_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimValueTypes
```

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

3.1.4.2.1.2 IClaimProviderWebService_ClaimValueTypes_OutputMessage

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

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

3.1.4.2.2 Elements

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

Element	Description
ClaimValueTypes	The input data for the ClaimValueTypes WSDL operation.
ClaimValueTypesResponse	The result data for the ClaimValueTypes WSDL operation.

3.1.4.2.2.1 ClaimValueTypes

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

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

providerNames: The list of claims provider names.

3.1.4.2.2.2 ClaimValueTypesResponse

The **ClaimValueTypesResponse** element specifies the result data for the **ClaimValueTypes** WSDL operation.

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

ClaimValueTypesResult: The list of claim value types. The claim value type should format as a URI.

3.1.4.2.3 Complex Types

None.

3.1.4.2.4 Simple Types

None.

3.1.4.2.5 Attributes

None.

3.1.4.2.6 Groups

None.

3.1.4.2.7 Attribute Groups

None.

3.1.4.3 EntityTypes

This operation is used to retrieve a list of all possible picker entity types from a list of claims providers.

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

```
<wsdl:operation name="EntityTypes" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/EntityTypes"
    message="tns:IClaimProviderWebService_EntityTypes_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/EntityTypesResponse"
    message="tns:IClaimProviderWebService_EntityTypes_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_EntityTypes_InputMessage** (section [3.1.4.3.1.1](#)) request WSDL message and the protocol server responds with an **IClaimProviderWebService_EntityTypes_OutputMessage** (section [3.1.4.3.1.2](#)) response message.

The protocol server MUST retrieve picker entity types from **claims providers** that meet both of the following criteria:

- The claims providers are associated with the Web application (1) specified in the input message.
- The claims providers are listed in the provider names in the input message.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.3.1 Messages

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

Message	Description
IClaimProviderWebService_EntityTypes_InputMessage	The request WSDL message for the EntityTypes WSDL operation.
IClaimProviderWebService_EntityTypes_OutputMessage	The response WSDL message for the EntityTypes WSDL operation.

3.1.4.3.1.1 IClaimProviderWebService_EntityTypes_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/EntityTypes
```

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

3.1.4.3.1.2 IClaimProviderWebService_EntityTypes_OutputMessage

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

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

3.1.4.3.2 Elements

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

Element	Description
EntityTypes	The input data for the EntityTypes WSDL operation.
EntityTypesResponse	The result data for the EntityTypes WSDL operation.

3.1.4.3.2.1 EntityTypes

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

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

providerNames: The list of claims provider names. If this is not specified, the protocol server MUST return results from all available claims providers.

3.1.4.3.2.2 EntityTypesResponse

The **EntityTypesResponse** element specifies the result data for the **EntityTypes** WSDL operation.

```
<xs:element name="EntityTypesResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
```

```

    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="EntityTypesResult"
type="tns:ArrayOfString"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

EntityTypesResult: The list of picker entity types supported by the claims providers.

3.1.4.3.3 Complex Types

None.

3.1.4.3.4 Simple Types

None.

3.1.4.3.5 Attributes

None.

3.1.4.3.6 Groups

None.

3.1.4.3.7 Attribute Groups

None.

3.1.4.4 GetHierarchy

This operation is used to retrieve a claims provider hierarchy tree from a claims provider.

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

```

<wsdl:operation name="GetHierarchy" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHiera
rchy" message="tns:IClaimProviderWebService_GetHierarchy_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHiera
rchyResponse" message="tns:IClaimProviderWebService_GetHierarchy_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IClaimProviderWebService_GetHierarchy_InputMessage** (section [3.1.4.4.1.1](#)) request WSDL message and the protocol server responds with an **IClaimProviderWebService_GetHierarchy_OutputMessage** (section [3.1.4.4.1.2](#)) response message.

The protocol server MUST retrieve a claims provider hierarchy tree from the claims provider that meets all the following criteria:

- The claims provider name is specified in the input message.
- The claims provider is associated with the Web application (1) specified in the input message.

- The claims provider supports hierarchy.

The protocol server MUST retrieve a hierarchy provider hierarchy tree if the name of an available hierarchy provider is specified in the input message.

3.1.4.4.1 Messages

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

Message	Description
IClaimProviderWebService_GetHierarchy_InputMessage	The request WSDL message for the GetHierarchy WSDL operation.
IClaimProviderWebService_GetHierarchy_OutputMessage	The response WSDL message for the GetHierarchy WSDL operation.

3.1.4.4.1.1 IClaimProviderWebService_GetHierarchy_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHierarchy
```

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

3.1.4.4.1.2 IClaimProviderWebService_GetHierarchy_OutputMessage

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

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

3.1.4.4.2 Elements

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

Element	Description
GetHierarchy	The input data for the GetHierarchy WSDL operation.
GetHierarchyResponse	The result data for the GetHierarchy WSDL operation.

3.1.4.4.2.1 GetHierarchy

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

```
<xs:element name="GetHierarchy" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerName" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="hierarchyNodeID" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="numberOfLevels" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
</xs:sequence>
</xs:complexType>
</xs:element>
```

providerName: The claims provider name. The value of this element MUST NOT be NULL.

principalType: The SPPincipalType value that is used to specify the type(s) of picker entities to be included in the output claims provider hierarchy tree.

hierarchyNodeID: Identifier of the node to be used as root of the returned claims provider hierarchy tree. If NULL is specified, then the protocol server MUST return the existing root of claims provider hierarchy tree.

numberOfLevels: The maximum number of levels that can be returned by the protocol server in any of the output claims provider hierarchy tree <3>. The value of this element MUST NOT be less than 1.

The protocol server MUST return all levels up to the specified number of levels from the root of the claims provider hierarchy tree. If a claims provider hierarchy tree has more than specified number of levels, then the protocol server MUST only return the specified number of levels from the root of the claims provider hierarchy tree.

The protocol server MUST return **ArgumentOutOfRangeException<"numberOfLevels">** message if the value of this element is less than 1.

3.1.4.4.2 GetHierarchyResponse

The **GetHierarchyResponse** element specifies the result data for the **GetHierarchy** WSDL operation.

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

GetHierarchyResult: A claims provider hierarchy tree.

3.1.4.4.3 Complex Types

None.

3.1.4.4.4 Simple Types

None.

3.1.4.4.5 Attributes

None.

3.1.4.4.6 Groups

None.

3.1.4.4.7 Attribute Groups

None.

3.1.4.5 GetHierarchyAll

This operation is used to retrieve a list of claims provider hierarchy trees from a list of claims providers.

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

```
<wsdl:operation name="GetHierarchyAll" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHierarchyAll" message="tns:IClaimProviderWebService_GetHierarchyAll_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHierarchyAllResponse" message="tns:IClaimProviderWebService_GetHierarchyAll_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_GetHierarchyAll_InputMessage** (section [3.1.4.5.1.1](#)) request WSDL message and the protocol server responds with an **IClaimProviderWebService_GetHierarchyAll_OutputMessage** (section [3.1.4.5.1.2](#)) response message.

The protocol server MUST retrieve claims provider hierarchy trees from claims providers that meet all the following criteria:

- The claims providers are associated with the Web application (1) specified in the input message the meet one of the following criteria:
- The claims providers are enabled and used by default in situations where the administrator has not specified the claims providers.
- The claims providers are associated with the authentication method for the Web application (1).
- The custom claims providers are configured for the Web application (1).
- The claims providers are listed in the provider names in the input message.
- The claims providers support hierarchy.

The protocol server MUST also retrieve hierarchy provider hierarchy tree if one is available. The name of the hierarchy provider is prefixed with "_HierarchyProvider_".

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.5.1 Messages

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

Message	Description
IClaimProviderWebService_GetHierarchyAll_InputMessage	The request WSDL message for the GetHierarchyAll WSDL operation.
IClaimProviderWebService_GetHierarchyAll_OutputMessage	The response WSDL message for the

Message	Description
	GetHierarchyAll WSDL operation.

3.1.4.5.1.1 IClaimProviderWebService_GetHierarchyAll_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHierarchyAll
```

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

3.1.4.5.1.2 IClaimProviderWebService_GetHierarchyAll_OutputMessage

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

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

3.1.4.5.2 Elements

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

Element	Description
GetHierarchyAll	The input data for the GetHierarchyAll WSDL operation.
GetHierarchyAllResponse	The result data for the GetHierarchyAll WSDL operation.

3.1.4.5.2.1 GetHierarchyAll

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

```
<xs:element name="GetHierarchyAll" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames" type="tns:ArrayOfString"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType"/>
      <xs:element minOccurs="1" maxOccurs="1" name="numberOfLevels" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerNames: The list of claims provider names. If this is not specified, the protocol server MUST return hierarchy tree for all available claims providers.

principalType: The SPPrincipalType value that is used to specify the type(s) of picker entities to be included in the output claims provider hierarchy tree.

numberOfLevels: The maximum number of levels that can be returned by the protocol server in any of the output claims provider hierarchy trees [<4>](#). The value of this element MUST NOT be less than 1.

The protocol server MUST return all levels up to the specified number of levels from the root of the claims provider hierarchy tree. If a claims provider hierarchy tree has more than the specified number of levels, then the protocol server MUST only return the specified number of levels from the root of the claims provider hierarchy tree.

The protocol server MUST return an **ArgumentOutOfRangeException** <"numberOfLevels"> message if the value of this element is less than 1.

3.1.4.5.2.2 GetHierarchyAllResponse

The **GetHierarchyAllResponse** element specifies the result data for the **GetHierarchyAll** WSDL operation.

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

GetHierarchyAllResult: The list of claims provider hierarchy trees.

The protocol server MUST return one claims provider hierarchy tree for each claims provider that match the criteria specified in the input message.

3.1.4.5.3 Complex Types

None.

3.1.4.5.4 Simple Types

None.

3.1.4.5.5 Attributes

None.

3.1.4.5.6 Groups

None.

3.1.4.5.7 Attribute Groups

None.

3.1.4.6 HierarchyProviderSchema

This operation is used to retrieve schema for the current hierarchy provider.

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

```
<wsdl:operation name="HierarchyProviderSchema" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Hierarch
```

```

yProviderSchema" message="tns:IClaimProviderWebService_HierarchyProviderSchema_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Hierarchy
yProviderSchemaResponse"
message="tns:IClaimProviderWebService_HierarchyProviderSchema_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IClaimProviderWebService_HierarchyProviderSchema_InputMessage** (section [3.1.4.6.1.1](#)) request WSDL message and the protocol server responds with an **IClaimProviderWebService_HierarchyProviderSchema_OutputMessage** (section [3.1.4.6.1.2](#)) response message.

3.1.4.6.1 Messages

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

Message	Description
IClaimProviderWebService_HierarchyProviderSchema_InputMessage	The request WSDL message for the HierarchyProviderSchema WSDL operation.
IClaimProviderWebService_HierarchyProviderSchema_OutputMessage	The response WSDL message for the HierarchyProviderSchema WSDL operation.

3.1.4.6.1.1 IClaimProviderWebService_HierarchyProviderSchema_InputMessage

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

The SOAP action value is:

```

http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/HierarchyProviderSche
ma

```

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

3.1.4.6.1.2 IClaimProviderWebService_HierarchyProviderSchema_OutputMessage

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

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

3.1.4.6.2 Elements

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

Element	Description
HierarchyProviderSchema	The input data for the HierarchyProviderSchema WSDL operation.
HierarchyProviderSchemaResponse	The result data for the HierarchyProviderSchema WSDL operation.

3.1.4.6.2.1 HierarchyProviderSchema

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

```
<xs:element name="HierarchyProviderSchema" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType/>
</xs:element>
```

3.1.4.6.2.2 HierarchyProviderSchemaResponse

The **HierarchyProviderSchemaResponse** element specifies the result data for the **HierarchyProviderSchema** WSDL operation.

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

HierarchyProviderSchemaResult: The schema of the current hierarchy provider.

3.1.4.6.3 Complex Types

None.

3.1.4.6.4 Simple Types

None.

3.1.4.6.5 Attributes

None.

3.1.4.6.6 Groups

None.

3.1.4.6.7 Attribute Groups

None.

3.1.4.7 ProviderSchemas

This operation is used to retrieve a list of claims provider schemas from a list of claims providers.

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

```
<wsdl:operation name="ProviderSchemas" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
```

```

    <wsdl:input
      wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ProviderSchemas" message="tns:IClaimProviderWebService ProviderSchemas InputMessage"
      xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
    <wsdl:output
      wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ProviderSchemasResponse" message="tns:IClaimProviderWebService ProviderSchemas_OutputMessage"
      xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  </wsdl:operation>

```

The protocol client sends an **IClaimProviderWebService_ProviderSchemas_InputMessage** (section [3.1.4.7.1.1](#)) request WSDL message and the protocol server responds with an **IClaimProviderWebService_ProviderSchemas_OutputMessage** (section [3.1.4.7.1.2](#)) response message.

The protocol server MUST retrieve claims provider schemas from claims providers that meet all the following criteria:

- The claims providers are enabled for the Web application (1) specified in the input message.
- The claims providers are listed in the provider names in the input message or site group provider.
- The claims providers support search and resolve.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.7.1 Messages

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

Message	Description
IClaimProviderWebService_ProviderSchemas_InputMessage	The request WSDL message for the ProviderSchemas WSDL operation.
IClaimProviderWebService_ProviderSchemas_OutputMessage	The response WSDL message for the ProviderSchemas WSDL operation.

3.1.4.7.1.1 IClaimProviderWebService_ProviderSchemas_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ProviderSchemas
```

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

3.1.4.7.1.2 IClaimProviderWebService_ProviderSchemas_OutputMessage

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

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

3.1.4.7.2 Elements

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

Element	Description
ProviderSchemas	The input data for the ProviderSchemas WSDL operation.
ProviderSchemasResponse	The result data for the ProviderSchemas WSDL operation.

3.1.4.7.2.1 ProviderSchemas

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

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

providerNames: The list of claims provider names. If this is not specified, the protocol server MUST return results from all available claims providers.

3.1.4.7.2.2 ProviderSchemasResponse

The **ProviderSchemasResponse** element specifies the result data for the **ProviderSchemas** WSDL operation.

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

ProviderSchemasResult: The list of claims provider schemas.

3.1.4.7.3 Complex Types

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

Complex type	Description
ArrayOfSPPProviderSchema	An array of SPPProviderSchema (section 2.2.4.11) elements.

3.1.4.7.3.1 ArrayOfSPPProviderSchema

Namespace: <http://schemas.microsoft.com/sharepoint/claims/>

An array of **SPPProviderSchema** (section [2.2.4.11](#)) elements.

```
<xs:complexType name="ArrayOfSPPProviderSchema" xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

```

    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="SPPProviderSchema" nillable="true"
type="tns:SPPProviderSchema"/>
    </xs:sequence>
  </xs:complexType>

```

SPPProviderSchema: The user interface display characteristics of a field in an entity.

3.1.4.7.4 Simple Types

None.

3.1.4.7.5 Attributes

None.

3.1.4.7.6 Groups

None.

3.1.4.7.7 Attribute Groups

None.

3.1.4.8 Resolve

This operation is used to resolve an input string to picker entities using a list of claims providers.

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

```

<wsdl:operation name="Resolve" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Resolve"
message="tns:IClaimProviderWebService Resolve InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveR
esponse" message="tns:IClaimProviderWebService Resolve OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IClaimProviderWebService_Resolve_InputMessage** (section [3.1.4.8.1.1](#)) request WSDL message and the protocol server responds with an **IClaimProviderWebService_Resolve_OutputMessage** (section [3.1.4.8.1.2](#)) response message.

The protocol server MUST resolve across all claims providers that meet all the following criteria:

- The claims providers are associated with the Web application (1) specified in the input message.
- The claims providers are listed in the provider names in the input message.
- The claims providers support resolve.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.8.1 Messages

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

Message	Description
IClaimProviderWebService_Resolve_InputMessage	The request WSDL message for the Resolve WSDL operation.
IClaimProviderWebService_Resolve_OutputMessage	The response WSDL message for the Resolve WSDL operation.

3.1.4.8.1.1 IClaimProviderWebService_Resolve_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Resolve
```

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

3.1.4.8.1.2 IClaimProviderWebService_Resolve_OutputMessage

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

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

3.1.4.8.2 Elements

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

Element	Description
Resolve	The input data for the Resolve WSDL operation.
ResolveResponse	The result data for the Resolve WSDL operation.

3.1.4.8.2.1 Resolve

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

```
<xs:element name="Resolve" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames" type="tns:ArrayOfString"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="resolveInput" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerNames: The list of claims provider names. If this is not specified, the protocol server **MUST** return results from all available claims providers.

principalType: The SPPrincipalType value that is used to specify the type(s) of picker entities to be included in the output claims provider hierarchy tree.

resolveInput: The input to be resolved. This MUST NOT be NULL. If this is NULL, the protocol server MUST return an **ArgumentNullException**<"value"> message.

3.1.4.8.2.2 ResolveResponse

The **ResolveResponse** element specifies the result data for the **Resolve** WSDL operation.

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

ResolveResult: The list of picker entities.

3.1.4.8.3 Complex Types

None.

3.1.4.8.4 Simple Types

None.

3.1.4.8.5 Attributes

None.

3.1.4.8.6 Groups

None.

3.1.4.8.7 Attribute Groups

None.

3.1.4.9 ResolveClaim

This operation is used to resolve an input claim (2) to picker entities using a list of claims providers.

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

```
<wsdl:operation name="ResolveClaim" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveC
laim" message="tns:IClaimProviderWebService_ResolveClaim_InputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveC
laimResponse" message="tns:IClaimProviderWebService_ResolveClaim_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_ResolveClaim_InputMessage** (section [3.1.4.9.1.1](#)) request WSDL message and the protocol server responds with an **IClaimProviderWebService_ResolveClaim_OutputMessage** (section [3.1.4.9.1.2](#)) response message.

The protocol server MUST resolve across all claims providers that meet all the following criteria:

- The claims providers are associated with the Web application (1) specified in the input message.
- The claims providers are listed in the provider names in the input message.
- The claims providers support resolve.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.9.1 Messages

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

Message	Description
IClaimProviderWebService_ResolveClaim_InputMessage	The request WSDL message for the ResolveClaim WSDL operation.
IClaimProviderWebService_ResolveClaim_OutputMessage	The response WSDL message for the ResolveClaim WSDL operation.

3.1.4.9.1.1 IClaimProviderWebService_ResolveClaim_InputMessage

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

The SOAP action value is:

`http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveClaim`

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

3.1.4.9.1.2 IClaimProviderWebService_ResolveClaim_OutputMessage

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

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

3.1.4.9.2 Elements

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

Element	Description
ResolveClaim	The input data for the ResolveClaim WSDL operation.
ResolveClaimResponse	The result data for the ResolveClaim WSDL operation.

3.1.4.9.2.1 ResolveClaim

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

```
<xs:element name="ResolveClaim" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames" type="tns:ArrayOfString"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="resolveInput" type="tns:SPClaim"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerNames: The list of claims provider names. If this is not specified, the protocol server MUST return results from all available claims providers.

principalType: The SPPrincipalType value that is used to specify the type(s) of picker entities to be included in the output claims provider hierarchy tree.

resolveInput: The input to be resolved, represented as an **SPClaim** (section [2.2.4.10](#)). The value of this element MUST NOT be NULL.

3.1.4.9.2.2 ResolveClaimResponse

The **ResolveClaimResponse** element specifies the result data for the **ResolveClaim** WSDL operation.

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

ResolveClaimResult: The list of picker entities.

3.1.4.9.3 Complex Types

None.

3.1.4.9.4 Simple Types

None.

3.1.4.9.5 Attributes

None.

3.1.4.9.6 Groups

None.

3.1.4.9.7 Attribute Groups

None.

3.1.4.10 ResolveMultiple

This operation is used to resolve a list of strings to picker entities using a list of claims providers.

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

```
<wsdl:operation name="ResolveMultiple" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveMultiple" message="tns:IClaimProviderWebService ResolveMultiple InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveMultipleResponse" message="tns:IClaimProviderWebService ResolveMultiple_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_ResolveMultiple_InputMessage** (section [3.1.4.10.1.1](#)) request WSDL message and the protocol server responds with an **IClaimProviderWebService_ResolveMultiple_OutputMessage** (section [3.1.4.10.1.2](#)) response message.

The protocol server MUST resolve across all claims providers that meet all the following criteria:

- The claims providers are associated with the Web application (1) specified in the input message.
- The claims providers are listed in the provider names in the input message.
- The claims providers support resolve.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.10.1 Messages

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

Message	Description
IClaimProviderWebService_ResolveMultiple_InputMessage	The request WSDL message for the ResolveMultiple WSDL operation.
IClaimProviderWebService_ResolveMultiple_OutputMessage	The response WSDL message for the ResolveMultiple WSDL operation.

3.1.4.10.1.1 IClaimProviderWebService_ResolveMultiple_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveMultiple
```

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

3.1.4.10.1.2 IClaimProviderWebService_ResolveMultiple_OutputMessage

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

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

3.1.4.10.2 Elements

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

Element	Description
ResolveMultiple	The input data for the ResolveMultiple WSDL operation.
ResolveMultipleResponse	The result data for the ResolveMultiple WSDL operation.

3.1.4.10.2.1 ResolveMultiple

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

```
<xs:element name="ResolveMultiple" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames" type="tns:ArrayOfString"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="resolveInput" type="tns:ArrayOfString"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerNames: The list of claims provider names. If this is not specified, the protocol server **MUST** return results from all available claims providers.

principalType: The SPPrincipalType value that is used to specify the type(s) of picker entities to be included in the output claims provider hierarchy tree.

resolveInput: The list of input strings to be resolved. This **MUST NOT** be NULL. If this is NULL, the protocol server **MUST** return an **ArgumentNullException** <"resolveInput"> message.

The protocol server **MUST** resolve each string across all claims providers.

3.1.4.10.2.2 ResolveMultipleResponse

The **ResolveMultipleResponse** element specifies the result data for the **ResolveMultiple** WSDL operation.

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

ResolveMultipleResult: The list of picker entities.

The list **MUST** contain one and only one picker entity per string in the input.

If there are multiple matches for an input string, then the protocol server MUST return a picker entity whose `IsResolved` is set to false and contains the multiple matched picker entities as the child entities.

3.1.4.10.3 Complex Types

None.

3.1.4.10.4 Simple Types

None.

3.1.4.10.5 Attributes

None.

3.1.4.10.6 Groups

None.

3.1.4.10.7 Attribute Groups

None.

3.1.4.11 ResolveMultipleClaim

This operation is used to resolve a list of claims to picker entities using a list of claims providers.

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

```
<wsdl:operation name="ResolveMultipleClaim" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveMultipleClaim" message="tns:IClaimProviderWebService ResolveMultipleClaim InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveMultipleClaimResponse"
    message="tns:IClaimProviderWebService ResolveMultipleClaim OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_ResolveMultipleClaim_InputMessage** (section [3.1.4.11.1.1](#)) request WSDL message and the protocol server responds with an **IClaimProviderWebService_ResolveMultipleClaim_OutputMessage** (section [3.1.4.11.1.2](#)) response message.

The protocol server MUST resolve across all claims providers that meet all the following criteria:

- The claims providers are associated with the Web application (1) specified in the input message.
- The claims providers are listed in the provider names in the input message.
- The claims providers support resolve.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.11.1 Messages

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

Message	Description
IClaimProviderWebService_ResolveMultipleClaim_InputMessage	The request WSDL message for the ResolveMultipleClaim WSDL operation.
IClaimProviderWebService_ResolveMultipleClaim_OutputMessage	The response WSDL message for the ResolveMultipleClaim WSDL operation.

3.1.4.11.1.1 IClaimProviderWebService_ResolveMultipleClaim_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveMultipleClaim
```

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

3.1.4.11.1.2 IClaimProviderWebService_ResolveMultipleClaim_OutputMessage

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

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

3.1.4.11.2 Elements

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

Element	Description
ResolveMultipleClaim	The input data for the ResolveMultipleClaim WSDL operation.
ResolveMultipleClaimResponse	The result data for the ResolveMultipleClaim WSDL operation.

3.1.4.11.2.1 ResolveMultipleClaim

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

```
<xs:element name="ResolveMultipleClaim" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames" type="tns:ArrayOfString"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="resolveInput" type="tns:ArrayOfSPClaim"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerNames: The list of claims provider names. If this is not specified, the protocol server MUST return results from all available claims providers.

principalType: The SPPrincipalType value that is used to specify the type(s) of picker entities to be included in the output claims provider hierarchy tree.

resolveInput: The list of claims (2) to be resolved. This MUST NOT be NULL. If this is NULL, the protocol server MUST return an **ArgumentNullException** <"resolveInput"> message.

3.1.4.11.2 ResolveMultipleClaimResponse

The **ResolveMultipleClaimResponse** element specifies the result data for the **ResolveMultipleClaim** WSDL operation.

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

ResolveMultipleClaimResult: The list of picker entities. The list MUST contain one and only one picker entity per one claim (2) in the input. If there are multiple matches for an input claim, the protocol server MUST return a picker entity whose **IsResolved** is set to **false** and contains the multiple matched picker entity as the child entities.

3.1.4.11.3 Complex Types

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

Complex type	Description
ArrayOfSPClaim	An array of elements of type SPClaim (section 2.2.4.9).

3.1.4.11.3.1 ArrayOfSPClaim

Namespace: <http://schemas.microsoft.com/sharepoint/claims/>

An array of elements of type **SPClaim** (section [2.2.4.9](#)).

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

SPClaim: The claim (2) associated with an entity.

3.1.4.11.4 Simple Types

None.

3.1.4.11.5 Attributes

None.

3.1.4.11.6 Groups

None.

3.1.4.11.7 Attribute Groups

None.

3.1.4.12 Search

This operation is used to perform a search for entities on a list of claims providers.

The input to this operation is a search string and the result of this operation is a list of claims provider hierarchy trees that contain the matched entities. If a hierarchy provider is available, the protocol server MUST also retrieve the hierarchy provider hierarchy tree and MAY move results returned by claims providers to hierarchy provider hierarchy tree.

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

```
<wsdl:operation name="Search" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Search"
    message="tns:IClaimProviderWebService_Search_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/SearchResponse"
    message="tns:IClaimProviderWebService_Search_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IClaimProviderWebService_Search_InputMessage** (section [3.1.4.12.1.1](#)) request WSDL message and the protocol server responds with an **IClaimProviderWebService_Search_OutputMessage** (section [3.1.4.12.1.2](#)) response message.

The protocol server MUST search across all claims providers that meet all the following criteria:

- The claims providers are associated with the Web application (1) specified in the input message.
- The claims providers are listed in the provider search arguments in the input message.
- The claims providers support search.

If no provider names are specified, the protocol server MUST use all the available claims providers. [<5>](#)

3.1.4.12.1 Messages

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

Message	Description
IClaimProviderWebService_Search_InputMessage	The request WSDL message for the Search WSDL operation.

Message	Description
IClaimProviderWebService_Search_OutputMessage	The response WSDL message for the Search WSDL operation.

3.1.4.12.1.1 IClaimProviderWebService_Search_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Search
```

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

3.1.4.12.1.2 IClaimProviderWebService_Search_OutputMessage

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

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

3.1.4.12.2 Elements

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

Element	Description
Search	The input data for the Search WSDL operation.
SearchResponse	The result data for the Search WSDL operation.

3.1.4.12.2.1 Search

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

```
<xs:element name="Search" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerSearchArguments"
type="tns:ArrayOfSPPProviderSearchArguments"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPPrincipalType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="searchPattern" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

providerSearchArguments: The list of **SPPProviderSearchArguments** (section [3.1.4.12.3.2](#)) that specify what claims providers to search, where to begin the search in each claims provider, and how many entities can be included in the output.

principalType: The SPPPrincipalType value that is used to specify the type(s) of picker entities to be included in the output claims provider hierarchy tree.

searchPattern: The search string. The protocol server MUST search for the string in each claims provider. The way in which the protocol server performs the search is implementation-specific. If this is NULL, the protocol server MUST return an **ArgumentNullException** <"searchPattern"> message.

3.1.4.12.2.2 SearchResponse

The **SearchResponse** element specifies the result data for the **Search** WSDL operation.

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

SearchResult: The list of claims provider hierarchy trees.

The protocol server MUST return one claims provider hierarchy tree for each claims provider that contains entities that match the search string.

3.1.4.12.3 Complex Types

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

Complex type	Description
ArrayOfSPPProviderSearchArguments	An array of SPPProviderSearchArguments (section 3.1.4.12.3.2).
SPPProviderSearchArguments	The information to control search in a claims provider.

3.1.4.12.3.1 ArrayOfSPPProviderSearchArguments

Namespace: http://schemas.microsoft.com/sharepoint/claims/

An array of **SPPProviderSearchArguments** (section [3.1.4.12.3.2](#)).

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

SPPProviderSearchArguments: The information to control search in a claims provider.

3.1.4.12.3.2 SPPProviderSearchArguments

Namespace: http://schemas.microsoft.com/sharepoint/claims/

The information to control search in a claims provider.

```
<xs:complexType name="SPPProviderSearchArguments" xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

```

<xs:sequence>
  <xs:element minOccurs="0" maxOccurs="1" name="ProviderName" type="xs:string"/>
  <xs:element minOccurs="0" maxOccurs="1" name="HierarchyNodeID" type="xs:string"/>
  <xs:element minOccurs="1" maxOccurs="1" name="MaxCount" type="xs:int"/>
</xs:sequence>
</xs:complexType>

```

ProviderName: The name of a claims provider.

HierarchyNodeID: The node identifier of a node in claims provider hierarchy tree where the protocol server begins the search. If NULL is specified, then the protocol server MUST begin the search at the root of the claims provider hierarchy tree.

MaxCount: The maximum number of matched entities to be returned by the protocol server in the output claims provider hierarchy tree for the specified claims provider. This value MUST be greater than or equal to 1.

3.1.4.12.4 Simple Types

None.

3.1.4.12.5 Attributes

None.

3.1.4.12.6 Groups

None.

3.1.4.12.7 Attribute Groups

None.

3.1.4.13 SearchAll

This operation is used to perform a search for entities on a list of claims providers. The result of this operation is a list of claims provider hierarchy trees. If a hierarchy provider is available, the protocol server MAY move entities returned by claims providers to hierarchy provider hierarchy tree.

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

```

<wsdl:operation name="SearchAll" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/SearchAll"
    message="tns:IClaimProviderWebService_SearchAll_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/SearchAllResponse"
    message="tns:IClaimProviderWebService_SearchAll_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IClaimProviderWebService_SearchAll_InputMessage** (section [3.1.4.13.1.1](#)) request WSDL message and the protocol server responds with an **IClaimProviderWebService_SearchAll_OutputMessage** (section [3.1.4.13.1.2](#)) response message.

The protocol server MUST search across all claims providers that meet all the following criteria:

- The claims providers are associated with the Web application (1) specified in the input message.
- The claims providers are listed in the provider names in the input message.
- The claims providers support search.

If no provider names are specified, the protocol server MUST use all the available claims providers.

3.1.4.13.1 Messages

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

Message	Description
IClaimProviderWebService_SearchAll_InputMessage	The request WSDL message for the SearchAll WSDL operation.
IClaimProviderWebService_SearchAll_OutputMessage	The response WSDL message for the SearchAll WSDL operation.

3.1.4.13.1.1 IClaimProviderWebService_SearchAll_InputMessage

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

The SOAP action value is:

```
http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/SearchAll
```

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

3.1.4.13.1.2 IClaimProviderWebService_SearchAll_OutputMessage

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

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

3.1.4.13.2 Elements

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

Element	Description
SearchAll	The input data for the SearchAll WSDL operation.
SearchAllResponse	The result data for the SearchAll WSDL operation.

3.1.4.13.2.1 SearchAll

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

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



```

    <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPPrincipalType"/>
    <xs:element minOccurs="0" maxOccurs="1" name="searchPattern" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="maxCount" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
</xs:element>

```

providerNames: The list of claims provider names. If this is NULL, the protocol server MUST return results from all available claims providers.

principalType: The SPPPrincipalType value that is used to specify the type(s) of picker entities to be included in the output claims provider hierarchy tree.

searchPattern: The search string. This MUST NOT be NULL. If list of provider names is NULL, the protocol server MUST search for each string in all claims providers. The way in which the protocol server performs the search is implementation-specific. If this is NULL, the protocol server MUST return an **ArgumentNullException** <"searchPattern"> message.

maxCount: The maximum number of matched entities to be returned in total across all the output claims provider hierarchy trees. This number MUST be greater than or equal to 1.

3.1.4.13.2 SearchAllResponse

The **SearchAllResponse** element specifies the result data for the **SearchAll** WSDL operation.

```

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

```

SearchAllResult: The list of claims provider hierarchy trees.

The protocol server MUST return one claims provider hierarchy tree for each claims provider that contains entities that match the search string.

3.1.4.13.3 Complex Types

None.

3.1.4.13.4 Simple Types

None.

3.1.4.13.5 Attributes

None.

3.1.4.13.6 Groups

None.

3.1.4.13.7 Attribute Groups

None.

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

Preliminary

4 Protocol Examples

4.1 GetHierarchyAll

The protocol server has four claims providers: System, AllUsers, Group and People. The System **claims provider** has claim types - Application Pool Identity and Farm identifier. The protocol server has a Web application (1) (<http://localhost>) that is configured to use claim-based authentication mode.

4.1.1 Retrieve Claim Provider Hierarchy Trees from All Available Claim Providers

In this example, claims provider hierarchy trees are fetched from all available claims providers. The protocol server has four claims providers: System, AllUsers, Group and People.

The protocol client constructs the following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetHierarchyAll xmlns="http://tempuri.org/"
      xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
      <providerNames>
        <string>System</string>
        <string>AllUsers</string>
        <string>Group</string>
        <string>People</string>
      </providerNames>
      <principalType>User</principalType>
      <numberOfLevels>3</numberOfLevels>
    </GetHierarchyAll>
  </s:Body>
</s:Envelope>
```

The protocol server responds with following message:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <GetHierarchyAllResponse xmlns="http://tempuri.org/">
      <GetHierarchyAllResult>
        <SPPProviderHierarchyTree>
          <Nm>System</Nm>
          <ProviderName>System</ProviderName>
          <HierarchyNodeID/>
          <IsLeaf>false</IsLeaf>
          <Children>
            <SPPProviderHierarchyNode>
              <Nm>Application Pool Identity</Nm>
              <ProviderName>System</ProviderName>
              <HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/processidentitylogonname</HierarchyNodeID>
              <IsLeaf>true</IsLeaf>
              <Children/>
              <EntityData/>
              <Count>-1</Count>
            </SPPProviderHierarchyNode>
          </SPPProviderHierarchyNode>
          <Nm>Farm ID</Nm>
        </GetHierarchyAllResult>
      </GetHierarchyAllResponse>
    </s:Body>
  </s:Envelope>
```

```

        <ProviderName>System</ProviderName>
<HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/farmid</HierarchyNode
ID>
        <IsLeaf>true</IsLeaf>
        <Children/>
        <EntityData/>
        <Count>-1</Count>
    </SPPProviderHierarchyNode>
</Children>
<EntityData/>
<Count>-1</Count>
<IsRoot>true</IsRoot>
</SPPProviderHierarchyTree>
<SPPProviderHierarchyTree>
    <Nm>AllUsers</Nm>
    <ProviderName>AllUsers</ProviderName>
    <HierarchyNodeID/>
    <IsLeaf>true</IsLeaf>
    <Children/>
    <EntityData/>
    <Count>-1</Count>
    <IsRoot>true</IsRoot>
</SPPProviderHierarchyTree>
<SPPProviderHierarchyTree>
    <Nm>Group</Nm>
    <ProviderName>Group</ProviderName>
    <HierarchyNodeID/>
    <IsLeaf>true</IsLeaf>
    <Children/>
    <EntityData/>
    <Count>-1</Count>
    <IsRoot>true</IsRoot>
</SPPProviderHierarchyTree>
<SPPProviderHierarchyTree>
    <Nm>People and Organizations</Nm>
    <ProviderName>People</ProviderName>
    <HierarchyNodeID/>
    <IsLeaf>true</IsLeaf>
    <Children/>
    <EntityData/>
    <Count>-1</Count>
    <IsRoot>true</IsRoot>
</SPPProviderHierarchyTree>
</GetHierarchyAllResult>
</GetHierarchyAllResponse>
</s:Body>
</s:Envelope>

```

4.1.2 Retrieve Claim Provider Hierarchy Trees from One of the Claim Providers

In this example, claims provider hierarchy trees are fetched from one of the claims providers ("System").

The protocol client constructs the following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetHierarchyAll xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-
instance">
      <context>http://localhost</context>
      <providerNames>
        <string>System</string>
      </providerNames>
      <numberOfLevels>3</numberOfLevels>
    </GetHierarchyAll>
  </s:Body>
</s:Envelope>

```

```

    <includeEntityData>true</includeEntityData>
  </GetHierarchyAll>
</s:Body>
</s:Envelope>

```

The protocol server responds with following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <GetHierarchyAllResponse xmlns="http://tempuri.org/">
      <GetHierarchyAllResult>
        <SPPProviderHierarchyTree>
          <Nm>System</Nm>
          <ProviderName>System</ProviderName>
          <HierarchyNodeID/>
          <IsLeaf>false</IsLeaf>
          <Children>
            <SPPProviderHierarchyNode>
              <Nm>Application Pool Identity</Nm>
              <ProviderName>System</ProviderName>
              <HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/processidentitylogonname</HierarchyNodeID>
              <IsLeaf>true</IsLeaf>
              <Children/>
              <EntityData/>
              <Count>-1</Count>
            </SPPProviderHierarchyNode>
          </SPPProviderHierarchyNode>
          <Nm>Farm ID</Nm>
          <ProviderName>System</ProviderName>
          <HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/farmid</HierarchyNodeID>
          <IsLeaf>true</IsLeaf>
          <Children/>
          <EntityData/>
          <Count>-1</Count>
        </SPPProviderHierarchyNode>
      </Children>
      <EntityData/>
      <Count>-1</Count>
      <IsRoot>true</IsRoot>
    </SPPProviderHierarchyTree>
  </GetHierarchyAllResult>
</GetHierarchyAllResponse>
</s:Body>
</s:Envelope>

```

4.2 SearchAll

In this example, protocol client calls the SearchAll method to search the entities on two claims providers ("System" and "All Users") with a search pattern "domain\user1". The **URI** of the Web application (1) (<http://localhost>) and the maximum number of matched entities to be returned (100) are also specified.

The protocol server has four claims providers: System, AllUsers, Group and People. The System claims provider has claim types - Application Pool Identity and Farm identifier. The protocol server has a Web application (1) (<http://localhost>) that is configured to use claim-based authentication mode.

The protocol client constructs the following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <SearchAll xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-
instance">
      <context>http://localhost</context>
      <providerNames>
        <string>System</string>
        <string>Groups</string>
      </providerNames>
      <searchPattern>Domain\User1</searchPattern>
      <maxCount>100</maxCount>
    </SearchAll>
  </s:Body>
</s:Envelope>

```

The protocol server responds with following message:



```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <SearchAllResponse xmlns="http://tempuri.org/">
      <SearchAllResult>
        <SPPProviderHierarchyTree>
          <Nm>System</Nm>
          <ProviderName>System</ProviderName>
          <HierarchyNodeID/>
          <IsLeaf>false</IsLeaf>
          <Children>
            <SPPProviderHierarchyNode>
              <Nm>Application Pool Identity</Nm>
              <ProviderName>System</ProviderName>
              <HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/processidentitylogonname</HierarchyNodeID>
              <IsLeaf>true</IsLeaf>
              <Children/>
              <EntityData>
                <PickerEntity>
                  <Key>c:0'0c|System|Domain\User1</Key>
                  <DisplayText>Domain\User1
(/sharepoint/2009/08/claims/processidentitylogonname)</DisplayText>
                  <IsResolved>true</IsResolved>
                  <Description/>
                  <EntityType>System</EntityType>
                  <EntityDataElements/>
                  <MultipleMatches/>
                </PickerEntity>
              </EntityData>
              <Count>-1</Count>
            </SPPProviderHierarchyNode>
            <SPPProviderHierarchyNode>
              <Nm>Farm ID</Nm>
              <ProviderName>System</ProviderName>
              <HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/farmid</HierarchyNodeID>
              <IsLeaf>true</IsLeaf>
              <Children/>
              <EntityData/>
              <Count>-1</Count>
            </SPPProviderHierarchyNode>
          </Children>
          <EntityData/>
          <Count>-1</Count>
          <IsRoot>true</IsRoot>
        </SPPProviderHierarchyTree>
      </SearchAllResult>
    </SearchAllResponse>
  </s:Body>
</s:Envelope>

```

```

    </SearchAllResponse>
  </s:Body>
</s:Envelope>

```

4.3 Search

In this example, protocol client calls the Search method to search the entities on one of the claims providers ("System") with a search pattern "domain\user1". The URI of the Web application (1) (<http://localhost>) and the maximum number of matched entities to be returned (100) are also specified.

The protocol server has four claims providers: System, AllUsers, Group and People. The System claims provider has claim types - Application Pool Identity and Farm identifier. The protocol server has a Web application (1) (<http://localhost>) that is configured to use claim-based authentication mode.

The protocol client constructs the following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <Search xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-
instance">
      <context>http://localhost</context>
      <providerSearchArguments>
        <SPPProviderSearchArguments>
          <ProviderName>System</ProviderName>
          <MaxCount>100</MaxCount>
        </SPPProviderSearchArguments>
      </providerSearchArguments>
      <searchPattern> domain\user1</searchPattern>
    </Search>
  </s:Body>
</s:Envelope>

```

The protocol server responds with the following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <SearchResponse xmlns="http://tempuri.org/">
      <SearchResult>
        <SPPProviderHierarchyTree>
          <Nm>System</Nm>
          <ProviderName>System</ProviderName>
          <HierarchyNodeID/>
          <IsLeaf>false</IsLeaf>
          <Children>
            <SPPProviderHierarchyNode>
              <Nm>Application Pool Identity</Nm>
              <ProviderName>System</ProviderName>
              <HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/processidentitylogonname/</HierarchyNodeID>
              <IsLeaf>true</IsLeaf>
              <Children/>
              <EntityData>
                <PickerEntity>
                  <Key>c:0'0c|System|Domain\User1</Key>
                  <DisplayText>Domain\User1
(/sharepoint/2009/08/claims/processidentitylogonname)</DisplayText>
                  <IsResolved>true</IsResolved>
                  <Description/>
                  <EntityType>System</EntityType>
                </PickerEntity>
              </EntityData>
            </SPPProviderHierarchyNode>
          </Children>
        </SPPProviderHierarchyTree>
      </SearchResult>
    </SearchResponse>
  </s:Body>
</s:Envelope>

```

```

        <EntityDataElements/>
        <MultipleMatches/>
      </PickerEntity>
    </EntityData>
    <Count>-1</Count>
  </SPPProviderHierarchyNode>
  <SPPProviderHierarchyNode>
    <Nm>Farm ID</Nm>
    <ProviderName>System</ProviderName>
  <HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/farmid</HierarchyNode
  ID>
    <IsLeaf>true</IsLeaf>
    <Children/>
    <EntityData/>
    <Count>-1</Count>
  </SPPProviderHierarchyNode>
</Children>
<EntityData/>
<Count>-1</Count>
<IsRoot>true</IsRoot>
</SPPProviderHierarchyTree>
</SearchResult>
</SearchResponse>
</s:Body>
</s:Envelope>

```

4.4 GetHierarchy

In this example, protocol client retrieves a claims provider hierarchy tree from a claims provider by specifying node identifier of the claims provider hierarchy tree.

The protocol server has four claims providers: System, AllUsers, Group and People. The System claims provider has claim types - Application Pool Identity and Farm identifier. The protocol server has a Web application (1) (<http://localhost>) that is configured to use claim-based authentication mode.

The protocol client constructs the following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetHierarchy xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-
    instance">
      <context>http://localhost</context>
      <providerName>System</providerName>

      <hierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/processidentitylogonn
      ame</hierarchyNodeID>
      <numberOfLevels>4</numberOfLevels>
      <includeEntityData>true</includeEntityData>
    </GetHierarchy>
  </s:Body>
</s:Envelope>

```

The protocol server responds with following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <GetHierarchyResponse xmlns="http://tempuri.org/">
      <GetHierarchyResult>
        <Nm>Application Pool Identity</Nm>

```



```

        <ProviderName>System</ProviderName>
<HierarchyNodeID>http://schemas.microsoft.com/sharepoint/2009/08/claims/processidentitylogonname</HierarchyNodeID>
    <IsLeaf>true</IsLeaf>
    <Children/>
    <EntityData/>
    <Count>-1</Count>
    <IsRoot>>false</IsRoot>
</GetHierarchyResult>
</GetHierarchyResponse>
</s:Body>
</s:Envelope>

```

4.5 EntityTypes

In this example, protocol client fetches the picker entity types from one of the claims providers ("System").

The protocol server has four claims providers: System, AllUsers, Group and People. The protocol server has a Web application (1) (<http://localhost>) that is configured to use claim-based authentication mode.

The protocol client constructs the following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <EntityTypes xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
      <context>http://localhost</context>
      <providerNames>
        <string>System</string>
      </providerNames>
    </EntityTypes>
  </s:Body>
</s:Envelope>

```

The protocol server responds with following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <EntityTypesResponse xmlns="http://tempuri.org/">
      <EntityTypesResult>
        <string>System</string>
      </EntityTypesResult>
    </EntityTypesResponse>
  </s:Body>
</s:Envelope>

```

4.6 ClaimTypes

In this example, protocol client fetches the claim types from one of the claims providers ("System").

The protocol server has four claims providers: System, AllUsers, Group and People. The protocol server has a Web application (1) (<http://localhost>) that is configured to use claim-based authentication mode.

The protocol client constructs the following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <ClaimTypes xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-
instance">
      <context>http://localhost</context>
      <providerNames>
        <string>System</string>
      </providerNames>
    </ClaimTypes>
  </s:Body>
</s:Envelope>

```

The protocol server responds with following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <ClaimTypesResponse xmlns="http://tempuri.org/">
      <ClaimTypesResult>
        <string>http://schemas.microsoft.com/sharepoint/2009/08/claims/farmid</string>
        <string>http://schemas.microsoft.com/sharepoint/2009/08/claims/processidentitylogonname</stri
ng>
        <string>http://schemas.microsoft.com/sharepoint/2009/08/claims/processidentitysid</string>

        <string>http://schemas.microsoft.com/sharepoint/2009/08/claims/windowstoken/handle</string>
        <string>http://sharepoint.microsoft.com/claims/2009/01/windowstoken/processid</string>
      </ClaimTypesResult>
    </ClaimTypesResponse>
  </s:Body>
</s:Envelope>

```

4.7 ClaimValueTypes

In this example, protocol client fetches the claim value types from one of the claims providers ("System").

The protocol server has four claims providers: System, AllUsers, Group and People. The protocol server has a Web application (1) (<http://localhost>) that is configured to use claims authentication mode.

The protocol client constructs the following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <ClaimValueTypes xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-
instance">
      <context>http://localhost</context>
      <providerNames>
        <string>System</string>
      </providerNames>
    </ClaimValueTypes>
  </s:Body>
</s:Envelope>

```

The protocol server responds with following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">

```

```

<ClaimValueTypesResponse xmlns="http://tempuri.org/">
  <ClaimValueTypesResult>
    <string>http://www.w3.org/2001/XMLSchema#Guid</string>
    <string>http://www.w3.org/2001/XMLSchema#string</string>
  </ClaimValueTypesResult>
</ClaimValueTypesResponse>
</s:Body>
</s:Envelope>

```

4.8 Resolve

In this example, protocol client calls **Resolve** method to resolve an input string ("domain\user1") to picker entities using the available claims providers.

The protocol server has four claims providers: System, AllUsers, Group and People. The System claims provider has claim types: Application Pool Identity and Farm identifier. The protocol server has a Web application (1) (http://localhost) that is configured to use claim-based authentication mode.

The protocol client constructs the following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <Resolve xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
      <providerNames>
        <string>System</string>
        <string>AllUsers</string>
        <string>Group</string>
        <string>People</string>
      </providerNames>
      <principalType>User</principalType>
      <resolveInput>domain\user1</resolveInput>
    </Resolve>
  </s:Body>
</s:Envelope>

```

The protocol server responds with following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <ResolveResponse xmlns="http://tempuri.org/">
      <ResolveResult>
        <PickerEntity>
          <Key>c:0'0c|System| domain\user1</Key>
          <DisplayText>user1</DisplayText>
          <IsResolved>true</IsResolved>
          <Description/>
          <EntityType>System</EntityType>
          <EntityDataElements/>
          <MultipleMatches/>
        </PickerEntity>
        <PickerEntity>
          <Key>i:0#0w| domain\user1</Key>
          <DisplayText>User1
(/sharepoint/2009/08/claims/processidentitylogonname)</DisplayText>
          <IsResolved>true</IsResolved>
          <Description> domain\user1</Description>
          <EntityType>User</EntityType>
          <EntityGroupName>User</EntityGroupName>
          <EntityDataElements>

```

```

    <Pair>
      <First xsi:type="xsd:string">Email</First>
      <Second xsi:type="xsd:string"/>
    </Pair>
    <Pair>
      <First xsi:type="xsd:string">SIPAddress</First>
    </Pair>
    <Pair>
      <First xsi:type="xsd:string">Title</First>
      <Second xsi:type="xsd:string"/>
    </Pair>
    <Pair>
      <First xsi:type="xsd:string">WorkPhone</First>
      <Second xsi:type="xsd:string"/>
    </Pair>
  </EntityDataElements>
  <MultipleMatches/>
</PickerEntity>
</ResolveResult>
</ResolveResponse>
</s:Body>
</s:Envelope>

```

4.9 ResolveMultiple

In this example, protocol client calls **ResolveMultiple** method to resolve two input strings ("domain\user1" and "domain\user2") to picker entities using the available claims providers.

The protocol server has four claims providers: System, AllUsers, Group and People. The System claims provider has the following claim types: Application Pool Identity, and Farm identifier. The protocol server has a Web application (1) (<http://localhost>) that is configured to use claim-based authentication mode.

The protocol client constructs the following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <ResolveMultiple xmlns="http://tempuri.org/" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
      <providerNames>
        <string>System</string>
        <string>AllUsers</string>
        <string>Group</string>
        <string>People</string>
      </providerNames>
      <principalType>User</principalType>
      <resolveInput>
        <string>domain\user1</string>
        <string>domain\user2</string>
      </resolveInput>
    </ResolveMultiple>
  </s:Body>
</s:Envelope>

```

The protocol server responds with following message:

```

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <ResolveMultipleResponse xmlns="http://tempuri.org/">
      <ResolveMultipleResult>

```

```

<PickerEntity>
  <Key/>
  <DisplayText>domain\user1</DisplayText>
  <IsResolved>>false</IsResolved>
  <Description/>
  <EntityType/>
  <EntityDataElements/>
  <MultipleMatches>
    <anyType xsi:type="PickerEntity">
      <Key>c:0'0c\System\domain\user1</Key>
      <DisplayText>User1
(/sharepoint/2009/08/claims/processidentitylogonname)</DisplayText>
      <IsResolved>>true</IsResolved>
      <Description/>
      <EntityType>System</EntityType>
      <EntityDataElements/>
      <MultipleMatches/>
    </anyType>
    <anyType xsi:type="PickerEntity">
      <Key>i:0#0w\domain\user1</Key>
      <DisplayText>User1</DisplayText>
      <IsResolved>>true</IsResolved>
      <Description>domain\user1</Description>
      <EntityType>User</EntityType>
      <EntityDataElements>
        <Pair>
          <First xsi:type="xsd:string">Email</First>
          <Second xsi:type="xsd:string"/>
        </Pair>
        <Pair>
          <First xsi:type="xsd:string">SIPAddress</First>
        </Pair>
        <Pair>
          <First xsi:type="xsd:string">Title</First>
          <Second xsi:type="xsd:string"/>
        </Pair>
        <Pair>
          <First xsi:type="xsd:string">WorkPhone</First>
          <Second xsi:type="xsd:string"/>
        </Pair>
      </EntityDataElements>
      <MultipleMatches/>
    </anyType>
  </MultipleMatches>
</PickerEntity>
<PickerEntity>
  <Key/>
  <DisplayText>domain\user2</DisplayText>
  <IsResolved>>false</IsResolved>
  <Description/>
  <EntityType/>
  <EntityDataElements/>
  <MultipleMatches>
    <anyType xsi:type="PickerEntity">
      <Key>c:0'0c\System\domain\user2</Key>
      <DisplayText>User2
(/sharepoint/2009/08/claims/processidentitylogonname)</DisplayText>
      <IsResolved>>true</IsResolved>
      <Description/>
      <EntityType>System</EntityType>
      <EntityDataElements/>
      <MultipleMatches/>
    </anyType>
    <anyType xsi:type="PickerEntity">
      <Key>i:0#0w\domain\user2</Key>
      <DisplayText>User2</DisplayText>
      <IsResolved>>true</IsResolved>
      <Description>domain\user2</Description>

```

```
<EntityType>User</EntityType>
<EntityDataElements>
  <Pair>
    <First xsi:type="xsd:string">Email</First>
    <Second xsi:type="xsd:string"/>
  </Pair>
  <Pair>
    <First xsi:type="xsd:string">SIPAddress</First>
  </Pair>
  <Pair>
    <First xsi:type="xsd:string">Title</First>
    <Second xsi:type="xsd:string"/>
  </Pair>
  <Pair>
    <First xsi:type="xsd:string">WorkPhone</First>
    <Second xsi:type="xsd:string"/>
  </Pair>
</EntityDataElements>
<MultipleMatches/>
</anyType>
</MultipleMatches>
</PickerEntity>
</ResolveMultipleResult>
</ResolveMultipleResponse>
</s:Body>
</s:Envelope>
```

Preliminary

5 Security

5.1 Security Considerations for Implementers

This protocol introduces no additional security considerations beyond those applicable to its underlying protocols.

5.2 Index of Security Parameters

None.

Preliminary

6 Appendix A: Full WSDL

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

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://schemas.microsoft.com/sharepoint/claims/"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"
targetNamespace="http://schemas.microsoft.com/sharepoint/claims/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types>
    <xs:schema xmlns:tns1="http://schemas.microsoft.com/sharepoint/claims/Imports"
targetNamespace="http://schemas.microsoft.com/sharepoint/claims/Imports">
      <xs:import namespace="http://schemas.microsoft.com/sharepoint/claims/" />
    </xs:schema>
  </wsdl:types>
  <wsdl:portType name="IClaimProviderWebService">
    <wsdl:operation name="ClaimTypes">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimTypes"
message="tns:IClaimProviderWebService_ClaimTypes_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimTypesResponse"
message="tns:IClaimProviderWebService_ClaimTypes_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="ClaimValueTypes">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimValueTypes"
message="tns:IClaimProviderWebService_ClaimValueTypes_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimValueTypesResponse"
message="tns:IClaimProviderWebService_ClaimValueTypes_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="EntityTypes">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/EntityTypes"
message="tns:IClaimProviderWebService_EntityTypes_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/EntityTypesResponse"
message="tns:IClaimProviderWebService_EntityTypes_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="ProviderSchemas">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ProviderSchemas"
message="tns:IClaimProviderWebService_ProviderSchemas_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ProviderSchemasResponse"
message="tns:IClaimProviderWebService_ProviderSchemas_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="HierarchyProviderSchema">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/HierarchyProviderSchema"
message="tns:IClaimProviderWebService_HierarchyProviderSchema_InputMessage"/>
      <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/HierarchyProviderSchemaResponse"
message="tns:IClaimProviderWebService_HierarchyProviderSchema_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="GetHierarchyAll">
      <wsdl:input
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHierarchyAll"
message="tns:IClaimProviderWebService_GetHierarchyAll_InputMessage"/>
```



```

    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHierarchyAllResponse" message="tns:IClaimProviderWebService GetHierarchyAll OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="GetHierarchy">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHierarchy" message="tns:IClaimProviderWebService_GetHierarchy_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHierarchyResponse" message="tns:IClaimProviderWebService GetHierarchy OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="SearchAll">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/SearchAll" message="tns:IClaimProviderWebService SearchAll InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/SearchAllResponse" message="tns:IClaimProviderWebService_SearchAll_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="Search">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Search" message="tns:IClaimProviderWebService_Search_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/SearchResponse" message="tns:IClaimProviderWebService_Search_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="Resolve">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Resolve" message="tns:IClaimProviderWebService Resolve InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveResponse" message="tns:IClaimProviderWebService_Resolve_OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="ResolveClaim">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveClaim" message="tns:IClaimProviderWebService ResolveClaim InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveClaimResponse" message="tns:IClaimProviderWebService ResolveClaim OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="ResolveMultiple">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveMultiple" message="tns:IClaimProviderWebService_ResolveMultiple_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveMultipleResponse" message="tns:IClaimProviderWebService ResolveMultiple OutputMessage"/>
    </wsdl:operation>
    <wsdl:operation name="ResolveMultipleClaim">
    <wsdl:input
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveMultipleClaim" message="tns:IClaimProviderWebService_ResolveMultipleClaim_InputMessage"/>
    <wsdl:output
wsaw:Action="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveMultipleClaimResponse" message="tns:IClaimProviderWebService_ResolveMultipleClaim_OutputMessage"/>
    </wsdl:operation>
    </wsdl:portType>
    <wsdl:binding name="DefaultBinding IClaimProviderWebService"
type="tns:IClaimProviderWebService">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="ClaimTypes">

```

```

    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimType
s" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ClaimValueTypes">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ClaimValu
eTypes" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="EntityTypes">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/EntityTyp
es" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ProviderSchemas">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ProviderS
chemas" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="HierarchyProviderSchema">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Hierarchy
ProviderSchema" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetHierarchyAll">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHierar
chyAll" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetHierarchy">

```

```

    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/GetHierar
chy" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SearchAll">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/SearchAll
" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="Search">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Search"
style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="Resolve">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/Resolve"
style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ResolveClaim">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveCl
aim" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ResolveMultiple">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveMu
ltiple" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="ResolveMultipleClaim">

```

```

    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/claims/IClaimProviderWebService/ResolveMultipleClaim" style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:message name="IClaimProviderWebService_ClaimTypes_InputMessage">
    <wsdl:part name="parameters" element="tns:ClaimTypes"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_ClaimTypes_OutputMessage">
    <wsdl:part name="parameters" element="tns:ClaimTypesResponse"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_ClaimValueTypes_InputMessage">
    <wsdl:part name="parameters" element="tns:ClaimValueTypes"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_ClaimValueTypes_OutputMessage">
    <wsdl:part name="parameters" element="tns:ClaimValueTypesResponse"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_EntityTypes_InputMessage">
    <wsdl:part name="parameters" element="tns:EntityTypes"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_EntityTypes_OutputMessage">
    <wsdl:part name="parameters" element="tns:EntityTypesResponse"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_GetHierarchy_InputMessage">
    <wsdl:part name="parameters" element="tns:GetHierarchy"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_GetHierarchy_OutputMessage">
    <wsdl:part name="parameters" element="tns:GetHierarchyResponse"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_GetHierarchyAll_InputMessage">
    <wsdl:part name="parameters" element="tns:GetHierarchyAll"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_GetHierarchyAll_OutputMessage">
    <wsdl:part name="parameters" element="tns:GetHierarchyAllResponse"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_HierarchyProviderSchema_InputMessage">
    <wsdl:part name="parameters" element="tns:HierarchyProviderSchema"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_HierarchyProviderSchema_OutputMessage">
    <wsdl:part name="parameters" element="tns:HierarchyProviderSchemaResponse"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_ProviderSchemas_InputMessage">
    <wsdl:part name="parameters" element="tns:ProviderSchemas"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_ProviderSchemas_OutputMessage">
    <wsdl:part name="parameters" element="tns:ProviderSchemasResponse"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_Resolve_InputMessage">
    <wsdl:part name="parameters" element="tns:Resolve"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_Resolve_OutputMessage">
    <wsdl:part name="parameters" element="tns:ResolveResponse"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_ResolveClaim_InputMessage">
    <wsdl:part name="parameters" element="tns:ResolveClaim"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_ResolveClaim_OutputMessage">
    <wsdl:part name="parameters" element="tns:ResolveClaimResponse"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_ResolveMultiple_InputMessage">
    <wsdl:part name="parameters" element="tns:ResolveMultiple"/>
</wsdl:message>

```

```
<wsdl:message name="IClaimProviderWebService_ResolveMultiple_OutputMessage">
  <wsdl:part name="parameters" element="tns:ResolveMultipleResponse"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_ResolveMultipleClaim_InputMessage">
  <wsdl:part name="parameters" element="tns:ResolveMultipleClaim"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_ResolveMultipleClaim_OutputMessage">
  <wsdl:part name="parameters" element="tns:ResolveMultipleClaimResponse"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_Search_InputMessage">
  <wsdl:part name="parameters" element="tns:Search"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_Search_OutputMessage">
  <wsdl:part name="parameters" element="tns:SearchResponse"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_SearchAll_InputMessage">
  <wsdl:part name="parameters" element="tns:SearchAll"/>
</wsdl:message>
<wsdl:message name="IClaimProviderWebService_SearchAll_OutputMessage">
  <wsdl:part name="parameters" element="tns:SearchAllResponse"/>
</wsdl:message>
</wsdl:definitions>
```

Preliminary

7 Appendix B: Full XML Schema

For ease of implementation, the following is the full XML schema for this protocol.

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:tns="http://schemas.microsoft.com/sharepoint/claims/"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.microsoft.com/sharepoint/claims/"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="ClaimTypes">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="providerNames"
          type="tns:ArrayOfString"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="ArrayOfString">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="string" nillable="true"
        type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="ClaimTypesResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="ClaimTypesResult"
          type="tns:ArrayOfString"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ClaimValueTypes">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="providerNames"
          type="tns:ArrayOfString"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ClaimValueTypesResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="ClaimValueTypesResult"
          type="tns:ArrayOfString"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="EntityTypes">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="providerNames"
          type="tns:ArrayOfString"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="EntityTypesResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="EntityTypesResult"
          type="tns:ArrayOfString"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ProviderSchemas">
    <xs:complexType>
      <xs:sequence>
```

```

        <xs:element minOccurs="0" maxOccurs="1" name="providerNames"
type="tns:ArrayOfString"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ProviderSchemasResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="ProviderSchemasResult"
type="tns:ArrayOfSPProviderSchema"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:complexType name="ArrayOfSPProviderSchema">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="SPProviderSchema" nillable="true"
type="tns:SPProviderSchema"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="SPProviderSchema">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="DisplayName" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="ProviderName" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="ProviderSchema"
type="tns:ArrayOfSPSchemaElement"/>
        <xs:element minOccurs="1" maxOccurs="1" name="SupportsHierarchy" type="xs:boolean"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfSPSchemaElement">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="SPSchemaElement" nillable="true"
type="tns:SPSchemaElement"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="SPSchemaElement">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="DisplayName" type="xs:string"/>
        <xs:element minOccurs="1" maxOccurs="1" name="Type" type="tns:SPSchemaElementType"/>
    </xs:sequence>
</xs:complexType>
<xs:simpleType name="SPSchemaElementType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="None"/>
        <xs:enumeration value="TableViewOnly"/>
        <xs:enumeration value="DetailViewOnly"/>
        <xs:enumeration value="Both"/>
    </xs:restriction>
</xs:simpleType>
<xs:element name="HierarchyProviderSchema">
    <xs:complexType/>
</xs:element>
<xs:element name="HierarchyProviderSchemaResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="HierarchyProviderSchemaResult"
type="tns:SPProviderSchema"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="GetHierarchyAll">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="providerNames"
type="tns:ArrayOfString"/>
            <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType"/>
            <xs:element minOccurs="1" maxOccurs="1" name="numberOfLevels" type="xs:int"/>

```

```

    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:simpleType name="SPPrincipalType">
  <xs:list>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="None"/>
        <xs:enumeration value="User"/>
        <xs:enumeration value="DistributionList"/>
        <xs:enumeration value="SecurityGroup"/>
        <xs:enumeration value="SharePointGroup"/>
        <xs:enumeration value="All"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:list>
</xs:simpleType>
<xs:element name="GetHierarchyAllResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="GetHierarchyAllResult"
type="tns:ArrayOfSPPProviderHierarchyTree"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:complexType name="ArrayOfSPPProviderHierarchyTree">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="SPPProviderHierarchyTree"
nillable="true" type="tns:SPPProviderHierarchyTree"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="SPPProviderHierarchyTree">
  <xs:complexContent mixed="false">
    <xs:extension base="tns:SPPProviderHierarchyElement">
      <xs:sequence>
        <xs:element minOccurs="1" maxOccurs="1" name="IsRoot" type="xs:boolean"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="SPPProviderHierarchyElement">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Nm" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="ProviderName" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="HierarchyNodeID" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="IsLeaf" type="xs:boolean"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Children"
type="tns:ArrayOfSPPProviderHierarchyNode"/>
    <xs:element minOccurs="0" maxOccurs="1" name="EntityData"
type="tns:ArrayOfPickerEntity"/>
    <xs:element minOccurs="1" maxOccurs="1" name="Count" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfSPPProviderHierarchyNode">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="SPPProviderHierarchyNode"
nillable="true" type="tns:SPPProviderHierarchyNode"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="SPPProviderHierarchyNode">
  <xs:complexContent mixed="false">
    <xs:extension base="tns:SPPProviderHierarchyElement"/>
  </xs:complexContent>
</xs:complexType>
<xs:complexType name="ArrayOfPickerEntity">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="PickerEntity" nillable="true"
type="tns:PickerEntity"/>
  </xs:sequence>
</xs:complexType>

```



```

    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="PickerEntity">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="Key" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="DisplayText" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="IsResolved" type="xs:boolean"/>
      <xs:element minOccurs="0" maxOccurs="1" name="Description" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="EntityType" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="EntityGroupName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="HierarchyIdentifier"/>
      <xs:element minOccurs="0" maxOccurs="1" name="EntityDataElements"
type="tns:ArrayOfPair"/>
      <xs:element minOccurs="0" maxOccurs="1" name="MultipleMatches"
type="tns:ArrayOfAnyType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="ProviderName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="ProviderDisplayName" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ArrayOfPair">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="Pair" nillable="true"
type="tns:Pair"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="Pair">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="First"/>
      <xs:element minOccurs="0" maxOccurs="1" name="Second"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ArrayOfAnyType">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="anyType" nillable="true"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="GetHierarchy">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="providerName" type="xs:string"/>
        <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType"/>
        <xs:element minOccurs="0" maxOccurs="1" name="hierarchyNodeID" type="xs:string"/>
        <xs:element minOccurs="1" maxOccurs="1" name="numberOfLevels" type="xs:int"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetHierarchyResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="GetHierarchyResult"
type="tns:SPProviderHierarchyTree"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="SearchAll">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="providerNames"
type="tns:ArrayOfString"/>
        <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType"/>
        <xs:element minOccurs="0" maxOccurs="1" name="searchPattern" type="xs:string"/>
        <xs:element minOccurs="1" maxOccurs="1" name="maxCount" type="xs:int"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="SearchAllResponse">

```

```

    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="SearchAllResult"
type="tns:ArrayOfSPPProviderHierarchyTree"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Search">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="providerSearchArguments"
type="tns:ArrayOfSPPProviderSearchArguments"/>
        <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPPrincipalType"/>
        <xs:element minOccurs="0" maxOccurs="1" name="searchPattern" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="ArrayOfSPPProviderSearchArguments">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="SPPProviderSearchArguments"
nillable="true" type="tns:SPPProviderSearchArguments"/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="SPPProviderSearchArguments">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="ProviderName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="HierarchyNodeID" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="MaxCount" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="SearchResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="SearchResult"
type="tns:ArrayOfSPPProviderHierarchyTree"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Resolve">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="providerNames"
type="tns:ArrayOfString"/>
        <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPPrincipalType"/>
        <xs:element minOccurs="0" maxOccurs="1" name="resolveInput" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ResolveResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="ResolveResult"
type="tns:ArrayOfPickerEntity"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="ResolveClaim">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="providerNames"
type="tns:ArrayOfString"/>
        <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPPrincipalType"/>
        <xs:element minOccurs="0" maxOccurs="1" name="resolveInput" type="tns:SPClaim"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```

```

</xs:element>
<xs:complexType name="SPClaim">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="ClaimType" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Value" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="ValueType" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="OriginalIssuer" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="ResolveClaimResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="ResolveClaimResult"
type="tns:ArrayOfPickerEntity"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ResolveMultiple">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames"
type="tns:ArrayOfString"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="resolveInput"
type="tns:ArrayOfString"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ResolveMultipleResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="ResolveMultipleResult"
type="tns:ArrayOfPickerEntity"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ResolveMultipleClaim">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="providerNames"
type="tns:ArrayOfString"/>
      <xs:element minOccurs="1" maxOccurs="1" name="principalType"
type="tns:SPPrincipalType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="resolveInput"
type="tns:ArrayOfSPClaim"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:complexType name="ArrayOfSPClaim">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="SPClaim" nillable="true"
type="tns:SPClaim"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="ResolveMultipleClaimResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="ResolveMultipleClaimResult"
type="tns:ArrayOfPickerEntity"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs: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.

- Microsoft Lync 2010
- Microsoft FAST Search Server 2010
- Microsoft Office 2010 suites
- Microsoft SharePoint Designer 2010
- Microsoft SharePoint Foundation 2010
- Microsoft SharePoint Workspace 2010
- Microsoft Visio 2010
- Microsoft Lync 2013
- Microsoft Office 2013
- Microsoft SharePoint Designer 2013
- Microsoft Visio 2013
- Microsoft SharePoint Foundation 2013
- com16
- Microsoft Office 2016 Preview
- Microsoft Visio 2016 Preview

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

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

[<1> Section 2.2.4.10](#): SharePoint Foundation 2010 and SharePoint Foundation 2013 require this value to be NULL.

[<2> Section 2.2.4.11](#): SharePoint Foundation 2010 and SharePoint Foundation 2013 do not use this element, and set it to -1.

[<3> Section 3.1.4.4.2.1](#): SharePoint Foundation 2010 and SharePoint Foundation 2013 do not use this element.

[<4> Section 3.1.4.5.2.1](#): SharePoint Foundation 2010 and SharePoint Foundation 2013 do not use this element.

[<5> Section 3.1.4.12](#): For this operation, the server will not use any available claims providers.

9 Change Tracking

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

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

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

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

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

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

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

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

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

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

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

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

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

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
8 Appendix C: Product Behavior	Updated list of supported products.	Y	Content updated due to protocol revision.

Preliminary

10 Index

A

Abstract data model
[server](#) 24
[Applicability](#) 13
[ArrayOfAnyType complex type](#) 15
[ArrayOfPair complex type](#) 15
[ArrayOfPickerEntity complex type](#) 16
[ArrayOfSPPProviderHierarchyNode complex type](#) 16
[ArrayOfSPPProviderHierarchyTree complex type](#) 16
[ArrayOfSPSchemaElement complex type](#) 17
[ArrayOfString complex type](#) 17
[Attribute groups](#) 23
[Attributes](#) 22

C

[Capability negotiation](#) 13
[Change tracking](#) 85
[ClaimTypes example](#) 65
[ClaimValueTypes example](#) 66

Client

[overview](#) 24
[Complex types](#) 15
[ArrayOfAnyType](#) 15
[ArrayOfPair](#) 15
[ArrayOfPickerEntity](#) 16
[ArrayOfSPPProviderHierarchyNode](#) 16
[ArrayOfSPPProviderHierarchyTree](#) 16
[ArrayOfSPSchemaElement](#) 17
[ArrayOfString](#) 17
[Pair](#) 17
[PickerEntity](#) 17
[SPClaim](#) 18
[SPPProviderHierarchyElement](#) 19
[SPPProviderHierarchyNode](#) 20
[SPPProviderHierarchyTree](#) 20
[SPPProviderSchema](#) 20
[SPSchemaElement](#) 21

D

Data model - abstract
[server](#) 24

E

[EntityTypes example](#) 65
Events
[local - server](#) 58
[timer - server](#) 58
Examples
[ClaimTypes](#) 65
[ClaimValueTypes](#) 66
[EntityTypes](#) 65
[GetHierarchy](#) 64
[GetHierarchyAll](#) 59
[Resolve](#) 67
[ResolveMultiple](#) 68

[retrieve claim provider hierarchy trees from all available claim providers](#) 59
[retrieve claim provider hierarchy trees from one of the claim providers](#) 60
[Search](#) 63
[SearchAll](#) 61

F

[Fields - vendor-extensible](#) 13
[Full WSDL](#) 72
[Full XML Schema](#) 78

G

[GetHierarchy example](#) 64
[GetHierarchyAll example](#) 59
[Glossary](#) 9
[Groups](#) 22

I

[Implementer - security considerations](#) 71
[Index of security parameters](#) 71
[Informative references](#) 11
Initialization
[server](#) 25
[Introduction](#) 9

L

Local events
[server](#) 58

M

Message processing
[server](#) 25
Messages
[ArrayOfAnyType complex type](#) 15
[ArrayOfPair complex type](#) 15
[ArrayOfPickerEntity complex type](#) 16
[ArrayOfSPPProviderHierarchyNode complex type](#) 16
[ArrayOfSPPProviderHierarchyTree complex type](#) 16
[ArrayOfSPSchemaElement complex type](#) 17
[ArrayOfString complex type](#) 17
[attribute groups](#) 23
[attributes](#) 22
[complex types](#) 15
[elements](#) 14
[enumerated](#) 14
[groups](#) 22
[namespaces](#) 14
[Pair complex type](#) 17
[PickerEntity complex type](#) 17
[simple types](#) 21
[SPClaim complex type](#) 18
[SPPPrincipalType simple type](#) 21
[SPPProviderHierarchyElement complex type](#) 19
[SPPProviderHierarchyNode complex type](#) 20

- [SPPProviderHierarchyTree complex type](#) 20
- [SPPProviderSchema complex type](#) 20
- [SPSchemaElement complex type](#) 21
- [SPSchemaElementType simple type](#) 22
- [syntax](#) 14
- [transport](#) 14

N

- [Namespaces](#) 14
- [Normative references](#) 11

O

Operations

- [ClaimTypes](#) 26
- [ClaimValueTypes](#) 28
- [EntityTypes](#) 30
- [GetHierarchy](#) 32
- [GetHierarchyAll](#) 35
- [HierarchyProviderSchema](#) 37
- [ProviderSchemas](#) 39
- [Resolve](#) 42
- [ResolveClaim](#) 44
- [ResolveMultiple](#) 47
- [ResolveMultipleClaim](#) 49
- [Search](#) 52
- [SearchAll](#) 55
- [Overview \(synopsis\)](#) 12

P

- [Pair complex type](#) 17
- [Parameters - security index](#) 71
- [PickerEntity complex type](#) 17
- [Preconditions](#) 12
- [Prerequisites](#) 12
- [Product behavior](#) 84
- Protocol Details
 - [overview](#) 24

R

References

- [informative](#) 11
- [normative](#) 11
- [Relationship to other protocols](#) 12
- [Resolve example](#) 67
- [ResolveMultiple example](#) 68
- [Retrieve claim provider hierarchy trees from all available claim providers example](#) 59
- [Retrieve claim provider hierarchy trees from one of the claim providers example](#) 60

S

- [Search example](#) 63
- [SearchAll example](#) 61
- Security
 - [implementer considerations](#) 71
 - [parameter index](#) 71
- Sequencing rules
 - [server](#) 25

Server

- [abstract data model](#) 24
- [ClaimTypes operation](#) 26
- [ClaimValueTypes operation](#) 28
- [details](#) 24
- [EntityTypes operation](#) 30
- [GetHierarchy operation](#) 32
- [GetHierarchyAll operation](#) 35
- [HierarchyProviderSchema operation](#) 37
- [initialization](#) 25
- [local events](#) 58
- [message processing](#) 25
- [overview](#) 24
- [ProviderSchemas operation](#) 39
- [Resolve operation](#) 42
- [ResolveClaim operation](#) 44
- [ResolveMultiple operation](#) 47
- [ResolveMultipleClaim operation](#) 49
- [Search operation](#) 52
- [SearchAll operation](#) 55
- [sequencing rules](#) 25
- [timer events](#) 58
- [timers](#) 25
- [Simple types](#) 21
 - [SPPPrincipalType](#) 21
 - [SPSchemaElementType](#) 22
- [SPClaim complex type](#) 18
- [SPPPrincipalType simple type](#) 21
- [SPPProviderHierarchyElement complex type](#) 19
- [SPPProviderHierarchyNode complex type](#) 20
- [SPPProviderHierarchyTree complex type](#) 20
- [SPPProviderSchema complex type](#) 20
- [SPSchemaElement complex type](#) 21
- [SPSchemaElementType simple type](#) 22
- [Standards assignments](#) 13
- Syntax
 - [messages - overview](#) 14

T

- Timer events
 - [server](#) 58
- Timers
 - [server](#) 25
- [Tracking changes](#) 85
- [Transport](#) 14
- Types
 - [complex](#) 15
 - [simple](#) 21

V

- [Vendor-extensible fields](#) 13
- [Versioning](#) 13

W

- [WSDL](#) 72

X

- [XML Schema](#) 78

Preliminary