

[MS-WOPI]: Web Application Open Platform Interface Protocol Specification

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.
- **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 (beta) 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 (beta) 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.

Revision Summary

Date	Revision History	Revision Class	Comments
01/20/2012	0.1	New	Released new document.

Table of Contents

1 Introduction	6
1.1 Glossary	6
1.2 References	6
1.2.1 Normative References	6
1.2.2 Informative References	7
1.3 Overview	7
1.4 Relationship to Other Protocols	8
1.5 Prerequisites/Preconditions	9
1.6 Applicability Statement	9
1.7 Versioning and Capability Negotiation	9
1.8 Vendor-Extensible Fields	9
1.9 Standards Assignments	9
2 Messages	10
2.1 Transport	10
2.2 Message Syntax	10
2.2.1 Custom HTTP Headers	10
2.2.2 Common URI Parameters	10
3 Protocol Details	12
3.1 WOPI Discovery Details	12
3.1.1 Message Processing Events and Sequencing Rules	12
3.1.1.1 HTTP://server/hosting/discovery	12
3.1.1.1.1 Discovery	12
3.1.1.1.1.1 Request Body	12
3.1.1.1.1.2 Response Body	12
3.1.1.1.1.2.1 Elements	12
3.1.1.1.1.2.1.1 wopi-discovery	12
3.1.1.1.1.2.2 Complex Types	13
3.1.1.1.1.2.2.1 ct_wopi-discovery	13
3.1.1.1.1.2.2.2 ct_net-zone	13
3.1.1.1.1.2.2.3 ct_app-name	13
3.1.1.1.1.2.2.4 ct_wopi-action	13
3.1.1.1.1.2.3 Simple Types	14
3.1.1.1.1.2.3.1 st_wopi-action-values	14
3.1.1.1.1.2.3.2 st_wopi-action-requirements	15
3.1.1.1.1.2.3.3 st_wopi-url-source	16
3.1.1.1.1.2.3.4 st_wopi-zone	17
3.1.1.1.1.3 Processing Details	17
3.2 WOPI Protocol Client Details	18
3.3 WOPI Protocol Server Details	18
3.3.1 Abstract Data Model	18
3.3.2 Timers	18
3.3.3 Initialization	18
3.3.4 Higher-Layer Triggered Events	19
3.3.5 Message Processing Events and Sequencing Rules	19
3.3.5.1 HTTP://server/<...>/wopi*/files/<id>	19
3.3.5.1.1 CheckFileInfo	19
3.3.5.1.1.1 Request Body	20
3.3.5.1.1.2 Response Body	20

3.3.5.1.1.3	Processing Details.....	23
3.3.5.1.2	PutRelativeFile	23
3.3.5.1.2.1	Request Body.....	24
3.3.5.1.2.2	Response Body.....	24
3.3.5.1.2.3	Processing Details.....	24
3.3.5.1.3	Lock	25
3.3.5.1.3.1	Request Body.....	25
3.3.5.1.3.2	Response Body.....	25
3.3.5.1.3.3	Processing Details.....	25
3.3.5.1.4	Unlock	26
3.3.5.1.4.1	Request Body.....	26
3.3.5.1.4.2	Response Body.....	26
3.3.5.1.4.3	Processing Details.....	26
3.3.5.1.5	RefreshLock.....	26
3.3.5.1.5.1	Request Body.....	27
3.3.5.1.5.2	Response Body.....	27
3.3.5.1.5.3	Processing Details.....	27
3.3.5.1.6	UnlockAndRelock.....	27
3.3.5.1.6.1	Request Body.....	28
3.3.5.1.6.2	Response Body.....	28
3.3.5.1.6.3	Processing Details.....	28
3.3.5.1.7	ExecuteCellStorageRequest	28
3.3.5.1.7.1	Request Body.....	29
3.3.5.1.7.2	Response Body.....	29
3.3.5.1.7.3	Processing Details.....	29
3.3.5.1.8	ExecuteCellStorageRelativeRequest	29
3.3.5.1.8.1	Request Body.....	30
3.3.5.1.8.2	Response Body.....	30
3.3.5.1.8.3	Processing Details.....	30
3.3.5.1.9	DeleteFile.....	30
3.3.5.1.9.1	Request Body.....	30
3.3.5.1.9.2	Response Body.....	31
3.3.5.1.9.3	Processing Details.....	31
3.3.5.2	HTTP://server/<...>/wopi*/folders/<id>	31
3.3.5.2.1	CheckFolderInfo	31
3.3.5.2.1.1	Request Body.....	31
3.3.5.2.1.2	Response Body.....	31
3.3.5.2.1.3	Processing Details.....	33
3.3.5.3	HTTP://server/<...>/wopi*/files/<id>/contents.....	33
3.3.5.3.1	GetFile.....	33
3.3.5.3.1.1	Request Body.....	34
3.3.5.3.1.2	Response Body.....	34
3.3.5.3.1.3	Processing Details.....	34
3.3.5.3.2	PutFile	34
3.3.5.3.2.1	Request Body.....	35
3.3.5.3.2.2	Response Body.....	35
3.3.5.3.2.3	Processing Details.....	35
3.3.5.4	HTTP://server/<...>/wopi*/folder/<id>/children	35
3.3.5.4.1	EnumerateChildren.....	35
3.3.5.4.1.1	Request Body.....	36
3.3.5.4.1.2	Response Body.....	36
3.3.5.4.1.3	Processing Details.....	36
3.3.6	Timer Events	36

3.3.7 Other Local Events	36
4 Protocol Examples	37
4.1 Accessing Discovery XML	37
4.2 Viewing a Document	37
5 Security	38
5.1 Security Considerations for Implementers	38
5.2 Index of Security Parameters	38
6 Appendix A: Full XML Schema	39
7 Appendix B: Full JSON Schema	41
7.1 CheckFileInfo JSON	41
7.2 CheckFolderInfo JSON	42
7.3 EnumerateChildren JSON	42
8 Appendix C: Product Behavior	43
9 Change Tracking	44
10 Index	45

1 Introduction

The Web Application Open Platform Interface Protocol (WOPI) defines a set of operations that enables a client to access and change files stored by a server. This allows the client to render files and provide file editing functionality for files stored by the server.

Sections 1.8, 2, and 3 of this specification are normative and contain RFC 2119 language. Sections 1.5 and 1.9 are also normative but cannot contain RFC 2119 language. All other sections and examples in this specification are informative.

1.1 Glossary

The following terms are defined in [\[MS-OFCGLOS\]](#):

file extension
Information Rights Management (IRM)
Uniform Resource Identifier (URI)

The following terms are specific to this document:

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

1.2 References

References to Microsoft Open Specification documents do not include a publishing year because links are to the latest version of the documents, which are updated frequently. References to other documents include a publishing year when one is available.

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. Please check the archive site, <http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624>, as an additional source.

[FIPS180-2] Federal Information Processing Standards Publication, "Secure Hash Standard", FIPS PUB 180-2, August 2002, <http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf>

[MS-FSSHHTTP] Microsoft Corporation, "[File Synchronization via SOAP over HTTP Protocol Specification](#)".

[MS-OMWWH] Microsoft Corporation, "[Office Mobile Word Web Handler Protocol Specification](#)".

[RFC1766] Alvestrand, H., "Tags for the Identification of Languages", RFC 1766, March 1995, <http://www.ietf.org/rfc/rfc1766.txt>

[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.ietf.org/rfc/rfc2616.txt>

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

[RFC3023] Murata, M., St.Laurent, S., and Kohn, D., "XML Media Types", RFC 3023, January 2001, <http://www.ietf.org/rfc/rfc3023.txt>

[RFC4627] Crockford, D., "The application/json Media Type for Javascript Object Notation (JSON)", RFC 4627, July 2006, <http://www.ietf.org/rfc/rfc4627.txt>

[RFC5323] Reschke, J., Ed., Reddy, S., Davis, J., and Babich, A., "Web Distributed Authoring and Versioning (WebDAV) SEARCH", RFC 5323, November 2008, <http://www.rfc-editor.org/rfc/rfc5323.txt>

[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.S., Ed., Beech, D., Ed., Maloney, M., Ed., and Mendelsohn, N., Ed., "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

[MS-OFCGLOS] Microsoft Corporation, "[Microsoft Office Master Glossary](#)".

1.3 Overview

WOPI defines a set of operations that enables a client to access and change files stored by a server. This allows the client to render files and provide file editing functionality for files stored by the server.

One example of how a client might use WOPI is by providing a browser-based viewer for a specific type of file. That client uses WOPI to get the contents of the file in order to present that content to the user as a web page in a browser. Here is how that might work...

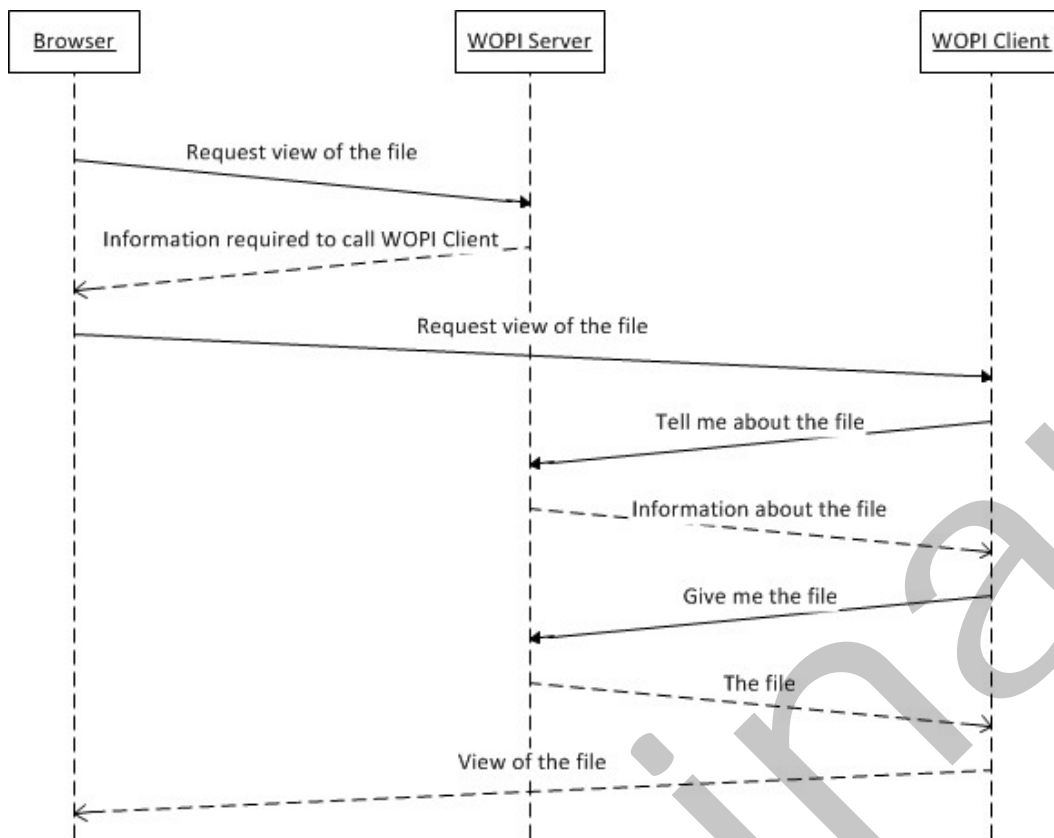


Figure 1: Using WOPI to provide a browser-based viewer for a specific type of file

A notable detail in the interaction in the preceding figure is that the WOPI server provides "information required to call the WOPI Client". This information is learned by the WOPI server through an interaction call WOPI discovery. WOPI clients provide a mechanism through which a WOPI server discovers the abilities of the WOPI client and methods for invoking those abilities. The following diagram depicts that interaction:

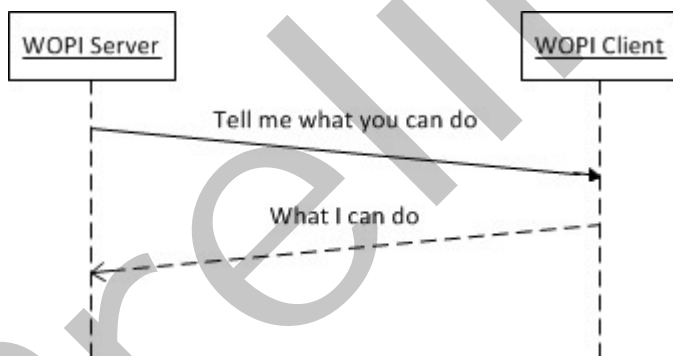


Figure 2: WOPI discovery

1.4 Relationship to Other Protocols

The WOPI Protocol uses HTTP as described in [\[RFC2616\]](#), and HTTPS as described in [\[RFC2818\]](#).

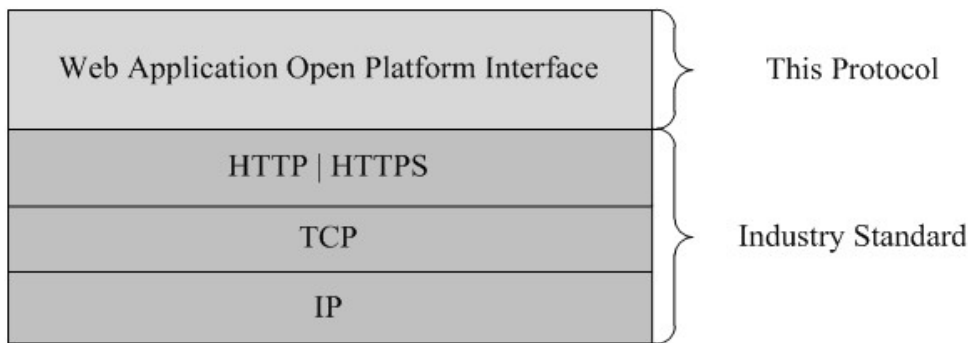


Figure 3: This protocol in relation to other protocols

1.5 Prerequisites/Preconditions

Some parts of the WOPI protocol require that the WOPI server and WOPI client have implemented the system described in "File Synchronization via SOAP over HTTP Protocol Specification" ([\[MS-FSSHHTTP\]](#)). This implementation is optional because WOPI servers and WOPI clients are not required to implement the complete WOPI protocol.

1.6 Applicability Statement

This document specifies a protocol for enabling the bidirectional transfer of file information using HTTP/HTTPS between systems designed to store and manage files and systems designed to render and manipulate files.

1.7 Versioning and Capability Negotiation

The WOPI client indicates its capabilities to the WOPI server via WOPI Discovery (section [3.1](#)).

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

This protocol uses standard IANA port assignments for HTTP and SSL. These standard port assignments use IANA-assigned ports, as listed in the following table.

Parameter	Value	Reference
IANA assigned port for HTTP	80	http://www.iana.org/assignments/part-number
IANA assigned port for SSL	443	http://www.iana.org/assignments/part-number

2 Messages

2.1 Transport

Messages MUST be transported using HTTP or HTTPS using the default ports for these protocols.

2.2 Message Syntax

This section contains common definitions used by this protocol.

2.2.1 Custom HTTP Headers

The following HTTP headers MAY be included with all WOPI requests.

Header	Description
X-WOPI-ClientVersion	A string specifying the version of the WOPI client. There is no standard for how this string should be formatted. This string MUST NOT be used for anything other than logging.
X-WOPI-MachineName	A string indicating the name of the machine making the call which MUST NOT be used for anything other than logging.
X-WOPI-PerfTraceRequested	A Boolean value that indicates that the WOPI client wants the WOPI server to return a value for X-WOPI-PerfTrace .
X-WOPI-CorrelationID	A string that the WOPI server MAY use when logging server activity to correlate that activity with WOPI client activity.

The following HTTP header MAY be included with all WOPI responses.

Header	Description
X-WOPI-ServerVersion	A string specifying the version of the WOPI server. There is no standard for how this string should be formatted. This string MUST NOT be used for anything other than logging.
X-WOPI-MachineName	A string specifying the name of the WOPI server which MUST NOT be used for anything other than logging.
X-WOPI-PerfTrace	A string that the WOPI client MAY use to track performance data.
X-WOPI-ServerError	A string indicating that an error occurred while processing the WOPI request. This string MAY include details about the error, and MUST NOT be used for anything other than logging.

2.2.2 Common URI Parameters

The following URI parameters MUST be included with all WOPI requests.

URI Parameter	Description
<token>	A string used by the WOPI server to determine the identity and permissions of the issuer of

URI Parameter	Description
	<p>the WOPI request. The token is generated by the WOPI server and learned by the WOPI client before the first WOPI request is issued. The WOPI client requires no understanding of the format or content of the token.</p> <p>The token SHOULD expire after a length of time deemed reasonable by the implementer of the WOPI server.</p> <p>The token SHOULD only provide access to resources associated with the WOPI request. That is, the token SHOULD be scoped to a specific user and set of resources.</p>
<id>	<p>The id represents the file or folder being operated on by the WOPI client. A WOPI server MUST issue a unique id for any resource that a WOPI client operates on.</p>

The <token> and <id> parameters are a core part of all WOPI requests. The URI syntax for using these parameters is described first in the **CheckFileInfo** operation (see section [3.3.5.1.1](#)). The WOPI server provides both <token> and <id> through the **urlsrc** parameter as described in the **st_wopi-url-source** simple type (see section [3.1.1.1.1.2.3.3](#)).

3 Protocol Details

3.1 WOPI Discovery Details

The WOPI client provides XML [\[RFC3023\]](#) that describes the supported abilities of the WOPI client and how to invoke these abilities through URIs. This XML is available through a predetermined URI that MUST be provided by a WOPI client.

3.1.1 Message Processing Events and Sequencing Rules

WOPI Discovery involves a single URI that takes no parameters.

Resource	Description
HTTP://server/hosting/discovery	Provides access to information about the capabilities of a WOPI client and the mechanisms for invoking those abilities through URIs.

3.1.1.1 HTTP://server/hosting/discovery

The data that describes the supported abilities of the WOPI client and how to invoke these abilities through URIs is provided through the following URI:

HTTP://server/hosting/discovery

Operation	Description
Discovery	Returns data that describes the supported abilities of the WOPI client and how to invoke these abilities through URIs

3.1.1.1.1 Discovery

The operation that returns data that describes the supported abilities of the WOPI client and how to invoke these abilities through URIs.

3.1.1.1.1.1 Request Body

None.

3.1.1.1.1.2 Response Body

The WOPI Discovery response is XML [\[RFC3023\]](#) with the following characteristics.

3.1.1.1.1.2.1 Elements

This section describes the elements of the Discovery XML.

3.1.1.1.1.2.1.1 wopi-discovery

A **ct_wopi-discovery** element (section [3.1.1.1.2.2.1](#)) specifies the capabilities and URIs for accessing the capabilities of the WOPI client. This element is the root element of WOPI Discovery XML.

The following W3C XML schema ([\[XMLSCHEMA1\]](#) section 2.1) fragment specifies the contents of this element.

```
<xs:element name="wopi-discovery" type="ct_wopi-discovery"/>
```

3.1.1.1.1.2.2 Complex Types

The following XML schema complex type definitions are specific to this operation.

3.1.1.1.1.2.2.1 ct_wopi-discovery

This **ct_wopi-discovery** complex type specifies the capabilities and URIs for accessing the capabilities of the WOPI client.

```
<xs:complexType name="ct_wopi-discovery">  
  <xs:sequence>  
    <xs:element name="net-zone" minOccurs="1" maxOccurs="4" type="ct_net-zone"/>  
  </xs:sequence>  
</xs:complexType>
```

3.1.1.1.1.2.2.2 ct_net-zone

This **ct_net-zone** complex type specifies the protocol and the intended network-type of the **ct_app-name** type elements (section [3.1.1.1.1.2.2.3](#)) it contains.

```
<xs:complexType name="ct_net-zone">  
  <xs:sequence>  
    <xs:element name="app-name" minOccurs="1" maxOccurs="unbounded"  
      type="ct_appname"/>  
  </xs:sequence>  
  <xs:attribute name="name" type="st_wopi-zone"/>  
</xs:complexType>
```

3.1.1.1.1.2.2.3 ct_app-name

The **ct_app-name** complex type specifies the application name that a set of WOPI actions are grouped under. The application name is intended as a convenient way of grouping similar actions such as all those pertaining to spreadsheets. It is not required that all actions contained within an element of this type have the same **file extension** or progid (a string that identifies a folder as being associated with a specific application).

```
<xs:complexType name="ct_app-name">  
  <xs:sequence>  
    <xs:element name="action" minOccurs="1" maxOccurs="unbounded"  
      type="ct_wopiaction"/>  
  </xs:sequence>  
</xs:complexType>
```

3.1.1.1.1.2.2.4 ct_wopi-action

The **ct_wopi-action** complex type specifies an action (such as viewing a file) that a WOPI client can perform.

Attributes:

name: A **st_wopi-action-values** attribute (section [3.1.1.1.1.2.3.1](#)) that specifies the name of the action.

default: A **Boolean** attribute that specifies whether the WOPI Server should use this action as the default action for this file type.

requires: A **st_wopi-action-requirements** attribute (section [3.1.1.1.1.2.3.2](#)) that specifies the required capabilities of the WOPI server in order to be able to use this action.

urlsrc: A **st_wopi-url-source** attribute (section [3.1.1.1.1.2.3.3](#)) that specifies the **URI** that the WOPI Server can use to invoke the action.

ext: A **string** attribute that specifies the file extension supported by this action. Note that either **ext** or **progid** MUST be present but that **ext** and **progid** MUST NOT be present in the same action.

progid: A **string** attribute that specifies the **progid** (a string that identifies a folder as being associated with a specific application) supported by this action. Note that either **ext** or **progid** MUST be present but that **ext** and **progid** MUST NOT be present in the same action.

newprogid: A **string** attribute that specifies the **progid** that the **editnew** (see **st_wopi-action-values** in section [3.1.1.1.1.2.3.1](#)) will use for the new file. Note that either **newext** or **newprogid** MUST be present when using **editnew** but that **newext** and **newprogid** MUST NOT be present in the same action.

newext: A **string** attribute that specifies the file extension that the **editnew** (see **st_wopi-action-values** in section [3.1.1.1.1.2.3.1](#)) will use for the new file. Note that either **newext** or **newprogid** MUST be present when using **editnew** but that **newext** and **newprogid** MUST NOT be present in the same action.

The following W3C XML schema ([\[XMLSCHEMA1\]](#) section 2.1) fragment specifies the contents of this complex type.

```
<xs:complexType name="ct_wopi-action">
  <xs:attribute name="name" use="required" type="st_wopi-action-values"/>
  <xs:attribute name="default" type="xs:boolean" default="false"/>
  <xs:attribute name="requires" type="st_wopi-action-requirements"/>
  <xs:attribute name="urlsrc" use="required" type="st_wopi-url-source"/>
  <xs:attribute name="ext" type="xs:string" use="optional"/>
  <xs:attribute name="progid" type="xs:string" use="optional"/>
  <xs:attribute name="newprogid" type="xs:string" use="optional"/>
  <xs:attribute name="newext" type="xs:string" use="optional"/>
</xs:complexType>
```

3.1.1.1.1.2.3 Simple Types

The following XML schema simple type definitions are specific to this operation.

3.1.1.1.1.2.3.1 st_wopi-action-values

This simple type specifies all the well-known WOPI actions that a WOPI client and WOPI server MAY support. It is important to note that while the following schema restricts this type, WOPI servers MUST not fail when they encounter additional values not included in this schema.

Enumeration Values:

view: An action that indicates that the WOPI client supports viewing the file type. All WOPI Clients MUST support this action for all file types that they support

edit: An action that indicates that the WOPI client supports editing the file type.

mobileview: An action that indicates that the WOPI client supports a view of the file type that is optimized for viewing on mobile devices such as cellular phones.

embedview: An action that indicates that the WOPI client supports a view of the file that can be embedded in a web page.

mobileclient: An action that indicates that the WOPI client supports viewing the file in clients that conform to [\[MS-OMWWH\]](#).

present: An action that indicates that the WOPI client supports a broadcast of the file. A broadcast is a special view that is driven by one or more presenters from a client for the purposes of being viewed by a set of attendees viewing in separate clients. The **present** action is associated with the URI used by presenters.

attend: An action that indicates that the WOPI client supports a broadcast of the file. A broadcast is a special view that is driven by one or more presenters from a client for the purposes of being viewed by a set of attendees viewing in separate clients. The **attend** action is associated with the URI used by attendees.

editnew: An action that indicates that the WOPI client supports the creation of a new file based on the file provided by the WOPI server. The **newext** and **newprogid** attributes defined by the `ct_wopi-action` complex type (see section [3.1.1.1.2.2.4](#)) indicate the type of file that will result from invoking the **editnew** action.

The following W3C XML schema ([\[XMLSCHEMA1\]](#) section 2.1) fragment specifies the contents of this simple type.

```
<xs:simpleType name="st_wopi-action-values">
  <xs:restriction base="xs:string">
    <xs:enumeration value="view" />
    <xs:enumeration value="edit" />
    <xs:enumeration value="mobileview" />
    <xs:enumeration value="embedview" />
    <xs:enumeration value="mobileclient" />
    <xs:enumeration value="present" />
    <xs:enumeration value="attend" />
    <xs:enumeration value="editnew" />
  </xs:restriction>
</xs:simpleType>
```

3.1.1.1.2.3.2 st_wopi-action-requirements

The **st_wopi-action-requirements** simple type specifies the capabilities that a WOPI server must support in order to invoke the associated WOPI action. It is important to note that while the following schema restricts this type to well-known requirements, WOPI servers MUST not fail when they encounter additional values not included in this schema.

Values:

update: Indicates that the WOPI server must support changes to the file through **PutFile** (see section [3.3.5.3.2](#))

locks: Indicates that the WOPI server must support the ability to lock the file for editing through **Lock** (see section [3.3.5.1.3](#)), **Unlock** (see section [3.3.5.1.4](#)), **RefreshLock** (see section [3.3.5.1.5](#)), and **UnlockAndRelock** (see section [3.3.5.1.6](#)).

cobalt: Indicates that the WOPI server must support the system describe in [\[MS-FSSHTTP\]](#) through **ExecuteCellStorageRequest** (see section [3.3.5.1.7](#)) and **ExecuteCellStorageRelativeRequest** (see section [3.3.5.1.8](#)).

containers: Indicates that the WOPI server supports accessing folders and folder hierarchy through **EnumerateChildren** (see section [3.3.5.4.1](#)), **GetFolder** (see section [3.3.5.2.1](#)) and **DeleteFile** (see section [3.3.5.1.9](#)).

The following W3C XML schema ([\[XMLSCHEMA1\]](#) section 2.1) fragment specifies the contents of this simple type.

```
<xs:simpleType name="st_wopi-action-requirements">
  <xs:restriction base="xs:string">
    <xs:pattern value="^((update|locks|cobalt|containers)\s*(,|,|s*|$))*
      ((update|locks|cobalt|containers)\s*$)"/>
  </xs:restriction>
</xs:simpleType>
```

3.1.1.1.2.3.3 st_wopi-url-source

The **st_wopi-url-source** simple type specifies the URI that is used to invoke the associated WOPI action.

When a WOPI server uses elements of this type, they MUST append the **WOPISrc** parameter and the **access_token** parameter to the URL.

WOPISrc: A URI that conforms to the pattern **http://server/<...>/wopi*/files/<id>** (see section [3.3.5.1](#)).

There are a number of optional query string parameters that this simple type may contain. As described in the following, the WOPI servers MUST replace each optional parameter when using URI of this type, either by removing them, or processing them by providing the appropriate string in place of the placeholder value. Each optional parameter conforms to the pattern

<name=PLACEHOLDER_VALUE[&]>. The WOPI server MAY always omit the optional parameter, but MUST only process it if the Placeholder Value is understood and the appropriate replacement value can be determined. If the value is understood, the "<" and ">" MUST be removed, the **name** left intact, and the optional "&" MUST be preserved if present. As an example, "**<ui=UI_LLCC>**" could be rewritten to "ui=en-us&".

Placeholder Values:

UI_LLCC: Indicates that the WOPI server MAY include the preferred UI language in the format described in [\[RFC1766\]](#).

DC_LLCC: Indicates that the WOPI server MAY include preferred data language in the format described in [\[RFC1766\]](#) for cases where language may affect data calculation.

EMBEDDED: Indicates that the WOPI server MAY include the value "true" to use the output of this action embedded in a web page.

DISABLE_ASYNC: Indicates that the WOPI server MAY include the value "true" to prevent the attendee from navigating a file. For example, when using the attendee action (see **st_wopi-action-values** in section 3.1.1.1.2.3.1).

SHOW_THUMBNAILS: Indicates that the WOPI server MAY include the value "true" if the associated action is to show thumbnails as part of the experience.

The following W3C XML schema ([\[XMLSCHEMA1\]](#) section 2.1) fragment specifies the contents of this simple type.

```
<xs:simpleType name="st_wopi-url-source">
  <xs:restriction base="xs:string">
    <xs:pattern value="/^([a-z0-9+.-]*)((?://(?:([a-z0-9-._~!$&'()*+,-;=]|%[0-9A-F]{2})*)@)?(?:([a-z0-9-._~!$&'()*+,-;=]|%[0-9A-F]{2})*)?(\d+)?(?:/([a-z0-9-._~!$&'()*+,-;=:@/]|%[0-9A-F]{2})*)?|(/(?:[a-z0-9-._~!$&'()*+,-;=:@/]|%[0-9A-F]{2})+)(?:[a-z0-9-._~!$&'()*+,-;=:@/]|%[0-9A-F]{2})*)?(\?(\?([a-z0-9-._~!$&'()*+,-;=/?@]|%[0-9A-F]{2})*)?)?(\#[a-z0-9-._~!$&'()*+,-;=/?@]|%[0-9A-F]{2})*)?$/i" />
  </xs:restriction>
</xs:simpleType>
```

3.1.1.1.2.3.4 st_wopi-zone

The **st_wopi-zone** simple type indicates the supported protocols, and the preferred network environments of each set of WOPI actions.

Values:

internal-http: Indicates that the WOPI actions associated with this type are intended for use on internal networks (not internet or extranet networks) and use the HTTP protocol.

internal-https: Indicates that the WOPI actions associated with this type are intended for use on internal networks (not internet or extranet networks) and use the HTTPS protocol.

external-http: Indicates that the WOPI actions associated with this type are intended for use on internet or extranet networks and use the HTTP protocol.

external-https: Indicates that the WOPI actions associated with this type are intended for use on internet or extranet networks and use the HTTPS protocol.

The following W3C XML schema ([\[XMLSCHEMA1\]](#) section 2.1) fragment specifies the contents of this simple type.

```
<xs:simpleType name="st_wopi-zone">
  <xs:restriction base="xs:string">
    <xs:enumeration value="internal-http" />
    <xs:enumeration value="internal-https" />
    <xs:enumeration value="external-http" />
    <xs:enumeration value="external-https" />
  </xs:restriction>
</xs:simpleType>
```

3.1.1.1.1.3 Processing Details

WOPI servers are expected to use the contents of WOPI Discovery XML to associate the functionality described in the XML with the file types in the XML. A WOPI server MAY choose to associate any set

of the functionality described in WOPI Discovery XML but the WOPI server MUST only expose actions that are known to it and that have requirements that it understands and can meet.

3.2 WOPI Protocol Client Details

The WOPI client issues requests to WOPI servers in order to:

- access information about files and folders.
- access the content of files and folders.
- change the content of files and folders.

For details about WOPI client requests, see WOPI Protocol Server Details (see section [3.3](#)).

3.3 WOPI Protocol Server Details

The WOPI server responds to WOPI requests from WOPI clients. It returns information about files and file content in its responses. The WOPI server never initiates communication with the WOPI client.

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

3.3.1 Abstract Data Model

This section describes a conceptual model of possible file state management that an implementation maintains to participate in this protocol. The described model 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 what is described in this document.

The WOPI server SHOULD manage the state of files such that it can respond to WOPI requests appropriately based on the current state of the file. This state management includes but is not limited to:

- User does not have permission to view or edit a file
- File is read-only
- File is locked for editing by another user
- File is checked out by another user

Additionally, the WOPI server SHOULD manage state in accordance with [\[MS-FSSHHTTP\]](#) to enable some specific WOPI server operations as described in section [3.3.5.1.7](#) and section [3.3.5.1.8](#) of this document.

3.3.2 Timers

None.

3.3.3 Initialization

None.

3.3.4 Higher-Layer Triggered Events

None.

3.3.5 Message Processing Events and Sequencing Rules

Resource	Description
HTTP://server/<...>/wopi*/files/<id>	Provides access to information about a file and allows for file-level operations.
HTTP://server/<...>/wopi*/folders/<id>	Provides access to information about a folder and allows for folder level operations.
HTTP://server/<...>/wopi*/files/<id>/contents	Provides access to operations that get and update the contents of a file.
HTTP://server/<...>/wopi*/folder/<id>/children	Provides access to the files and folders in a folder.

3.3.5.1 HTTP://server/<...>/wopi*/files/<id>

The file being accessed by WOPI is identified by the following URI:

HTTP://server/<...>/wopi*/files/<id>

The syntax URI parameters are defined by the following ABNF:

```
id = STRING
```

id: A string that specifies a unique identifier of a file managed by the WOPI server. This string **MUST** be URL safe.

Operation	Description
CheckFileInfo	Returns information about a file
PutRelativeFile	Creates a copy of a file on the WOPI server
Lock	Takes a lock for editing a file
Unlock	Releases a lock for editing a file
RefreshLock	Refreshes a lock for editing a file
UnlockAndRelock	Releases and then retakes a lock for editing a file
ExecuteCellStorageRequest	Changes the contents of the file in accordance with [MS-FSSHTTP]
ExecuteCellStorageRelativeRequest	Changes the contents of the file in accordance with [MS-FSSHTTP]
DeleteFile	Removes a file from the WOPI server

3.3.5.1.1 CheckFileInfo

Return information about the file and permissions that the current user has relative to that file.

HTTP Verb: GET

URI: HTTP://server/<...>/wopi*/files/<id>?access_token=<token>

The response message for this operation can result in the following status codes.

Status code	Description
200	Success
401	Invalid Token
404	File unknown/User Unauthorized
500	Server error

3.3.5.1.1.1 Request Body

None.

3.3.5.1.1.2 Response Body

The response body is JSON (as specified in [RFC4627](#)) with the following parameters:

JSON:

```
{
  "BaseFileName": {"type": "string", "optional": false},
  "BreadcrumbBrandIconUrl": {"type": "string", "default": "", "optional": true},
  "BreadcrumbBrandName": {"type": "string", "default": "", "optional": true},
  "BreadcrumbBrandUrl": {"type": "string", "default": "", "optional": true},
  "BreadcrumbDocName": {"type": "string", "default": "", "optional": true},
  "BreadcrumbDocUrl": {"type": "string", "default": "", "optional": true},
  "BreadcrumbFolderName": {"type": "string", "default": "", "optional": true},
  "BreadcrumbFolderUrl": {"type": "string", "default": "", "optional": true},
  "ClientUrl": {"type": "string", "default": "", "optional": true},
  "CloseUrl": {"type": "string", "default": "", "optional": true},
  "DownloadUrl": {"type": "string", "default": "", "optional": true},
  "FileUrl": {"type": "string", "default": "", "optional": true},
  "HostEditUrl": {"type": "string", "default": "", "optional": true},
  "HostEmbeddedEditUrl": {"type": "string", "default": "", "optional": true},
  "HostEmbeddedViewUrl": {"type": "string", "default": "", "optional": true},
  "HostName": {"type": "string", "default": "", "optional": true},
  "HostViewUrl": {"type": "string", "default": "", "optional": true},
  "IrmPolicyDescription": {"type": "string", "default": "", "optional": true},
  "IrmPolicyTitle": {"type": "string", "default": "", "optional": true},
  "OwnerId": {"type": "string", "optional": false},
  "PrivacyUrl": {"type": "string", "default": "", "optional": true},
  "ProtectInClient": {"type": "bool", "default": false, "optional": true},
  "ReadOnly": {"type": "bool", "default": false, "optional": true},
  "SHA256": {"type": "string", "optional": false},
  "SignoutUrl": {"type": "string", "default": "", "optional": true},
  "Size": {"type": "int", "optional": false},
  "SupportsCoauth": {"type": "bool", "default": false, "optional": true},
  "SupportsCobalt": {"type": "bool", "default": false, "optional": true},
  "SupportsDeletePartition": {"type": "bool", "default": false, "optional": true},
  "SupportsFolders": {"type": "bool", "default": false, "optional": true},
  "SupportsLocks": {"type": "bool", "default": false, "optional": true},
  "SupportsUpdate": {"type": "bool", "default": false, "optional": true},
  "TermsOfUseUrl": {"type": "string", "default": "", "optional": true},
}
```

```

    "UserCanAttend":{"type":"bool","default":false,"optional":true},
    "UserCanPresent":{"type":"bool","default":false,"optional":true},
    "UserCanWrite":{"type":"bool","default":false,"optional":true},
    "UserFriendlyName":{"type":"string","default":"","optional":true},
    "UserId":{"type":"string","default":"","optional":true},
    "Version":{"type":"string","optional":false}
  }

```

BaseFileName: The name of the file without the path. Typically used for display in UI, and determining the extension of the file.

BreadcrumbBrandIconUri: A URI to an image that the WOPI client MAY display to the user as the branding image of the WOPI server.

BreadcrumbBrandName: A string that the WOPI client MAY display to the user that indicates the brand name of the WOPI server.

BreadcrumbBrandUri: A URI to a web page that the WOPI client MAY navigate to when the user clicks on UI that displays BreadcrumbBrandName.

BreadcrumbDocName: A string that the WOPI client MAY display to the user that indicates the name of the file.

BreadcrumbDocUri: A URI to a web page that the WOPI client MAY navigate to when the user clicks on UI that displays BreadcrumbDocName.

BreadcrumbFolderName: A string that the WOPI client MAY display to the user that indicates the name of the folder that contains the file.

BreadcrumbFolderUri: A URI to a web page that the WOPI client MAY navigate to when the user clicks on UI that displays BreadcrumbFolderName.

ClientUri: A user-accessible URI directly to the file intended for opening the file through a client. Typically a DAV URL ([RFC5323](#)) but MAY be any URL that can be handled by a client that can open a file of the given type.

CloseURL: A URI to a web page that the implementer deems useful to a user in the event that the user closes the rendering or editing client currently using this file.

DownloadUri: A user-accessible URI to the file intended to allow the user to download a copy of the file.

FileUri: A URI to the file location that the WOPI client uses to get the file. If this is provided, a WOPI client MUST use this URI to get the file instead of `HTTP://server/<...>/wopi*/files/<id>/contents` (see section [3.3.5.3](#)).

HostEditUri: A URI to a web page that provides an editing experience for the file, utilizing the WOPI client.

HostEmbeddedEditUri: A URI to a web page that provides access to an editing experience for the file that can be embedded in another HTML page. For example, a page that provides an HTML snippet that can be inserted into the HTML of a blog.

HostEmbeddedViewUri: A URI to a web page that provides access to a viewing experience for the file that can be embedded in another HTML page. For example, a page that provides an HTML snippet that can be inserted into the HTML of a blog.

HostName: A string that is the name provided by the WOPI server used to identify it for logging and other informational purposes.

HostViewUrl: A URI to a web page that provides a viewing experience for the file utilizing the WOPI client.

IrmPolicyDescription: A string that the WOPI client SHOULD display to the user indicating the **IRM** policy for the file. This value SHOULD be combined with IrmPolicyTitle.

IrmPolicyTitle: A string that the WOPI client SHOULD display to the user indicating the IRM policy for the file. This value SHOULD be combined with IrmPolicyDescription.

OwnerId: A string that SHOULD uniquely identify the owner of the file.

PrivacyUrl: A URI to a webpage that explains the privacy policy of the WOPI server.

ProtectInClient: A Boolean value that indicates that the WOPI client SHOULD take measures to prevent copying and printing of the file. This is intended to help enforce IRM in WOPI clients.

ReadOnly: Indicates that, for this user, the file cannot be changed.

SHA256: A 256 bit SHA-2-encoded [\[FIPS180-2\]](#) hash of the file contents.

SignoutUrl: A URI that will sign the current user out of the WOPI server supported authentication system.

Size: The size of the file expressed in bytes.

SupportsCoauth: A Boolean value that indicates that the WOPI server supports multiple users making changes to this file simultaneously.

SupportsCobalt: A Boolean value that indicates that the WOPI server supports **ExecuteCellStorageRequest** (see section [3.3.5.1.7](#)) and **ExecuteCellStorageRelativeRequest** (see section [3.3.5.1.8](#)) operations for this file.

SupportsDeletePartition: Deprecated.

SupportsFolders: A Boolean value that indicates that the WOPI server supports **EnumerateChildren** (see section [3.3.5.4.1](#)) and **DeleteFile** (see section [3.3.5.1.9](#)) operations for this file.

SupportsLocks: A Boolean value that indicates that the WOPI server supports **Lock** (see section [3.3.5.1.3](#)), **Unlock** (see section [3.3.5.1.4](#)), **RefreshLock** (see section [3.3.5.1.5](#)) and **UnlockAndRelock** (see section [3.3.5.1.6](#)) operations for this file.

SupportsUpdate: A Boolean value that indicates that the WOPI server supports **PutFile** (see section [3.3.5.3.2](#)) and **PutRelativeFile** (see section [3.3.5.1.2](#)) operations for this file.

TermsOfUseUrl: A URI to a webpage that explains the terms of use policy of the WOPI server.

UserCanAttend: A Boolean value that indicates that the user has permission to view a broadcast of this file. A broadcast is file activity that involves one or more presenters controlling the view of the file for a set of attendees. For example, a slideshow may be broadcast by a presenter to many attendees.

UserCanPresent: A Boolean value that indicates that the user has permission to broadcast this file to a set of users who have permission to broadcast or view a broadcast of this file. A broadcast is

file activity that involves one or more presenters controlling the view of the file for a set of attendees. For example, a slideshow may be broadcast by a presenter to many attendees.

UserCanWrite: Indicates that the user has permissions to alter the file.

UserFriendlyName: A string that is the name of the user. If blank, the WOPI client MAY choose to use a placeholder string in some scenarios or show no name at all.

UserId: A string that is used by the WOPI server to uniquely identify the user.

Version: The current version of the file based on the server's file versioning schema. This value MUST change when the file changes, and version values MUST never repeat for a given file.

3.3.5.1.1.3 Processing Details

The **id** represents the file that the WOPI Server is providing information about. The **token** represents the credentials that the WOPI Server MUST use when determining the permissions the WOPI Client has when interacting with the file.

3.3.5.1.2 PutRelativeFile

Create a new file on the WOPI server based on the current file.

HTTP Verb: POST

URI: HTTP://server/<...>/wopi*/files/<id>?access_token=<token>

The request message for this operation contains the following HTTP headers.

Request header	Usage	Value
X-WOPI-Override	A string specifying the requested operation from the WOPI server. Required.	The string "PUT_RELATIVE".
X-WOPI-SuggestedTarget	A string specifying either a file extension or a full file name. If only the extension is provided, the name of the initial file without extension SHOULD be combined with the extension to create the proposed name. The WOPI server MUST modify the proposed name as needed in order to create a new file that is both legally named and does not overwrite any existing file while preserving the file extension. This header MUST be present if X-WOPI-RelativeTarget is not present.	A string that is a file extension or a full file name.
X-WOPI-RelativeTarget	A string that specifies a file name. The WOPI server MUST NOT modify the name in order to fulfill the request.	A string that is a full file name.
X-WOPI-OverwriteRelativeTarget	A Boolean value that specifies whether the host MUST overwrite the file name if it exists.	A Boolean value.
X-WOPI-Size	A integer that specifies the size of the file in bytes	An integer.

The response message for this operation can result in the following status codes.

Status code	Description
200	Success
400	Specified name is illegal
401	Token is invalid
404	File unknown/User Unauthorized
409	Target file already exists or is locked
413	File is too large
500	Server error
501	Unsupported

3.3.5.1.2.1 Request Body

The binary contents of the file.

3.3.5.1.2.2 Response Body

The response body is JSON (as specified in [RFC4627](#)) with the following parameters:

JSON:

```
{
  "Name":{"type":"string","optional":false},
  "Url":{"type":"string","default":"","optional":false},
  "HostViewUrl":{"type":"string","default":"","optional":true},
  "HostEditUrl":{"type":"string","default":"","optional":true},
}
```

Name: The name of the file without the path.

URL: A URI that is the WOPI Server URI of the newly created file in the form:

HTTP://server/<...>/wopi*/files/<id>?access_token=<token>

Where **id** is the WOPI server's **id** for the new file, and **token** is the WOPI server's credentials token for accessing the new file.

HostViewUrl: A URI to a web page that provides a viewing experience for the new file utilizing the WOPI client.

HostEditUrl: A URI to a web page that provides an editing experience for the new file utilizing the WOPI client.

3.3.5.1.2.3 Processing Details

The WOPI server MUST use the content in the POST body to create a new file. The rules for naming the new file and handling errors are described in the introductory section of this operation.

3.3.5.1.3 Lock

Take a lock for editing a file.

HTTP Verb: POST

URI: HTTP://server/<...>/wopi*/files/<id>?access_token=<token>

The request message for this operation contains the following HTTP headers.

Request header	Usage	Value
X-WOPI-Override	A string specifying the requested operation from the WOPI server. Required.	The string "LOCK".
X-WOPI-Lock	A string provided by the WOPI client that the WOPI server MUST use to identify the lock on the file. Required.	A string.

The response message for this operation contains the following HTTP headers.

Request header	Usage	Value
X-WOPI-OldLock	A string provided by the WOPI client that the WOPI server MUST use to identify the lock on the file. Required.	A string.

The response message for this operation can result in the following status codes.

Status code	Description
200	Success
401	Token is invalid
404	File unknown/User Unauthorized
409	Lock mismatch/Locked by another interface
500	Server error
501	Unsupported

3.3.5.1.3.1 Request Body

None.

3.3.5.1.3.2 Response Body

None.

3.3.5.1.3.3 Processing Details

The WOPI server MUST use the string provided in the X-WOPI-Lock header to create a lock on a file. When a WOPI client has a lock on a file the WOPI client is able to safely make changes to the file. The lock MUST last for 30 minutes, unless modified by a subsequent WOPI client operation.

3.3.5.1.4 Unlock

Release a lock for editing a file.

HTTP Verb: POST

URI: HTTP://server/<...>/wopi*/files/<id>?access_token=<token>

The request message for this operation contains the following HTTP headers.

Request header	Usage	Value
X-WOPI-Override	A string specifying the requested operation from the WOPI server. Required.	The string "UNLOCK".
X-WOPI-Lock	A string provided by the WOPI client that the WOPI server MUST use to identify the lock on the file. Required.	The string provided by the WOPI client through Lock (see section 3.3.5.1.3)

The response message for this operation can result in the following status codes.

Status code	Description
200	Success
401	Token is invalid
404	File unknown/User Unauthorized
409	Lock mismatch/Locked by another interface
500	Server error
501	Unsupported

3.3.5.1.4.1 Request Body

None.

3.3.5.1.4.2 Response Body

None.

3.3.5.1.4.3 Processing Details

By releasing a lock the WOPI client is indicating that it is not making further changes to a file.

3.3.5.1.5 RefreshLock

Refresh an existing lock for modifying a file.

HTTP Verb: POST

URI: HTTP://server/<...>/wopi*/files/<id>?access_token=<token>

The request message for this operation contains the following HTTP headers.

Request header	Usage	Value
X-WOPI-Override	A string specifying the requested operation from the WOPI server. Required.	The string "REFRESH_LOCK".
X-WOPI-Lock	A string provided by the WOPI client that the WOPI server MUST use to identify the lock on the file. Required.	The string provided by the WOPI client through Lock (see section 3.3.5.1.3)

The response message for this operation can result in the following status codes.

Status code	Description
200	Success
401	Invalid token
404	File unknown/User Unauthorized
409	Lock mismatch/Locked by another interface
500	Server error
501	Unsupported

3.3.5.1.5.1 Request Body

None.

3.3.5.1.5.2 Response Body

None.

3.3.5.1.5.3 Processing Details

Renews an existing lock on a file. The lock MUST expire 30 minutes after this operation unless the lock is modified by a subsequent WOPI client operation.

3.3.5.1.6 UnlockAndRelock

Release and retake a lock for editing a file.

HTTP Verb: POST

URI: HTTP://server/<...>/wopi*/files/<id>?access_token=<token>

The request message for this operation contains the following HTTP headers.

Request header	Usage	Value
X-WOPI-Override	A string specifying the requested operation from the WOPI server. Required.	The string "LOCK".
X-WOPI-	A string provided by the WOPI client that the WOPI server MUST use to identify the lock on the file.	A string.

Request header	Usage	Value
Lock	Required.	
X-WOPI-OldLock	A string previously provided by the WOPI client that the WOPI server MUST have used to identify the lock on the file. Required.	The string provided by the WOPI client through Lock (see section 3.3.5.1.3)

The response message for this operation can result in the following status codes.

Status code	Description
200	Success
401	Invalid token
404	File unknown/User Unauthorized
409	Lock mismatch/Locked by another interface
500	Server error
501	Unsupported

3.3.5.1.6.1 Request Body

None.

3.3.5.1.6.2 Response Body

None.

3.3.5.1.6.3 Processing Details

The WOPI server uses the X-WOPI-OldLock to identify which lock to release, and the X-WOPI-Lock as the **id** of the new lock. The new lock MUST expire 30 minutes after this operation, unless modified by a subsequent WOPI client action.

3.3.5.1.7 ExecuteCellStorageRequest

Make a change to a file using the standards defined by [\[MS-FSSHTTP\]](#).

HTTP Verb: POST

URI: HTTP://server/<...>/wopi*/files/<id>?access_token=<token>

The request message for this operation contains the following HTTP headers.

Request header	Usage	Value
X-WOPI-Override	A string specifying requested operation from the WOPI server. Required	The string "COBALT".
X-WOPI-Size	An integer specifying the size of the request body.	An integer.

The response message for this operation can result in the following status codes.

Status code	Description
200	Success
401	Invalid token
404	File unknown/User Unauthorized
500	Server error
501	Unsupported

3.3.5.1.7.1 Request Body

Content that conforms to [\[MS-FSSHTTP\]](#).

3.3.5.1.7.2 Response Body

Content that conforms to [\[MS-FSSHTTP\]](#).

3.3.5.1.7.3 Processing Details

Both the WOPI client and the WOPI server MUST follow the interaction model defined in [\[MS-FSSHTTP\]](#). Success indicates that the request executed correctly.

3.3.5.1.8 ExecuteCellStorageRelativeRequest

Make a change or create a new file in a location relative to the current file or folder using the standards defined by [\[MS-FSSHTTP\]](#).

HTTP Verb: POST

URI: HTTP://server/<...>/wopi*/files/<id>?access_token=<token>

The request message for this operation contains the following HTTP headers.

Request header	Usage	Value
X-WOPI-Override	A string specifying requested operation from the WOPI server. Required	The string "COBALT".
X-WOPI-Size	An integer specifying the size of the request body.	An integer.
X-WOPI-RelativeTarget	A string that is the name of the file being created or changed by this operation. This location of this file is relative to the current file or folder. That is, it MUST be a sibling of a file and a child of a folder.	A string.

The response message for this operation can result in the following status codes.

Status code	Description
200	Success
401	Invalid Token

Status code	Description
404	File unknown/User Unauthorized
500	Server error
501	Unsupported

3.3.5.1.8.1 Request Body

Content that conforms to [\[MS-FSSHTTP\]](#).

3.3.5.1.8.2 Response Body

Content that conforms to [\[MS-FSSHTTP\]](#).

3.3.5.1.8.3 Processing Details

Both the WOPI client and the WOPI server MUST follow the interaction model defined in [\[MS-FSSHTTP\]](#). Success indicates that the request executed correctly.

3.3.5.1.9 DeleteFile

Delete a file.

HTTP Verb: POST

URI: HTTP://server/<...>/wopi*/files/<id>?access_token=<token>

The request message for this operation contains the following HTTP headers.

Request header	Usage	Value
X-WOPI-Override	A string specifying requested operation from the WOPI server. Required	The string "DELETE".

The response message for this operation can result in the following status codes.

Status code	Description
200	Success
401	Invalid Token
404	File unknown/User Unauthorized
409	Lock mismatch/Locked by another interface
500	Server error
501	Unsupported

3.3.5.1.9.1 Request Body

None.

3.3.5.1.9.2 Response Body

None.

3.3.5.1.9.3 Processing Details

The WOPI server MUST delete the file if possible given the permissions and state of the file.

3.3.5.2 HTTP://server/<...>/wopi*/folders/<id>

The folder being accessed by WOPI is identified by the following URI:

HTTP://server/<...>/wopi*/folders/<id>

The syntax URI parameters are defined by the following ABNF:

```
id = STRING
```

id: A unique identifier of a folder managed by the WOPI server.

Operation	Description
CheckFolderInfo	Returns information about a folder.

3.3.5.2.1 CheckFolderInfo

Return information about the folder and permissions that the current user has relative to that file.

HTTP Verb: GET

URI: HTTP://server/<...>/wopi*/folders/<id>?access_token=<token>

3.3.5.2.1.1 Request Body

None.

3.3.5.2.1.2 Response Body

The response body is JSON (as specified in [RFC4627](#)) with the following parameters:

JSON:

```
{
  "FolderName": {"type": "string", "optional": false},
  "BreadcrumbBrandImageUrl": {"type": "string", "default": "", "optional": true},
  "BreadcrumbBrandName": {"type": "string", "default": "", "optional": true},
  "BreadcrumbBrandUrl": {"type": "string", "default": "", "optional": true},
  "BreadcrumbDocName": {"type": "string", "default": "", "optional": true},
  "BreadcrumbDocUrl": {"type": "string", "default": "", "optional": true},
  "BreadcrumbFolderName": {"type": "string", "default": "", "optional": true},
  "BreadcrumbFolderUrl": {"type": "string", "default": "", "optional": true},
  "ClientUrl": {"type": "string", "default": "", "optional": true},
  "CloseUrl": {"type": "string", "default": "", "optional": true},
  "HostEditUrl": {"type": "string", "default": "", "optional": true},
  "HostEmbeddedEditUrl": {"type": "string", "default": "", "optional": true},
}
```

```

"HostEmbeddedViewUrl":{"type":"string","default":"","optional":true},
"HostName":{"type":"string","default":"","optional":true},
"HostViewUrl":{"type":"string","default":"","optional":true},
"OwnerId":{"type":"string","optional":false},
"PrivacyUrl":{"type":"string","default":"","optional":true},
"SignoutUrl":{"type":"string","default":"","optional":true},
"TermsOfUseUrl":{"type":"string","default":"","optional":true},
"UserCanWrite":{"type":"bool","default":false,"optional":true},
"UserFriendlyName":{"type":"string","default":"","optional":true},
"UserId":{"type":"string","default":"","optional":true},
}

```

FolderName: The name of the folder without the path. Typically used for display in UI.

BreadcrumbBrandIconUri: A URI to an image that the WOPI client MAY display to the user as the branding image of the WOPI server.

BreadcrumbBrandName: A string that the WOPI client MAY display to the user that indicates the brand name of the WOPI server.

BreadcrumbBrandUri: A URI to a web page that the WOPI client MAY navigate to when the user clicks on UI that displays BreadcrumbBrandName.

BreadcrumbDocName: A string that the WOPI client MAY display to the user that indicates the name of the file.

BreadcrumbDocUri: A URI to a web page that the WOPI client MAY navigate to when the user clicks on UI that displays BreadcrumbDocName.

BreadcrumbFolderName: A string that the WOPI client MAY display to the user that indicates the name of the folder that contains the file.

BreadcrumbFolderUri: A URI to a web page that the WOPI client MAY navigate to when the user clicks on UI that displays BreadcrumbFolderName.

ClientUri: A user-accessible URI directly to the folder intended for opening the file through a client. Typically a DAV URL ([\[RFC5323\]](#)) but MAY be any URL that can be handled by a client that can open a file of the given type.

CloseURL: A URI to a web page that the implementer deems useful to a user in the event that the user closes the rendering or editing client currently using this folder.

HostEditUri: A URI to a web page that provides an editing experience for the folder utilizing the WOPI client.

HostEmbeddedEditUri: A URI to a web page that provides access to an editing experience for the folder that can be embedded in another HTML page. For example, a page that provides an HTML snippet that can be inserted into the HTML of a blog.

HostEmbeddedViewUri: A URI to a web page that provides access to a viewing experience for the folder that can be embedded in another HTML page. For example, a page that provides an HTML snippet that can be inserted into the HTML of a blog.

HostName: A string that is the name provided by the WOPI server used to identify it for logging and other informational purposes.

HostViewUrl: A URI to a web page that provides a viewing experience for the folder utilizing the WOPI client.

OwnerId: A string that SHOULD uniquely identify the owner of the file.

PrivacyUrl: A URI to a webpage that explains the privacy policy of the WOPI server.

SignoutUrl: A URI that will sign the current user out of the WOPI server supported authentication system.

TermsOfUseUrl: A URI to a webpage that explains the terms of use policy of the WOPI server.

UserCanWrite: Indicates that the user has permissions to alter the folder.

UserFriendlyName: A string that is the name of the user. If blank, the WOPI client MAY choose to use a placeholder string in some scenarios or show no name at all.

UserId: A string that is used by the WOPI server to uniquely identify the user.

3.3.5.2.1.3 Processing Details

The **id** represents the folder that the WOPI Server is providing information about. The **token** represents the credentials that the WOPI Server MUST use when determining the permissions the WOPI Client has when interacting with the folder.

3.3.5.3 HTTP://server/<...>/wopi*/files/<id>/contents

The content of a file being accessed by WOPI is identified by the following URI:

HTTP://server/<...>/wopi*/files/<id>/contents

The syntax URI parameters are defined by the following ABNF:

```
id = STRING
```

id: A unique identifier of a file managed by the WOPI server.

Operation	Description
GetFile	Request message to retrieve a file for the HTTP://server/<...>/wopi*/files/<id>/contents operation.
PutFile	Request message to update a file for the HTTP://server/<...>/wopi*/files/<id>/contents operation.

3.3.5.3.1 GetFile

Get a file.

HTTP Verb: GET

URI: HTTP://server/<...>/wopi*/files/<id>/contents?access_token=<token>

The request message for this operation contains the following HTTP headers.

Request header	Usage	Value
X-WOPI-MaxExpectedSize	An integer specifying the upper bound of the expected size of the file being requested. Optional.	An integer.

The response message for this operation can result in the following status codes.

Status code	Description
200	Success
401	Invalid Token
404	File unknown/User Unauthorized
412	File is larger than X-WOPI-MaxExpectedSize
500	Server error

3.3.5.3.1.1 Request Body

None.

3.3.5.3.1.2 Response Body

The binary contents of the file.

3.3.5.3.1.3 Processing Details

The WOPI server MUST return the complete binary of the file identified by <id> in the response body, if the response indicates Success.

3.3.5.3.2 PutFile

Update a file on the WOPI server.

HTTP Verb: POST

URI: HTTP://server/<...>/wopi*/files/<id>/contents?access_token=<token>

The request message for this operation contains the following HTTP headers.

Request header	Usage	Value
X-WOPI-Override	A string specifying requested operation from the WOPI server. Required	The string "PUT".
X-WOPI-Size	An integer specifying the size of the request body. Optional.	An integer.
X-WOPI-Lock	A string provided by the WOPI client that the WOPI server used to identify the lock on the file. Required.	The string provided by the WOPI client through Lock (see section 3.3.5.1.3)

The response message for this operation can result in the following status codes.

Status code	Description
200	Success
401	Invalid Token
404	File unknown/User Unauthorized
409	Lock mismatch/Locked by another interface
413	File is too large
500	Server error
501	Unsupported

3.3.5.3.2.1 Request Body

The binary contents of the file.

3.3.5.3.2.2 Response Body

None.

3.3.5.3.2.3 Processing Details

The WOPI server MUST update the binary of the file identified by <id> to match the binary contents in the request body, if the response indicates Success. If the file is currently associated with a lock established by the **Lock** operation (see section [3.3.5.1.3](#)) or the **UnlockAndRelock** operation (see section [3.3.5.1.6](#)) the WOPI server MUST provide the matching lock value in order for this operation to succeed.

3.3.5.4 HTTP://server/<...>/wopi*/folder/<id>/children

The contents of a folder being accessed by WOPI are identified by the following URI:

HTTP://server/<...>/wopi*/folder/<id>/children

The syntax URI parameters are defined by the following ABNF:

```
id = STRING
```

id: A unique identifier of a folder managed by the WOPI server.

Operation	Description
EnumerateChildren	Returns a set of URIs that provides access to resources in the folder.

3.3.5.4.1 EnumerateChildren

The EnumerateChildren method returns the contents of a folder on the WOPI server.

HTTP Verb: GET

URI: HTTP://server/<...>/wopi*/folder/<id>/contents?access_token=<token>

The response message for this operation can result in the following status codes.

Status code	Description
200	Success
401	Invalid Token
404	File unknown/User Unauthorized
500	Server error
501	Unsupported

3.3.5.4.1.1 Request Body

None.

3.3.5.4.1.2 Response Body

The response body is JSON (as specified in [RFC4627](#)) with the following parameters:

JSON:

```
{
  "Children":
  [
    {
      "Name": "<name>",
      "Url": "<url>",
      "Version": "<version>"
    },
  ]
}
```

Name: The name of the child resource.

Url: The URI if the child resource of the form `http://server/<...>/wopi*/files/<id>?access_token=<token>` where **id** is the WOPI server's unique **id** of the resource and **token** is the **token** that provides access to the resource.

Version: The current version of the resource based on the server's file versioning schema. This value **MUST** change when the file changes and **MUST** match the value that would be provided by the "Version" field in the response to CheckFileInfo (see section [3.3.5.1.1](#)).

3.3.5.4.1.3 Processing Details

The WOPI server returns URLs in the response body JSON that the WOPI client **MAY** then use to initiate further WOPI operations on the items in the folder.

3.3.6 Timer Events

None.

3.3.7 Other Local Events

None.

4 Protocol Examples

The following two scenarios show how a WOPI server discovers a WOPI client and how a user can view a document on that WOPI server.

4.1 Accessing Discovery XML

In this scenario, the WOPI server discovers a WOPI client through WOPI Discovery.

A WOPI server administrator uses the administrator console to instruct the WOPI server to request Discovery XML from a WOPI client. Typically, the administrator need only provide the host name of the WOPI client as the specific URI will always be `http://server/hosting/discovery`.

The WOPI server makes the request to `http://server/hosting/discovery` and receives the Discovery XML. Based on the content of the Discovery XML, the WOPI server is now able to offer viewing of .docx files through the WOPI client.

The WOPI server updates its UI to reflect this new functionality.

4.2 Viewing a Document

In this scenario, a user of a WOPI server is able to view a .docx file through a WOPI client.

A user of a WOPI server identifies the .docx file to be viewed in a browser. The WOPI server exposes UI that allows the user to do this through the WOPI client that the WOPI server has already learned about via Discovery XML.

The user clicks the UI. The WOPI server navigates the user's browser to a new web page. On that web page the WOPI server creates an `iframe` HTML element that navigates to the URI for viewing .docx files. This URI was constructed using the `urlsrc` that the WOPI server learned about through Discovery XML.

The WOPI client uses the `id` and `token` provided by the WOPI server to access the .docx file on the WOPI server. The WOPI client renders a view of this file inside the `iframe` created by the WOPI server.

5 Security

5.1 Security Considerations for Implementers

The WOPI protocol uses a token to pass credentials from between the WOPI server, the WOPI client and the user client (for example, a browser). This token is a string and, if compromised, allows anyone with the token and information about the file to act as the original user in the context of that single file. To protect this token use HTTPS for all WOPI network traffic.

5.2 Index of Security Parameters

None.

6 Appendix A: 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 id="Discovery"
  targetNamespace="http://tempuri.org/Discovery.xsd"
  elementFormDefault="qualified"
  xmlns="http://tempuri.org/Discovery.xsd"
  xmlns:mstns="http://tempuri.org/Discovery.xsd"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  >
  <xs:simpleType name="st_wopi-action-values">
    <xs:restriction base="xs:string">
      <xs:enumeration value="view" />
      <xs:enumeration value="edit" />
      <xs:enumeration value="mobileView" />
      <xs:enumeration value="embedview" />
      <xs:enumeration value="mobileclient" />
      <xs:enumeration value="present" />
      <xs:enumeration value="attend" />
      <xs:enumeration value="editnew" />
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="st_wopi-action-requirements">
    <xs:restriction base="xs:string">
      <xs:pattern
        value="/^((update|locks|cobalt|containers) \s*(, \s*|$)) * ((update|locks|cobalt|containers) \s*$)
        /" />
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="st_wopi-url-source">
    <xs:restriction base="xs:string">
      <xs:pattern value="REGEX FOR URI" />
    </xs:restriction>
  </xs:simpleType>

  <xs:complexType name="ct_wopi-action">
    <xs:attribute name="name" use="required" type="mstns:st_wopi-action-values"/>
    <xs:attribute name="default" type="xs:boolean" default="false"/>
    <xs:attribute name="requires" type="st_wopi-action-requirements"/>
    <xs:attribute name="urlsrc" use="required" type="mstns:st_wopi-url-source"/>
    <xs:attribute name="ext" type="xs:string" use="optional"/>
    <xs:attribute name="progid" type="xs:string" use="optional"/>
    <xs:attribute name="newprogid" type="xs:string" use="optional"/>
    <xs:attribute name="newext" type="xs:string" use="optional"/>
  </xs:complexType>

  <xs:simpleType name="st_wopi-zone">
    <xs:restriction base="xs:string">
      <xs:enumeration value="internal-http" />
      <xs:enumeration value="internal-https" />
      <xs:enumeration value="external-http" />
      <xs:enumeration value="external-https" />
    </xs:restriction>
  </xs:simpleType>
```

```
<xs:complexType name="ct_app-name">
  <xs:sequence>
    <xs:element name="action" minOccurs="1" maxOccurs="unbounded" type="ct_wopi-action"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="ct_net-zone">
  <xs:sequence>
    <xs:element name="app-name" minOccurs="1" maxOccurs="unbounded" type="ct_app-name"/>
  </xs:sequence>
  <xs:attribute name="name" type="mstns:st_wopi-zone"/>
</xs:complexType>

<xs:complexType name="ct_wopi-discovery">
  <xs:sequence>
    <xs:element name="net-zone" minOccurs="1" maxOccurs="4" type="ct_net-zone"/>
  </xs:sequence>
</xs:complexType>

<xs:element name="wopi-discovery" type="ct_wopi-discovery"/>

</xs:schema>
```


7 Appendix B: Full JSON Schema

For ease of implementation, the following sections provide the full JSON schema for this protocol.

Schema name	Section
CheckFileInfo JSON	7.1
CheckFolderInfo JSON	7.2
EnumerateChildren JSON	7.3

7.1 CheckFileInfo JSON

```
{
  "BaseFileName": {"type": "string", "optional": false},
  "BreadcrumbBrandIconUrl": {"type": "string", "default": "", "optional": true},
  "BreadcrumbBrandName": {"type": "string", "default": "", "optional": true},
  "BreadcrumbBrandUrl": {"type": "string", "default": "", "optional": true},
  "BreadcrumbDocName": {"type": "string", "default": "", "optional": true},
  "BreadcrumbDocUrl": {"type": "string", "default": "", "optional": true},
  "BreadcrumbFolderName": {"type": "string", "default": "", "optional": true},
  "BreadcrumbFolderUrl": {"type": "string", "default": "", "optional": true},
  "ClientUrl": {"type": "string", "default": "", "optional": true},
  "CloseUrl": {"type": "string", "default": "", "optional": true},
  "DownloadUrl": {"type": "string", "default": "", "optional": true},
  "FileUrl": {"type": "string", "default": "", "optional": true},
  "HostEditUrl": {"type": "string", "default": "", "optional": true},
  "HostEmbeddedEditUrl": {"type": "string", "default": "", "optional": true},
  "HostEmbeddedViewUrl": {"type": "string", "default": "", "optional": true},
  "HostName": {"type": "string", "default": "", "optional": true},
  "HostViewUrl": {"type": "string", "default": "", "optional": true},
  "IrmPolicyDescription": {"type": "string", "default": "", "optional": true},
  "IrmPolicyTitle": {"type": "string", "default": "", "optional": true},
  "OwnerId": {"type": "string", "optional": false},
  "PrivacyUrl": {"type": "string", "default": "", "optional": true},
  "ProtectInClient": {"type": "bool", "default": false, "optional": true},
  "ReadOnly": {"type": "bool", "default": false, "optional": true},
  "SHA256": {"type": "string", "optional": false},
  "SignoutUrl": {"type": "string", "default": "", "optional": true},
  "Size": {"type": "int", "optional": false},
  "SupportsCoauth": {"type": "bool", "default": false, "optional": true},
  "SupportsCobalt": {"type": "bool", "default": false, "optional": true},
  "SupportsDeletePartition": {"type": "bool", "default": false, "optional": true},
  "SupportsFolders": {"type": "bool", "default": false, "optional": true},
  "SupportsLocks": {"type": "bool", "default": false, "optional": true},
  "SupportsUpdate": {"type": "bool", "default": false, "optional": true},
  "TermsOfUseUrl": {"type": "string", "default": "", "optional": true},
  "UserCanAttend": {"type": "bool", "default": false, "optional": true},
  "UserCanPresent": {"type": "bool", "default": false, "optional": true},
  "UserCanWrite": {"type": "bool", "default": false, "optional": true},
  "UserFriendlyName": {"type": "string", "default": "", "optional": true},
  "UserId": {"type": "string", "default": "", "optional": true},
  "Version": {"type": "string", "optional": false}
}
```

7.2 CheckFolderInfo JSON

```
{
  "FolderName":{"type":"string","optional":false},
  "BreadcrumbBrandIconUrl":{"type":"string","default":"","optional":true},
  "BreadcrumbBrandName":{"type":"string","default":"","optional":true},
  "BreadcrumbBrandUrl":{"type":"string","default":"","optional":true},
  "BreadcrumbDocName":{"type":"string","default":"","optional":true},
  "BreadcrumbDocUrl":{"type":"string","default":"","optional":true},
  "BreadcrumbFolderName":{"type":"string","default":"","optional":true},
  "BreadcrumbFolderUrl":{"type":"string","default":"","optional":true},
  "ClientUrl":{"type":"string","default":"","optional":true},
  "CloseUrl":{"type":"string","default":"","optional":true},
  "HostEditUrl":{"type":"string","default":"","optional":true},
  "HostEmbeddedEditUrl":{"type":"string","default":"","optional":true},
  "HostEmbeddedViewUrl":{"type":"string","default":"","optional":true},
  "HostName":{"type":"string","default":"","optional":true},
  "HostViewUrl":{"type":"string","default":"","optional":true},
  "OwnerId":{"type":"string","optional":false},
  "PrivacyUrl":{"type":"string","default":"","optional":true},
  "SignoutUrl":{"type":"string","default":"","optional":true},
  "TermsOfUseUrl":{"type":"string","default":"","optional":true},
  "UserCanWrite":{"type":"bool","default":false,"optional":true},
  "UserFriendlyName":{"type":"string","default":"","optional":true},
  "UserId":{"type":"string","default":"","optional":true},
}
```

7.3 EnumerateChildren JSON

```
{
  "Children":
  [
    {
      "Name":"<name>",
      "Url":"<url>",
      "Version":"<version>"
    },
  ]
}
```

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 Server 15 Technical Preview
- Microsoft® Exchange Server 15 Technical Preview
- Microsoft® Office 15 Technical Preview
- Microsoft® SharePoint® Server 15 Technical Preview
- Microsoft® SharePoint® Foundation 15 Technical 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.

9 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

Preliminary

10 Index

A

[Applicability](#) 9

[Vendor-extensible fields](#) 9
[Versioning](#) 9

C

[Capability negotiation](#) 9

[Change tracking](#) 44

F

[Fields - vendor-extensible](#) 9

G

[Glossary](#) 6

I

[Implementer - security considerations](#) 38

[Index of security parameters](#) 38

[Informative references](#) 7

[Introduction](#) 6

N

[Normative references](#) 6

O

[Overview \(synopsis\)](#) 7

P

[Parameters - security index](#) 38

[Preconditions](#) 9

[Prerequisites](#) 9

[Product behavior](#) 43

R

References

[informative](#) 7

[normative](#) 6

[Relationship to other protocols](#) 8

S

Security

[implementer considerations](#) 38

[parameter index](#) 38

[Standards assignments](#) 9

T

[Tracking changes](#) 44

V