

[MS-WOPI]: Web Application Open Platform Interface 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, email 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.

Revision Summary

Date	Revision History	Revision Class	Comments
01/20/2012	0.1	New	Released new document.
04/11/2012	0.1	No change	No changes to the meaning, language, or formatting of the technical content.
07/16/2012	0.2	Minor	Clarified the meaning of the technical content.
10/08/2012	1.0	Major	Significantly changed the technical content.
02/11/2013	1.0	No change	No changes to the meaning, language, or formatting of the technical content.
07/30/2013	1.0	No change	No changes to the meaning, language, or formatting of the technical content.
11/18/2013	1.0	No change	No changes to the meaning, language, or formatting of the technical content.
02/10/2014	1.0	No change	No changes to the meaning, language, or formatting of the technical content.
04/30/2014	2.0	Major	Significantly changed the technical content.
07/31/2014	2.1	Minor	Clarified the meaning of the technical content.
10/30/2014	2.2	Minor	Clarified the meaning of the technical content.

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.....	9
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	11
3	Protocol Details.....	13
3.1	WOPI Discovery Details	13
3.1.1	Abstract Data Model	13
3.1.2	Timers	13
3.1.3	Initialization	13
3.1.4	Higher-Layer Triggered Events.....	13
3.1.5	Message Processing Events and Sequencing Rules.....	13
3.1.5.1	HTTP://server/hosting/discovery	13
3.1.5.1.1	Discovery.....	13
3.1.5.1.1.1	Request Body.....	13
3.1.5.1.1.2	Response Body.....	14
3.1.5.1.1.2.1	Elements.....	14
3.1.5.1.1.2.1.1	wopi-discovery.....	14
3.1.5.1.1.2.2	Complex Types	14
3.1.5.1.1.2.2.1	ct_wopi-discovery	14
3.1.5.1.1.2.2.2	ct_net-zone	14
3.1.5.1.1.2.2.3	ct_app-name	14
3.1.5.1.1.2.2.4	ct_wopi-action	15
3.1.5.1.1.2.2.5	ct_proof-key	16
3.1.5.1.1.2.3	Simple Types.....	16
3.1.5.1.1.2.3.1	st_wopi-action-values	17
3.1.5.1.1.2.3.2	st_wopi-action-requirements	18
3.1.5.1.1.2.3.3	st_wopi-url-source.....	19
3.1.5.1.1.2.3.4	st_wopi-zone	20
3.1.5.1.1.3	Processing Details.....	21
3.1.6	Timer Events	21
3.1.7	Other Local Events	21
3.2	WOPI Protocol Client Details.....	21
3.2.1	Abstract Data Model	21
3.2.2	Timers	21
3.2.3	Initialization	21

3.2.4	Higher-Layer Triggered Events	21
3.2.5	Message Processing Events and Sequencing Rules	22
3.2.6	Timer Events	22
3.2.7	Other Local Events	22
3.3	WOPI Protocol Server Details	22
3.3.1	Abstract Data Model	22
3.3.2	Timers	22
3.3.3	Initialization	22
3.3.4	Higher-Layer Triggered Events	22
3.3.5	Message Processing Events and Sequencing Rules	23
3.3.5.1	HTTP://server/<...>/wopi*/files/<id>	23
3.3.5.1.1	CheckFileInfo	23
3.3.5.1.1.1	Request Body	24
3.3.5.1.1.2	Response Body	24
3.3.5.1.1.3	Processing Details	28
3.3.5.1.2	PutRelativeFile	28
3.3.5.1.2.1	Request Body	29
3.3.5.1.2.2	Response Body	29
3.3.5.1.2.3	Processing Details	30
3.3.5.1.3	Lock	30
3.3.5.1.3.1	Request Body	31
3.3.5.1.3.2	Response Body	31
3.3.5.1.3.3	Processing Details	31
3.3.5.1.4	Unlock	31
3.3.5.1.4.1	Request Body	32
3.3.5.1.4.2	Response Body	32
3.3.5.1.4.3	Processing Details	32
3.3.5.1.5	RefreshLock	32
3.3.5.1.5.1	Request Body	32
3.3.5.1.5.2	Response Body	33
3.3.5.1.5.3	Processing Details	33
3.3.5.1.6	UnlockAndRelock	33
3.3.5.1.6.1	Request Body	34
3.3.5.1.6.2	Response Body	34
3.3.5.1.6.3	Processing Details	34
3.3.5.1.7	ExecuteCellStorageRequest	34
3.3.5.1.7.1	Request Body	34
3.3.5.1.7.2	Response Body	34
3.3.5.1.7.3	Processing Details	35
3.3.5.1.8	ExecuteCellStorageRelativeRequest	35
3.3.5.1.8.1	Request Body	35
3.3.5.1.8.2	Response Body	35
3.3.5.1.8.3	Processing Details	35
3.3.5.1.9	DeleteFile	36
3.3.5.1.9.1	Request Body	36
3.3.5.1.9.2	Response Body	36
3.3.5.1.9.3	Processing Details	36
3.3.5.1.10	ReadSecureStore	36
3.3.5.1.10.1	Request Body	37
3.3.5.1.10.2	Response Body	37
3.3.5.1.10.3	Processing Details	38
3.3.5.1.11	GetRestrictedLink	38
3.3.5.1.11.1	Request Body	38

3.3.5.1.11.2	Response Body	39
3.3.5.1.11.3	Processing Details	39
3.3.5.1.12	RevokeRestrictedLink	39
3.3.5.1.12.1	Request Body	39
3.3.5.1.12.2	Response Body	39
3.3.5.1.12.3	Processing Details	40
3.3.5.2	HTTP://server/<...>/wopi*/folders/<id>	40
3.3.5.2.1	CheckFolderInfo	40
3.3.5.2.1.1	Request Body	40
3.3.5.2.1.2	Response Body	40
3.3.5.2.1.3	Processing Details	43
3.3.5.3	HTTP://server/<...>/wopi*/files/<id>/contents	43
3.3.5.3.1	GetFile	43
3.3.5.3.1.1	Request Body	44
3.3.5.3.1.2	Response Body	44
3.3.5.3.1.3	Processing Details	44
3.3.5.3.2	PutFile	44
3.3.5.3.2.1	Request Body	45
3.3.5.3.2.2	Response Body	45
3.3.5.3.2.3	Processing Details	45
3.3.5.4	HTTP://server/<...>/wopi*/folders/<id>/children	45
3.3.5.4.1	EnumerateChildren	45
3.3.5.4.1.1	Request Body	46
3.3.5.4.1.2	Response Body	46
3.3.5.4.1.3	Processing Details	46
3.3.6	Timer Events	46
3.3.7	Other Local Events	46
4	Protocol Examples	47
4.1	Accessing Discovery XML	47
4.2	Viewing a Document	48
5	Security	50
5.1	Security Considerations for Implementers	50
5.2	Index of Security Parameters	50
6	Appendix A: Full XML Schema	51
7	Appendix B: Full JSON Schema	53
7.1	CheckFileInfo JSON	53
7.2	CheckFolderInfo JSON	54
7.3	EnumerateChildren JSON	54
7.4	PutRelativeFile JSON	55
8	Appendix C: Product Behavior	56
9	Change Tracking	57
10	Index	59

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 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 defined in [MS-OFCGLOS]:

Augmented Backus-Naur Form (ABNF)
file extension
Hypertext Transfer Protocol (HTTP)
Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)
Information Rights Management (IRM)
Internet Assigned Numbers Authority (IANA)
JavaScript Object Notation (JSON)
Representational State Transfer (REST)
Secure Sockets Layer (SSL)
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 defined 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.

[FIPS180-2] National Institute of Standards and Technology, "Secure Hash Standard", FIPS PUB 180-2, August 2002, <http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf>

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

[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.rfc-editor.org/rfc/rfc2616.txt>

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

[UNICODE] The Unicode Consortium, "The Unicode Consortium Home Page", 2006, <http://www.unicode.org/>

[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/>

1.2.2 Informative References

[MS-OBPAS] Microsoft Corporation, "[Office Broadcast Participant Service](#)".

[MS-OBPRS] Microsoft Corporation, "[Office Broadcast Presentation Service](#)".

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

[MS-SSWPS] Microsoft Corporation, "[Secure Store Web Service Protocol](#)".

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 to present that content to the user as a web page in a browser. The following diagram shows an example of how that might work.

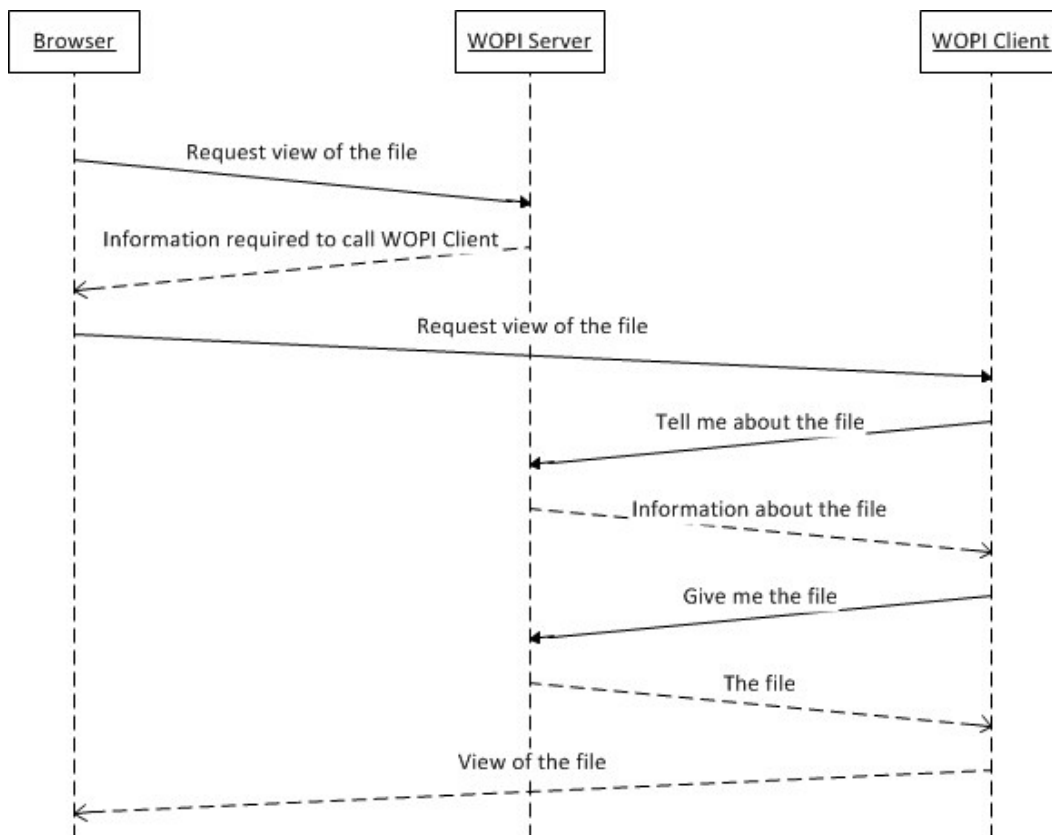


Figure 1: Using WOPI to provide a browser-based viewer for a specific type of file

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 called 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:

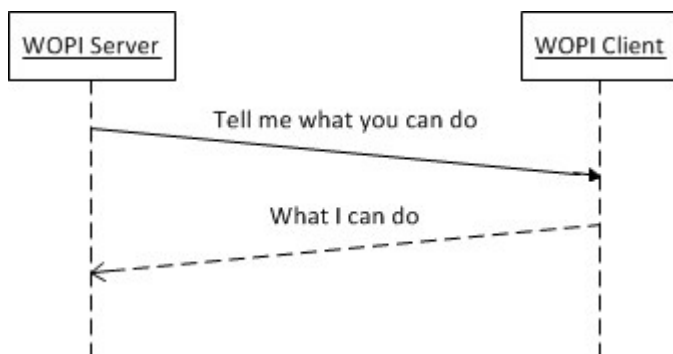


Figure 2: WOPI discovery

1.4 Relationship to Other Protocols

The WOPI Protocol uses **Hypertext Transfer Protocol (HTTP)** as specified in [\[RFC2616\]](#), and **Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)** as specified in [\[RFC2818\]](#).

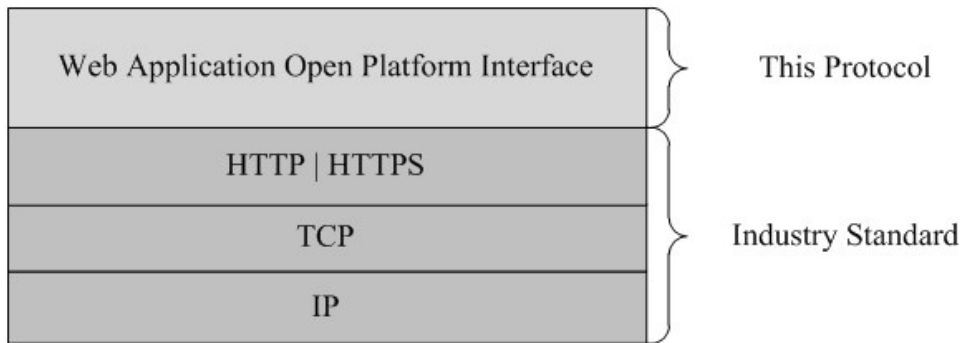


Figure 3: This protocol in relation to other protocols

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" ([\[MS-FSSHHTTP\]](#)). 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 **Internet Assigned Numbers Authority (IANA)** port assignments for HTTP and **Secure Sockets Layer (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 header MUST be included in all WOPI requests.

Header	Description
Authorization	Defined in [RFC2616] . This header MUST have the value "Bearer " + <token> as specified in section 2.2.2 . Note that there must be a space between "Bearer" and <token> .

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 is to 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 has requested 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.
X-WOPI-UsingRestrictedScenario	<p>A restricted scenario is a case where a user is able to operate on a file in a limited way. For example, a user might be allowed to change a file in the course of filling out a form while not having permission to freely edit the file. The value of this header varies depending on the scenario. The value of this header is determined through convention understood by the client and server implementer.</p> <p>The header MUST be present and the value must be correct in cases where the WOPI action (see section 3.1.5.1.1.2.2.4) represents a restricted scenario.</p>
X-WOPI-Proof	<p>A set of data signed using a SHA256 (A 256 bit SHA-2-encoded [FIPS180-2]) encryption algorithm. The value of X-WOPI-Proof is decrypted using the value of oldvalue in ct_wopi-proof-key section 3.1.5.1.1.2.2.5 in Discovery (section 3.1.5.1.1) as the public key.</p> <p>The value of X-WOPI-Proof MUST match the following pattern.</p> <p>4 bytes in network byte order representing the length of the <token> (see section 2.2.2) as an integer + the <token> represented in UTF-8 [UNICODE1] +</p> <p>4 bytes in network byte order representing the length of the URL of the WOPI</p>

Header	Description
	request as an integer + the absolute URL of the WOPI request in uppercase + 4 bytes in network byte order representing the length of X-WOPI-TimeStamp (see this section) + the value of X-WOPI-TimeStamp The intent of passing this header is to allow the WOPI server to validate that the WOPI request originated from the WOPI client that provided the public key in Discovery via ct_wopi-proof-key .
X-WOPI-ProofOld	A set of data signed using a SHA256 (A 256 bit SHA-2-encoded [FIPS180-2]) encryption algorithm. The value of X-WOPI-ProofOld is decrypted using the value of oldvalue or value in ct_wopi-proof-key section 3.1.5.1.1.2.2.5) in Discovery (section 3.1.5.1.1) as the public key. The value of X-WOPI-Proof MUST match the following pattern. 4 bytes in network byte order representing the length of the <token> (see section 2.2.2) as an integer + the <token> represented in UTF-8 [UNICODE] + 4 bytes in network byte order representing the length of the URL of the WOPI request as an integer + the absolute URL of the WOPI request in uppercase + 4 bytes in network byte order representing the length of X-WOPI-TimeStamp (see this section) + the value of X-WOPI-TimeStamp The intent of passing this header is to allow the WOPI server to validate that the WOPI request originated from the WOPI client that provided the public key in Discovery via ct_wopi-proof-key .
X-WOPI-TimeStamp	A 64-bit integer that represents the number of 100-nanosecond intervals that have elapsed between 12:00:00 midnight, January 1, 0001 and the time of the request. The WOPI client MUST include this HTTP header if it includes X-WOPI-Proof or X-WOPI-ProofOld.

The following HTTP headers can be included with all WOPI responses.

Header	Description
X-WOPI-ServerVersion	A string specifying the version of the WOPI server and MUST be included with all WOPI responses. There is no standard for how this string is to 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 and MUST be included with all WOPI responses, 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. It is included in a WOPI response if the header X-WOPI-PerfTraceRequest in the request is present and equal to "true".
X-WOPI-ServerError	A string indicating that an error occurred while processing the WOPI request, which is included in a WOPI response if the status code is 500. 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>	<p>A string used by the WOPI server to determine the identity and permissions of the issuer of the WOPI request. The token is generated by the WOPI server and learned by the WOPI client before the first WOPI request. The WOPI server passes the token to the WOPI client in the initiating request described in section 3.1.5.1.1.2.3.3.</p> <p>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 [3.1.5.1.1.2.3.3](#)).

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

URI Parameter	Description
<sc>	<p>The sc is a string that the WOPI server MAY include with a WOPI request. The WOPI client MAY in turn return the value of sc as the value of the X-WOPI-SessionContext HTTP header when making the CheckFileInfo (section 3.3.5.1.1) and CheckFolderInfo (section 3.3.5.2.1) WOPI requests.</p>

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 call these abilities through URIs. This XML is available through a predetermined URI that MUST be provided by a WOPI client.

3.1.1 Abstract Data Model

None.

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Higher-Layer Triggered Events

None.

3.1.5 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.

3.1.5.1 HTTP://server/hosting/discovery

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

3.1.5.1.1 Discovery

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

3.1.5.1.1.1 Request Body

None.

3.1.5.1.1.2 Response Body

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

3.1.5.1.1.2.1 Elements

This section describes the elements of the WOPI Discovery XML.

3.1.5.1.1.2.1.1 wopi-discovery

A **ct_wopi-discovery** element (section [3.1.5.1.1.2.2.1](#)) 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" minOccurs="1" maxOccurs="1" type="ct_wopi-discovery"/>
```

3.1.5.1.1.2.2 Complex Types

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

3.1.5.1.1.2.2.1 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:element name="proof-key" minOccurs="0" maxOccurs="1" type="ct_proof-key"/>
  </xs:sequence>
</xs:complexType>
```

3.1.5.1.1.2.2.2 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 [3.1.5.1.1.2.2.3](#)) it contains.

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

3.1.5.1.1.2.2.3 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).

Attributes:

name: A **string** attribute that specifies the name of the application.

favIconUrl: A **string** attribute that specifies the URL of an image resource that a WOPI server MAY use as the Favorites Icon for a page showing the output of a WOPI client.

checkLicense: A **Boolean** attribute that specifies that a WOPI server SHOULD enforce license restrictions for file types within the **ct_app-name** block.

```
<xs:complexType name="ct_app-name">
  <xs:sequence>
    <xs:element name="action" minOccurs="1" maxOccurs="unbounded"
      type="ct_wopi-action"/>
  </xs:sequence>
  <xs:attribute name="name" type="xs:string" use="required"/>
  <xs:attribute name="favIconUrl" type="xs:string" use="optional"/>
  <xs:attribute name="checkLicense" type="xs:boolean" use="optional"
    default="false"/>
</xs:complexType>
```

3.1.5.1.1.2.2.4 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 [3.1.5.1.1.2.3.1](#)) that specifies the name of the action.

default: A **Boolean** attribute that specifies whether the WOPI server is to use this action as the default action for this file type.

requires: A **st_wopi-action-requirements** attribute (section [3.1.5.1.1.2.3.2](#)) that specifies the required capabilities of the WOPI server to be able to use this action.

urlsrc: A **st_wopi-url-source** attribute (section [3.1.5.1.1.2.3.3](#)) that specifies the URI that the WOPI server can use to call 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 [3.1.5.1.1.2.3.1](#)) 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 [3.1.5.1.1.2.3.1](#)) 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.

useParent: A **Boolean** attribute that specifies that the WOPI server MUST call the WOPI action on the parent folder of the file.

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:attribute name="useParent" type="xs:boolean" default="false"/>
</xs:complexType>
```

3.1.5.1.1.2.2.5 ct_proof-key

The **ct_proof-key** complex type specifies two public keys that a WOPI server uses to decrypt the values passed in the **X-WOPI-Proof** and **X-WOPI-ProofOld** HTTP headers (section [2.2.1](#)). Note that the public keys passed in this complex type decrypt either of the HTTP header values.

If you include proof keys in discovery, you SHOULD include **X-WOPI-Proof** and **X-WOPI-ProofOld** HTTP headers (section [2.2.1](#)) in all WOPI requests.

Attributes:

oldvalue: A public key that a WOPI server uses to decrypt the value of the **X-WOPI-Proof** or **X-WOPI-ProofOld** HTTP headers (section [2.2.1](#)).

value: A public key that a WOPI server uses to decrypt the value of **X-WOPI-ProofOld** HTTP headers (section [2.2.1](#)).

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

```
<xs:complexType name="ct_proof-key">
  <xs:attribute name="oldvalue" type="xs:string" use="required"/>
  <xs:attribute name="value" type="xs:string" use="required"/>
</xs:complexType>
```

3.1.5.1.1.2.3 Simple Types

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

3.1.5.1.1.2.3.1 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.

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.

presentservice: An action that indicates the location of a broadcast endpoint for broadcast presenters. Interaction with the endpoint is described in [\[MS-OBPRS\]](#).

attendservice: An action that indicates the location of a broadcast endpoint for broadcast attendees. Interaction with the endpoint is described in [\[MS-OBPAS\]](#).

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 [3.1.5.1.1.2.2.4](#)) indicate the type of file that will result from invoking the **editnew** action.

imagepreview: An action that indicates that the WOPI client supports providing an image preview of the file type.

interactivepreview: An action that indicates that the WOPI client supports providing an interactive preview of the file type. For example, a smaller version of a presentation that a user can still advance from slide to slide.

formssubmit: An action that indicates that the WOPI client supports accepting changes to the file type via a form-style interface. For example, a user might be able to use this action to change the content of a workbook even if they did not have permission to use the **edit** action.

formedit: An action that indicates that the WOPI client supports editing the file type in a mode better suited to working with files that have been used to collect form data via the **formssubmit** action.

rest: An action that indicates that the WOPI client supports interacting with the file type via additional URL parameters that are specific to the file type in question.

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="presentservice" />
    <xs:enumeration value="attend" />
    <xs:enumeration value="attendservice" />
    <xs:enumeration value="editnew" />
    <xs:enumeration value="imagepreview" />
    <xs:enumeration value="interactivepreview" />
    <xs:enumeration value="formsubmit" />
    <xs:enumeration value="formedit" />
    <xs:enumeration value="rest" />
  </xs:restriction>
</xs:simpleType>
```

3.1.5.1.1.2.3.2 st_wopi-action-requirements

The **st_wopi-action-requirements** simple type specifies the capabilities that a WOPI server MUST support to call 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 described 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](#)), **CheckFolderInfo** (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>

3.1.5.1.1.2.3.3 st_wopi-url-source

The **st_wopi-url-source** simple type specifies the URI that is used to call 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.

A WOPI server MAY also include an **access_token_TTL** parameter.

access_token: A parameter that is the **<token>** (see section [2.2.2](#)).

The URI parameter MUST have the form "access_token=**<token>**".

Because the **token** represents user credentials it needs to be protected. To protect the **token** the WOPI client MUST accept the **token** as part of the body of an HTTP POST. The format of the POST body MUST be "access_token=**<token>**"

The WOPI client SHOULD NOT include the URI parameter **access_token** and the POST body content in the same request.

access_token_TTL: A 64-bit **integer** that represents the number of 100-nanosecond intervals until the **<token>** will no longer be valid.

The WOPI client MUST accept the **access_token_TTL** as part of the body of an HTTP POST. The format of the POST body MUST be "access_token_TTL=[a 64-bit integer as described above]"

The WOPI client SHOULD NOT include the URI parameter **access_token_TTL** and the POST body content in the same request.

WOPISrc: A URI that conforms to the pattern **http://server/<...>/wopi*/files/<id>** (see section [3.3.5.1](#)). The URI MUST be encoded to a URL string.

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 corresponding string in place of the placeholder value. Each optional parameter conforms to the pattern **<name=PLACEHOLDER_VALUE[&]>**. The WOPI server MAY omit the optional parameter, but MUST only process it if the placeholder value is understood and the corresponding 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 can 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_BROADCAST: Indicates that the WOPI server MAY include the value "true" to load a view of the document that does not create or join a broadcast session. This view looks and behaves like a regular broadcast frame.

RECORDING: Indicates that the WOPI server MAY include the value "true" to load the file type with a minimal user interface.

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

Additionally, WOPI servers MUST append to the **WOPISrc** any query string parameters included with the WOPI server URL that begin with the characters: "wd". That is, if the WOPI server receives a request from a browser that indicates to the WOPI server that the browser wishes to call a WOPI client action, then all query string parameters included on the request to the WOPI server MUST be passed through on the **WOPISrc**.

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

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.

20 / 60

```

<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>

```

3.1.5.1.1.3 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.

3.1.6 Timer Events

None.

3.1.7 Other Local Events

None.

3.2 WOPI Protocol Client Details

The WOPI client issues requests to WOPI servers 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](#)).

3.2.1 Abstract Data Model

None.

3.2.2 Timers

None.

3.2.3 Initialization

None.

3.2.4 Higher-Layer Triggered Events

None.

3.2.5 Message Processing Events and Sequencing Rules

None.

3.2.6 Timer Events

None.

3.2.7 Other Local Events

None.

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 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 [\[MS-FSSHTTP\]](#) 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*/folders/<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 **Augmented Backus-Naur Form (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.

The following operations are allowed to be performed on this resource.

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 following HTTP header MAY be included with this request.

Header	Description
X-WOPI-SessionContext	The value of the <sc> URI parameter (section 2.2.2).

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 **JavaScript Object Notation (JSON)** (as specified in [RFC4627](#)) with the following parameters:

JSON:

```
{
  "AllowExternalMarketplace":{"type":"bool","default":false,"optional":true},
  "BaseFileName":{"type":"string","optional":false},
  "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},
  "CloseButtonClosesWindow":{"type":"bool","default":false,"optional":true},
  "CloseUrl":{"type":"string","default":"","optional":true},
  "DisableBrowserCachingOfUserContent":{"type":"bool","default":false,"optional":true},
  "DisablePrint":{"type":"bool","default":false,"optional":true},
  "DisableTranslation":{"type":"bool","default":false,"optional":true},
  "DownloadUrl":{"type":"string","default":"","optional":true},
  "FileSharingUrl":{"type":"string","default":"","optional":true},
  "FileUrl":{"type":"string","default":"","optional":true},
  "HostAuthenticationId":{"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},
  "HostNotes":{"type":"string","default":"","optional":true},
  "HostRestUrl":{"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},
    "PresenceProvider":{"type":"string","default":"","optional":true},
    "PresenceUserId":{"type":"string","default":"","optional":true},
    "PrivacyUrl":{"type":"string","default":"","optional":true},
    "ProtectInClient":{"type":"bool","default":false,"optional":true},
    "ReadOnly":{"type":"bool","default":false,"optional":true},
    "RestrictedWebViewOnly":{"type":"bool","default":false,"optional":true},
    "SHA256":{"type":"string","optional":true},
    "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},
    "SupportsFolders":{"type":"bool","default":false,"optional":true},
    "SupportsLocks":{"type":"bool","default":false,"optional":true},
    "SupportsScenarioLinks":{"type":"bool","default":false,"optional":true},
    "SupportsSecureStore":{"type":"bool","default":false,"optional":true},
    "SupportsUpdate":{"type":"bool","default":false,"optional":true},
    "TenantId":{"type":"string","default":"","optional":true},
    "TermsOfUseUrl":{"type":"string","default":"","optional":true},
    "TimeZone":{"type":"string","default":"","optional":true},
    "UserCanAttend":{"type":"bool","default":false,"optional":true},
    "UserCanNotWriteRelative":{"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}
    "WebEditingDisabled":{"type":"bool","default":false,"optional":true},
}

```

AllowExternalMarketplace: A **Boolean** value that indicates the WOPI client MAY allow connections to external services referenced in the file (for example, a marketplace of embeddable JavaScript apps). If this value is false, then the WOPI client MUST NOT allow such connections.

BaseFileName: The name of the file without the path. Used for display in user interface (UI), and determining the extension of the file.

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

BreadcrumbBrandUrl: 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.

BreadcrumbDocUrl: 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.

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

ClientUrl: A user-accessible URI directly to the file intended for opening the file through a client. Can be 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.

CloseButtonClosesWindow: A **Boolean** value that indicates that the WOPI client SHOULD close the browser window containing the output of the WOPI client when the user calls the close UI.

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.

DisableBrowserCachingOfUserContent: A **Boolean** value that indicates that the WOPI client MUST disable caching of file contents in the browser cache.

DisablePrint: A **Boolean** value that indicates that the WOPI client MUST disable any print functionality under its control.

DisableTranslation: A **Boolean** value that indicates that the WOPI client MUST not permit the use of machine translation functionality that is exposed by the WOPI client.

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

FileSharingUrl: A URI to a location that allows the user to share the file.

FileUrl: 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](#)).

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

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

HostEmbeddedEditUrl: 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.

HostEmbeddedViewUrl: 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.

HostNotes: A **string** that is used by the WOPI server to pass arbitrary information to the WOPI client. The WOPI client MAY ignore this string if it does not recognize the contents. A WOPI server MUST NOT require that a WOPI client understand the contents of this string to operate.

HostRestUrl: A URI that is the base URI for **REST** operations for the file.

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 **Information Rights Management (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.

PresenceProvider: A **string** that identifies the provider of information that a WOPI client MAY use to discover information about the user's online status (for example, whether a user is available via instant messenger). A WOPI client requires knowledge of specific presence providers to be able to take advantage of this value.

PresenceUserId: A **string** that identifies the user in the context of the **PresenceProvider**.

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.

RestrictedWebViewOnly: A **Boolean** value that indicates that the WOPI client MUST NOT allow the user to download the file or open the file in a separate application.

SHA256: If it is present and not empty, it is 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. It MUST be false.

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.

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.

SupportsScenarioLinks: A **Boolean** value that indicates that the WOPI server supports scenarios where users can operate on files in limited ways via restricted URLs.

SupportsSecureStore: A **Boolean** value that indicates that the WOPI server supports calls to a secure data store utilizing credentials stored in the 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.

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

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

TimeZone: A **string** that is used to pass time zone information to a WOPI client in the format chosen by 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 can be broadcast by a presenter to many attendees.

UserCanNotWriteRelative: A **Boolean** value that indicates the user does not have sufficient permissions to create new files on the WOPI server.

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 can be broadcast by a presenter to many attendees.

UserCanWrite: A **Boolean** value that 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 be configured to use a placeholder **string** in some scenarios, or to 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.

WebEditingDisabled: A **Boolean** value that indicates that the WOPI client MUST NOT allow the user to use the WOPI client's editing functionality to operate on the file. This does not mean that the user doesn't have rights to edit the 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	A string that is a file extension or a full file name.

Request header	Usage	Value
	name as needed 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.	
X-WOPI-RelativeTarget	A string that specifies a file name. The WOPI server MUST NOT modify the name 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	An 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
413	File is too large. The maximum size is implementation-specific.
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 if the status code is 200.

Response 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

Status code	Description
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

Status code	Description
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	Token is invalid
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-Lock	A string provided by the WOPI client that the WOPI server MUST use to identify the lock on the file. Required.	A string .
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 contains the following HTTP headers if the status code is 200.

Response 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

3.3.5.1.6.1 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	Token is invalid
404	File unknown/User unauthorized
500	Server error
501	Unsupported

3.3.5.1.7.1 Request Body

Content that conforms to the **Body** element specified in [\[MS-FSSHTTP\]](#) section 2.2.2.1.

3.3.5.1.7.2 Response Body

Content that conforms to the **Body** element specified in [\[MS-FSSHTTP\]](#) section 2.2.2.2.

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 the 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. The 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	Token is invalid
404	File unknown/User unauthorized
500	Server error
501	Unsupported

3.3.5.1.8.1 Request Body

Content that conforms to the **Body** element specified in [\[MS-FSSHTTP\]](#) section 2.2.2.1.

3.3.5.1.8.2 Response Body

Content that conforms to the **Body** element specified in [\[MS-FSSHTTP\]](#) section 2.2.2.2.

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	Token is invalid
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.

3.3.5.1.10 ReadSecureStore

Access the WOPI server's implementation of a secure store as specified in [\[MS-SSWPS\]<1>](#). Note that the WOPI client is not required to implement [MS-SSWPS]. However, an understanding of the protocol and the acquisition of information that is part of the protocol is required to use ReadSecureStore.

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 "READ_SECURE_STORE".
X-WOPI-ApplicationId	[MS-SSWPS] section 3.1.4.2 describes how a WOPI client obtains the application ID required to access a specific secure store. This value is equal to the appId element of the application element nested under the CreateApplication element. This value is required.	The value of the Application ID.

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
500	Server error
501	Unsupported

3.3.5.1.10.1 Request Body

None.

3.3.5.1.10.2 Response Body

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

JSON:

```
{
  "UserName":{"type":"string","optional":false},
  "Password":{"type":"string","default":"","optional":false},
  "IsWindowsCredentials":{"type":"bool","default":"false","optional":true},
  "IsGroup":{"type":"bool","default":"false","optional":true},
}
```

UserName: The **UserName** (see [\[MS-SSWPS\]](#) section 2.2.5.4).

Password: The **Password** (see [\[MS-SSWPS\]](#) section 2.2.5.4).

IsWindowsCredentials: A **Boolean** value that specifies that **UserName** corresponds to **WindowsUserName** and **Password** corresponds to **WindowsPassword** (see [\[MS-SSWPS\]](#) section 2.2.5.4).

IsGroup: A **Boolean** value that specifies that the secure store application is a **Group** (see [\[MS-SSWPS\]](#) section 2.2.5.5).

3.3.5.1.10.3 Processing Details

The WOPI client MAY use the credentials returned by the WOPI server to access other server types on behalf of the user.

3.3.5.1.11 GetRestrictedLink

Gets a link to a file through which a user is able to operate on a file in a limited way. For example, a user might be allowed to change a file in the course of filling out a form while not having permission to freely edit the 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 "GET_RESTRICTED_LINK".
X-WOPI-RestrictedLink	A string specifying the type of restricted link being requested by the WOPI client. The valid values of this string are determined through convention. The only valid value currently is "FORMS". Required	A string that is a valid value for a restricted link type. For example, "FORMS".

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

Request header	Usage	Value
X-WOPI-RestrictedUseLink	A string specifying the URL that the WOPI client requested. For example, the URL that allows the user to fill out a form. Required.	A string that is a valid URL.

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
500	Server error
501	Unsupported

3.3.5.1.11.1 Request Body

None.

3.3.5.1.11.2 Response Body

None.

3.3.5.1.11.3 Processing Details

This operation is directly related to the use of the **X-WOPI-UsingRestrictedScenario** HTTP header (section [2.2.1](#)).

3.3.5.1.12 RevokeRestrictedLink

Revokes all links to a file through which a number of users are able to operate on a file in a limited way. For example, users might be allowed to change a file in the course of filling out a form while not having permission to freely edit the file. This operation will render all links of that type invalid.

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 "REVOKE_RESTRICTED_LINK".
X-WOPI-RestrictedLink	A string specifying the type of restricted link being revoked by the WOPI client. The valid values of this string are determined through convention. The only valid value currently is "FORMS".	A string that is a valid value for a restricted link type. For example, "FORMS".

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
500	Server error
501	Unsupported

3.3.5.1.12.1 Request Body

None.

3.3.5.1.12.2 Response Body

None.

3.3.5.1.12.3 Processing Details

This operation is directly related to the use of the **X-WOPI-UsingRestrictedScenario** HTTP header (section [2.2.1](#)).

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.

The following operation is allowed to be performed on this resource.

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>

The following HTTP header MAY be included with this request.

Header	Description
X-WOPI-SessionContext	The value of the <sc> URI parameter (section 2.2.2).

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},
  "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},
"CloseButtonClosesWindow":{"type":"bool","default":false,"optional":true},
"CloseUrl":{"type":"string","default":"","optional":true},
"FileSharingUrl":{"type":"string","default":"","optional":true},
"HostAuthenticationId":{"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},
"PresenceProvider":{"type":"string","default":"","optional":true},
"PresenceUserId":{"type":"string","default":"","optional":true},
"PrivacyUrl":{"type":"string","default":"","optional":true},
"SignoutUrl":{"type":"string","default":"","optional":true},
"SupportsSecureStore":{"type":"bool","default":false,"optional":true},
"TenantId":{"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},
"WebEditingDisabled":{"type":"bool","default":false,"optional":true},
}

```

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

BreadcrumbBrandIconUrl: 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.

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

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

BreadcrumbDocUrl: A URI to a web page that the WOPI client MAY navigate to when the user clicks on the 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.

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

ClientUrl: A user-accessible URI directly to the folder intended for opening the file through a client. Can be 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.

CloseButtonClosesWindow: A **Boolean** value that indicates that the WOPI client SHOULD close the browser window containing the output of the WOPI client when the user calls the close UI.

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.

FileSharingURL: A URI to a location that allows the user to share the file.

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

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

HostEmbeddedEditUrl: 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.

HostEmbeddedViewUrl: 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.

PresenceProvider: A **string** that identifies the provider of information that a WOPI client MAY use to discover information about the user's online status (for example, whether a user is available via instant messenger). A WOPI client requires knowledge of specific presence providers to be able to take advantage of this value.

PresenceUserId: A **string** that identifies the user in the context of the **PresenceProvider**.

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.

SupportsSecureStore: A **Boolean** value that indicates that the WOPI server supports calls to a secure data store utilizing credentials stored in the file.

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

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 be configured to use a placeholder **string** in some scenarios, or to show no name at all.

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

WebEditingDisabled: A **Boolean** value that indicates that the WOPI client MUST NOT allow the user to use the WOPI client's editing functionality to operate on the file. This does not mean that the user doesn't have rights to edit the file.

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.

The following operations are allowed to be performed on this resource.

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. The WOPI server will use the maximum value of a 4-byte integer if this value is not set in the request.	An integer .

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.
412	File is larger than X-WOPI-MaxExpectedSize.

Status code	Description
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 the requested operation from the WOPI server. Required.	The string "PUT".
X-WOPI-Size	An integer specifying the size of the request body. Optional. The WOPI server will read the entire request body if this value is not set in the request.	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	Token is invalid
404	File unknown/User unauthorized
409	Lock mismatch/Locked by another interface
413	File is too large. The maximum file size is implementation-specific.
500	Server error

Status code	Description
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*/folders/<id>/children

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

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

The following operation is allowed to be performed on this resource.

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*/folders/<id>/children?access_token=<token>

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

Status code	Description
200	Success.
401	Token is invalid.

Status code	Description
404	Folder 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 of 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.

Request from WOPI server:

```
GET http://WOPIClient/hosting/discovery HTTP/1.1
```

Response from WOPI client:

```
<?xml version="1.0" encoding="utf-8"?>
<wopi-discovery>
<net-zone name="internal-http">
  <app name="Word" favIconUrl="http://WOPIClient/wv/resources/1033/FavIcon_Word.ico"
  checkLicense="true">
    .....
    <action name="view" ext="docx" default="true"
    urlsrc="http://WOPIClient/wv/wordviewerframe.aspx?&lt;ui=UI_LLCC&amp;&gt;&lt;rs=DC_LLCC&amp;&
    gt;&lt;showpagestats=PERFSTATS&amp;&gt;" />
    .....
    <action name="edit" ext="docx" requires="locks,cobalt,update"
    urlsrc="http://WOPIClient/we/wordeditorframe.aspx?&lt;ui=UI_LLCC&amp;&gt;&lt;rs=DC_LLCC&amp;&
    gt;&lt;showpagestats=PERFSTATS&amp;&gt;" />
    .....
    <action name="editnew" ext="docx" requires="locks,cobalt,update"
    urlsrc="http://WOPIClient/we/wordeditorframe.aspx?new=1&amp;&lt;ui=UI_LLCC&amp;&gt;&lt;rs=DC_
    LLCC&amp;&gt;&lt;showpagestats=PERFSTATS&amp;&gt;" />
    .....
    <action name="imagepreview" ext="docx"
    urlsrc="http://WOPIClient/wv/WordPreviewHandler.ashx?&lt;ui=UI_LLCC&amp;&gt;&lt;rs=DC_LLCC&am
    p;&gt;&lt;showpagestats=PERFSTATS&amp;&gt;" />
    .....
    <action name="interactivepreview" ext="docx"
    urlsrc="http://WOPIClient/wv/wordviewerframe.aspx?embed=1&amp;&lt;ui=UI_LLCC&amp;&gt;&lt;rs=D
    C_LLCC&amp;&gt;&lt;showpagestats=PERFSTATS&amp;&gt;" />
    .....
    <action name="mobileView" ext="docx"
    urlsrc="http://WOPIClient/wv/wordviewerframe.aspx?&lt;ui=UI_LLCC&amp;&gt;&lt;rs=DC_LLCC&amp;&
    gt;&lt;showpagestats=PERFSTATS&amp;&gt;" />
    .....
    <action name="embedview" ext="docx"
    urlsrc="http://WOPIClient/wv/wordviewerframe.aspx?embed=1&amp;&lt;ui=UI_LLCC&amp;&gt;&lt;rs=D
    C_LLCC&amp;&gt;&lt;showpagestats=PERFSTATS&amp;&gt;" />
```

```

.....
    <action name="attend" ext="docx"
    urlsrc="http://WOPIClient/m/ParticipantFrame.aspx?a=1&&lt;e=EMBEDDED&&lt;fs=FULLS
    CREEN&&lt;rec=RECORDING&&lt;thm=THEME_ID&&lt;na=DISABLE_ASYNC&&lt;
    ;&&lt;vp=DISABLE_BROADCAST&&lt;ui=UI_LLCC&&lt;rs=DC_LLCC&&lt;showpages
    tats=PERFSTATS&&lt;" />
    .....
    <action name="convert" ext="doc" targetext="docx"
    urlsrc="http://WOPIClient/wv/WordConvert.aspx?&&lt;ui=UI_LLCC&&lt;rs=DC_LLCC&&lt;
    ;&&lt;showpagestats=PERFSTATS&&lt;" />
    </app>
</net-zone>
<proof-key oldvalue=""
value="BgIAAAckAABSU0ExAagAAAEAAQDtu862xwZwZukFLJsbuxPc7COBv4VxNzpm1jceQdtDfedUtC1GPa9omFkoyw
171X0HFMKRKCVCESiLo45YK5FfghlOoPG+uVzR11+s8bXznSEValBFYgkHWq47e1afvhEMwonqiYaa99i1o6TQyAjYUylj
xgl6S75Uvp3OxXjYiIwh8MfhcW1sPg3FEfx3iQ9s1sYb7Bpu75EQcEEz0p8VVgDRGMRqs4cEZIBVUNTvJVpBwXnipDfxv
M5uWaz10nQ14+cQU6kUrs1fMl/CxIdFTOoAg5AkI8jez7c5sl/s6W1oKVtbsuyEMcSSJqU7bNkmfj+593Sr1+rTNmENy7
jDH" />
</wopi-discovery>

```

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 uses **GetFile** operation to get this file and then a view of this file inside the iframe created by the WOPI server. The WOPI server returns the file to user.

When implementing a WOPI client page it is important to set up the page such that it is not cached. Caching the page may introduce undesirable behaviors.

GetFile request from WOPI client:

```

GET
http://WOPIClient/wv/ResReader.ashx?n=p_1_10.xml&WOPIsrc=http%3A%2F%2FWOPIServer%2F%5Fvti%5Fb
in%2Fwopi%2Eashx%2Ffiles%2Fcc8e6af1e0554661b71faeaa0d45fdd7&access_token=eyJ0eXAiOiJKV1QiLCJh
bGciOiJSUzI1NiIsIngldCI6ImlpDdVJLUG1HSU1DNThhZW9QOS1EblY1M0VJOCJ9%2EeyJhdWQiOiJ3b3BpL3N1dG8xNT
MxQDAxNjVmYjU3LWQzMTYtNDQxOS04ODE5LWTE5ZDRiMzdiY2JlMSIsIm1lzc3I6IjAwMDAwMDAaLTAwMDAtMGZmMS1jZTA
wLTAwMDAwMDAwMDAwMEAwMTY1ZmI1Ny1kMzE2LTQ0MTktODgxOS0xOWQ0YjM3YmNiZTEiLCJuYmYiOiIxMzQzODcyOTg5
IiwiaXhwaWoiMTM0MzkwODk4OSIsIm5hbWVpZCI6IjAjLnd8ZmFyZWZzdFxccGV0dGVzdCI6Im5paSI6Im1pY3Jvc29md
C5zaGFyZXBvaW50IiwiaXN1c2VyIjoiaHJlZSIsImNhY2hla2V5IjoiaMkud3xzLTETNS0yMS0yMTQ2NzczMDg1LTkwMz
M2MzI4NS03MTkzNDQ3MDctNjkyMTU2IiwiaXNsb29wYmFjayI6IjRydwUuILCJhcHBjdHgiOiJjYzh1NmFmMWUwNTU0NjY
xYjcxZmF1WEwZDQ1ZmRkNzt2SUtMTktTa0tBU21nYTNJR1ZKU0w2Q21wTUU9O0RlZmF1bHQ7OzdGRkZGRkZGRkZGRkZG
RkY7VHJlZSJ9%2EFuxDXJ4Wk1rTrSSDuFbbM%5F61BGW%2DxBz5JBHdpG64GiLTSe9VRhxOaS3%2DbJDuDHDJaqzsbSOa6
UoVnVKtGe9Mift2fMuF4nxcXoDZx3QxcJncs%5FTbcUKQLDN6Z4ZHIZJeM455ms53Mfs1JoIwwmvWaTm76YJYwbawEYII
xI7dPSbxUaN1102EVmmT61YSZnk0xaUaoFXPlYHgbSjFgqENfSCC1M%5Fc%2Da19DTPe9www1hRwWWep5eDS1bwRYIJI
ETXBVF6vtmLWBk7rT53S%5FesQF4VV0pPjMmULWZ0yEarXOa8bKslJ%5F000U%5F7LpmEJK7GvQ5A3yPP40gfpLrKi6KT
CWQ&access_token_ttl=1343908989210&z=%2522%257BCC8E6AF1%252DE055%252D4661%252DB71F%252DAEAA0D
45FDD7%257D%252C4%2522&v=00000000-0000-0000-0000-000000000602&usid=4cd12455-fd24-4bf9-a5a6-
647994930a60 HTTP/1.1

```


GetFile response from WOPI server:

```
<?xml version="1.0"?>
<Pages>
  <Page id="1">
    <Header>
      <P storyId="2" id="6C2D515C">
        <CN/>
        <T id="0" l="96.1" t="49.4" w="9" h="15" b="1" cw=",7;"> </T>
      </P>
    </Header>
    <Footer>
      <P storyId="3" id="4F36214B">
        <CN/>
        <T id="1" l="96.1" t="992.2" w="9" h="15" b="1" cw=",7;"> </T>
      </P>
    </Footer>
    <P storyId="1" id="4F75DE7D">
      <T id="2" l="96.1" t="97.4" w="33" h="15" b="1" cw="410,4;286,6;400,5;410,3;,7;">test
    </T>
  </P>
</Page>
</Pages>
```

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="presentservice" />
    <xs:enumeration value="attend" />
    <xs:enumeration value="attendservice" />
    <xs:enumeration value="editnew" />
    <xs:enumeration value="imagepreview" />
    <xs:enumeration value="interactivepreview" />
    <xs:enumeration value="formsubmit" />
    <xs:enumeration value="formedit" />
    <xs:enumeration value="rest" />
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="st_wopi-action-requirements">
  <xs:restriction base="xs:string">
    <xs:pattern
value="/^((update|locks|cobalt|containers)\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" 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:element name="proof-key" minOccurs="0" maxOccurs="1" type="ct_proof-key"/>
    </xs:sequence>
  </xs:complexType>

  <xs:element name="wopi-discovery" minOccurs="1" maxOccurs="1" type="ct_wopi-discovery"/>

  <xs:complexType name="ct_app-name">
    <xs:sequence>
      <xs:element name="action" minOccurs="1" maxOccurs="unbounded"
        type="ct_wopi-action"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="favIconUrl" type="xs:string" use="optional"/>
    <xs:attribute name="checkLicense" type="xs:boolean" use="optional"
      default="false"/>
  </xs:complexType>

  <xs:complexType name="ct_proof-key">
    <xs:attribute name="oldvalue" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
  </xs:complexType>

</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
PutRelativeFile JSON	7.4

7.1 CheckFileInfo JSON

```
{
  "AllowExternalMarketplace":{"type":"bool","default":false,"optional":true},
  "BaseFileName":{"type":"string","optional":false},
  "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},
  "CloseButtonClosesWindow":{"type":"bool","default":false,"optional":true},
  "CloseUrl":{"type":"string","default":"","optional":true},
  "DisableBrowserCachingOfUserContent":{"type":"bool","default":false,"optional":true},
  "DisablePrint":{"type":"bool","default":false,"optional":true},
  "DisableTranslation":{"type":"bool","default":false,"optional":true},
  "DownloadUrl":{"type":"string","default":"","optional":true},
  "FileUrl":{"type":"string","default":"","optional":true},
  "HostAuthenticationId":{"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},
  "HostNotes":{"type":"string","default":"","optional":true},
  "HostRestUrl":{"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},
  "PresenceProvider":{"type":"string","default":"","optional":true},
  "PresenceUserId":{"type":"string","default":"","optional":true},
  "PrivacyUrl":{"type":"string","default":"","optional":true},
  "ProtectInClient":{"type":"bool","default":false,"optional":true},
  "ReadOnly":{"type":"bool","default":false,"optional":true},
  "RestrictedWebViewOnly":{"type":"bool","default":false,"optional":true},
  "SHA256":{"type":"string","optional":true},
  "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},
  "SupportsFolders":{"type":"bool","default":false,"optional":true},
```

```

"SupportsLocks":{"type":"bool","default":false,"optional":true},
"SupportsScenarioLinks":{"type":"bool","default":false,"optional":true},
"SupportsSecureStore":{"type":"bool","default":false,"optional":true},
"SupportsUpdate":{"type":"bool","default":false,"optional":true},
"TenantId":{"type":"string","default":"","optional":true},
"TermsOfUseUrl":{"type":"string","default":"","optional":true},
"TimeZone":{"type":"string","default":"","optional":true},
"UserCanAttend":{"type":"bool","default":false,"optional":true},
"UserCanNotWriteRelative":{"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}
"WebEditingDisabled":{"type":"bool","default":false,"optional":true},
}

```

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},
  "CloseButtonClosesWindow":{"type":"bool","default":false,"optional":true},
  "CloseUrl":{"type":"string","default":"","optional":true},
  "HostAuthenticationId":{"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},
  "PresenceProvider":{"type":"string","default":"","optional":true},
  "PresenceUserId":{"type":"string","default":"","optional":true},
  "PrivacyUrl":{"type":"string","default":"","optional":true},
  "SignoutUrl":{"type":"string","default":"","optional":true},
  "SupportsSecureStore":{"type":"bool","default":false,"optional":true},
  "TenantId":{"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},
  "WebEditingDisabled":{"type":"bool","default":false,"optional":true},
}

```

7.3 EnumerateChildren JSON

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

7.4 PutRelativeFile JSON

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

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 2013
- Microsoft Exchange Server 2013
- Microsoft Office 2013
- Microsoft SharePoint Server 2013
- Microsoft SharePoint Foundation 2013

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 3.3.5.1.10:](#) SharePoint Foundation 2013 does not support the **ReadSecureStore** operation.

9 Change Tracking

This section identifies changes that were made to the [MS-WOPI] protocol document between the July 2014 and October 2014 releases. 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
3.3.5.4.1 EnumerateChildren	Updated the description of the status code 404.	N	Content updated.

10 Index

A

Abstract data model
 [WOPI protocol server details](#) 22
[Accessing discovery XML example](#) 47
[Applicability](#) 9

C

[Capability negotiation](#) 9
[Change tracking](#) 57
[CheckFileInfo JSON](#) 53
[CheckFolderInfo JSON](#) 54
[Common URI parameters](#) 11
[Custom HTTP headers](#) 10

D

Data model – abstract
 [WOPI protocol server details](#) 22

E

[EnumerateChildren JSON](#) 54
Examples
 [accessing discovery XML](#) 47
 [overview](#) 47
 [viewing a document](#) 48

F

[Fields - vendor-extensible](#) 9
[Full JSON schema](#) 53
 [CheckFileInfo](#) 53
 [CheckFolderInfo](#) 54
 [EnumerateChildren](#) 54
 [PutRelativeFile](#) 55
[Full XML schema](#) 51

G

[Glossary](#) 6

H

Headers
 [custom HTTP](#) 10

I

[Implementer - security considerations](#) 50
[Index of security parameters](#) 50
[Informative references](#) 7
[Introduction](#) 6

J

[JSON Schema](#) 53

[CheckFileInfo](#) 53
[CheckFolderInfo](#) 54
[EnumerateChildren](#) 54
[PutRelativeFile](#) 55

M

Message processing
 [WOPI discovery details](#) 13
 [WOPI protocol server details](#) 23
Messages
 [syntax](#) 10
 [transport](#) 10

N

[Normative references](#) 6

O

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

P

Parameters
 [common URI](#) 11
[Parameters - security index](#) 50
[Preconditions](#) 9
[Prerequisites](#) 9
[Product behavior](#) 56
[PutRelativeFile JSON](#) 55

R

References
 [informative](#) 7
 [normative](#) 6
[Relationship to other protocols](#) 9

S

Security
 [implementer considerations](#) 50
 [parameter index](#) 50
[Standards assignments](#) 9
Syntax
 [messages - overview](#) 10

T

[Tracking changes](#) 57
[Transport](#) 10

V

[Vendor-extensible fields](#) 9
[Versioning](#) 9
[Viewing a document example](#) 48

W

- [WOPI discovery details](#) 13
 - [message processing](#) 13
- [WOPI protocol client details](#) 21
- [WOPI protocol server details](#) 22
 - [abstract data model](#) 22
 - [message processing](#) 23
- WOPI discovery details
 - [sequencing rules](#) 13
- WOPI protocol server details
 - [sequencing rules](#) 23

X

- [XML Schema](#) 51