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

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation (“this documentation”) for protocols, file formats, data portability, computer languages, and standards support. Additionally, overview documents cover inter-protocol relationships and interactions.

- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you can make copies of it in order to develop implementations of the technologies that are described in this documentation and can distribute portions of it in your implementations that use these technologies or in your documentation as necessary to properly document the implementation. You can also distribute in your implementation, with or without modification, any schemas, IDLs, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications documentation.

- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.

- **Patents.** Microsoft has patents that might cover your implementations of the technologies described in the Open Specifications documentation. Neither this notice nor Microsoft's delivery of this documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open Specifications document might be covered by the Microsoft Open Specifications Promise or the Microsoft Community Promise. If you would prefer a written license, or if the technologies described in this documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.

- **License Programs.** To see all of the protocols in scope under a specific license program and the associated patents, visit the Patent Map.

- **Trademarks.** The names of companies and products contained in this documentation might be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit www.microsoft.com/trademarks.

- **Fictitious Names.** The example companies, organizations, products, domain names, email addresses, logos, people, places, and events that are depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

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

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

**Support.** For questions and support, please contact dochelp@microsoft.com.
## Revision Summary

<table>
<thead>
<tr>
<th>Date</th>
<th>Revision History</th>
<th>Revision Class</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/20/2012</td>
<td>0.1</td>
<td>New</td>
<td>Released new document.</td>
</tr>
<tr>
<td>4/11/2012</td>
<td>0.1</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>7/16/2012</td>
<td>0.2</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>10/8/2012</td>
<td>1.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>2/11/2013</td>
<td>1.0</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>7/30/2013</td>
<td>1.0</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>11/18/2013</td>
<td>1.0</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>2/10/2014</td>
<td>1.0</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>4/30/2014</td>
<td>2.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>7/31/2014</td>
<td>2.1</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>10/30/2014</td>
<td>2.2</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>3/16/2015</td>
<td>3.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>9/10/2015</td>
<td>4.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>6/23/2016</td>
<td>5.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>9/7/2016</td>
<td>6.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>9/14/2016</td>
<td>6.0</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>10/17/2016</td>
<td>7.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>3/28/2017</td>
<td>8.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>6/20/2017</td>
<td>9.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>9/19/2017</td>
<td>10.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>12/12/2017</td>
<td>11.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>7/24/2018</td>
<td>11.0</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>10/1/2018</td>
<td>11.1</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
</tbody>
</table>
# Table of Contents

1 Introduction ................................................................................. 6
   1.1 Glossary .............................................................................. 6
   1.2 References .......................................................................... 7
   1.2.1 Normative References ....................................................... 7
   1.2.2 Informative References ..................................................... 8
   1.3 Overview ............................................................................. 8
   1.4 Relationship to Other Protocols .............................................. 10
   1.5 Prerequisites/Preconditions .................................................. 10
   1.6 Applicability Statement ......................................................... 10
   1.7 Versioning and Capability Negotiation .................................. 10
   1.8 Vendor-Extensible Fields ...................................................... 10
   1.9 Standards Assignments ........................................................ 10

2 Messages ...................................................................................... 12
   2.1 Transport ............................................................................. 12
   2.2 Message Syntax ..................................................................... 12
   2.2.1 Custom HTTP Headers ....................................................... 12
   2.2.2 Common URI Parameters .................................................. 12

3 Protocol Details .......................................................................... 15
   3.1 WOPI Discovery Details ....................................................... 15
      3.1.1 Abstract Data Model ....................................................... 15
      3.1.2 Timers .......................................................................... 15
      3.1.3 Initialization ................................................................. 15
      3.1.4 Higher-Layer Triggered Events ....................................... 15
      3.1.5 Message Processing Events and Sequencing Rules ............ 15
         3.1.5.1 HTTP://server/hosting/discovery ......... 15
            3.1.5.1.1 Discovery ......................................................... 15
               3.1.5.1.1.1 Request Body ........................................... 16
               3.1.5.1.1.2 Response Body ....................................... 16
               3.1.5.1.1.2.1 Elements ............................................. 16
                  3.1.5.1.1.2.1.1 wopi-discovery ....................... 16
               3.1.5.1.1.2.2 Complex Types ............................ 16
                  3.1.5.1.1.2.2.1 ct_wopi-discovery ................ 16
                  3.1.5.1.1.2.2.2 ct_net-zone ................................ 16
                  3.1.5.1.1.2.2.3 ct_app-name ................................ 16
                  3.1.5.1.1.2.2.4 ct_wopi-action ....................... 17
                  3.1.5.1.1.2.2.5 ct_proof-key ................................ 18
               3.1.5.1.1.2.3 Simple Types ...................................... 19
                  3.1.5.1.1.2.3.1 st_wopi-action-values ............... 19
                  3.1.5.1.1.2.3.2 st_wopi-action-requirements .......... 21
                  3.1.5.1.1.2.3.3 st_wopi-url-source................... 21
                  3.1.5.1.1.2.3.4 st_wopi-zone .............................. 23
               3.1.5.1.1.3 Processing Details ................................. 23
      3.1.6 Timer Events ................................................................. 24
      3.1.7 Other Local Events ....................................................... 24

2 WOPI Protocol Client Details ..................................................... 24
   2.2 Abstract Data Model .......................................................... 24
   2.2.2 Timers .......................................................................... 24
   2.2.3 Initialization ................................................................. 24
   2.2.4 Higher-Layer Triggered Events ....................................... 24
   2.2.5 Message Processing Events and Sequencing Rules ............ 24
   2.2.6 Timer Events ................................................................. 24
   2.2.7 Other Local Events ....................................................... 24

3.3 WOPI Protocol Server Details ................................................ 24
3.3.1 Abstract Data Model ........................................................................................................... 25
3.3.2 Timers ................................................................................................................................... 25
3.3.3 Initialization .......................................................................................................................... 25
3.3.4 Higher-Layer Triggered Events .............................................................................................. 25
3.3.5 Message Processing Events and Sequencing Rules ................................................................. 25
  3.3.5.1 HTTP://server/<...>/wopi*/files/<id> ...................................................................... 25
    3.3.5.1.1 CheckFileInfo .......................................................................................................... 27
    3.3.5.1.1.1 Request Body ...................................................................................................... 27
    3.3.5.1.1.2 Response Body .................................................................................................. 27
    3.3.5.1.1.3 Processing Details ............................................................................................... 32
    3.3.5.1.2 DeleteFile ............................................................................................................... 32
    3.3.5.1.2.1 Request Body ..................................................................................................... 33
    3.3.5.1.2.2 Response Body .................................................................................................. 33
    3.3.5.1.2.3 Processing Details ............................................................................................... 33
    3.3.5.1.3 ExecuteCellStorageRelativeRequest ....................................................................... 33
      3.3.5.1.3.1 Request Body ................................................................................................... 34
      3.3.5.1.3.2 Response Body ................................................................................................. 34
      3.3.5.1.3.3 Processing Details ............................................................................................. 34
    3.3.5.1.4 ExecuteCellStorageRequest ..................................................................................... 34
      3.3.5.1.4.1 Request Body ................................................................................................... 34
      3.3.5.1.4.2 Response Body ................................................................................................. 34
      3.3.5.1.4.3 Processing Details ............................................................................................. 34
    3.3.5.1.5 GetLock ..................................................................................................................... 34
      3.3.5.1.5.1 Request Body ................................................................................................... 35
      3.3.5.1.5.2 Response Body ................................................................................................. 35
      3.3.5.1.5.3 Processing Details ............................................................................................. 36
    3.3.5.1.6 GetRestrictedLink ....................................................................................................... 36
      3.3.5.1.6.1 Request Body ................................................................................................... 36
      3.3.5.1.6.2 Response Body ................................................................................................. 36
      3.3.5.1.6.3 Processing Details ............................................................................................. 37
    3.3.5.1.7 GetShareUrl ............................................................................................................... 37
      3.3.5.1.7.1 Request Body ................................................................................................... 37
      3.3.5.1.7.2 Response Body ................................................................................................. 37
      3.3.5.1.7.3 Processing Details ............................................................................................. 37
    3.3.5.1.8 Lock ........................................................................................................................... 38
      3.3.5.1.8.1 Request Body ................................................................................................... 39
      3.3.5.1.8.2 Response Body ................................................................................................. 39
      3.3.5.1.8.3 Processing Details ............................................................................................. 39
    3.3.5.1.9 PutRelativeFile ........................................................................................................... 39
      3.3.5.1.9.1 Request Body ................................................................................................... 40
      3.3.5.1.9.2 Response Body ................................................................................................. 40
      3.3.5.1.9.3 Processing Details ............................................................................................. 41
    3.3.5.1.10 PutUserInfo ............................................................................................................... 41
      3.3.5.1.10.1 Request Body ................................................................................................... 41
      3.3.5.1.10.2 Response Body ................................................................................................. 41
      3.3.5.1.10.3 Processing Details ............................................................................................. 41
    3.3.5.1.11 ReadSecureStore ...................................................................................................... 42
      3.3.5.1.11.1 Request Body ................................................................................................... 42
      3.3.5.1.11.2 Response Body ................................................................................................. 42
      3.3.5.1.11.3 Processing Details ............................................................................................. 43
    3.3.5.1.12 RefreshLock ............................................................................................................... 43
      3.3.5.1.12.1 Request Body ................................................................................................... 44
      3.3.5.1.12.2 Response Body ................................................................................................. 44
      3.3.5.1.12.3 Processing Details ............................................................................................. 44
    3.3.5.1.13 RenameFile ............................................................................................................... 44
      3.3.5.1.13.1 Request Body ................................................................................................... 45
      3.3.5.1.13.2 Response Body ................................................................................................. 45
      3.3.5.1.13.3 Processing Details ............................................................................................. 45
Appendix C: Product Behavior
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 1.5, 1.8, 1.9, 2, and 3 of this specification are normative. All other sections and examples in this specification are informative.

1.1 Glossary

This document uses the following terms:

- **ASCII**: The American Standard Code for Information Interchange (ASCII) is an 8-bit character-encoding scheme based on the English alphabet. ASCII codes represent text in computers, communications equipment, and other devices that work with text. ASCII refers to a single 8-bit ASCII character or an array of 8-bit ASCII characters with the high bit of each character set to zero.

- **atomic transaction**: A shared activity that provides mechanisms for achieving the atomicity, consistency, isolation, and durability (ACID) properties when state changes occur inside participating resource managers.

- **Augmented Backus-Naur Form (ABNF)**: A modified version of Backus-Naur Form (BNF), commonly used by Internet specifications. ABNF notation balances compactness and simplicity with reasonable representational power. ABNF differs from standard BNF in its definitions and uses of naming rules, repetition, alternatives, order-independence, and value ranges. For more information, see [RFC5234].

- **base64 encoding**: A binary-to-text encoding scheme whereby an arbitrary sequence of bytes is converted to a sequence of printable ASCII characters, as described in [RFC4648].

- **Coordinated Universal Time (UTC)**: A high-precision atomic time standard that approximately tracks Universal Time (UT). It is the basis for legal, civil time all over the Earth. Time zones around the world are expressed as positive and negative offsets from UTC. In this role, it is also referred to as Zulu time (Z) and Greenwich Mean Time (GMT). In these specifications, all references to UTC refer to the time at UTC-0 (or GMT).

- **file extension**: The sequence of characters in a file's name between the end of the file's name and the last "." character. Vendors of applications choose such sequences for the applications to uniquely identify files that were created by those applications. This allows file management software to determine which application are to be used to open a file.

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

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

- **Information Rights Management (IRM)**: A technology that provides persistent protection to digital data by using encryption, certificates, and authentication. Authorized recipients or users acquire a license to gain access to the protected files according to the rights or business rules that are set by the content owner.

- **Internet Assigned Numbers Authority (IANA)**: A central repository for the protocol name and number registries that are used in many Internet protocols.
**JavaScript Object Notation (JSON)**: A text-based, data interchange format that is used to transmit structured data, typically in Asynchronous JavaScript + XML (AJAX) web applications, as described in [RFC7159]. The JSON format is based on the structure of ECMAScript (Jscript, JavaScript) objects.

**Representational State Transfer (REST)**: A software architecture implementation for distributed hypermedia systems, such as the World Wide Web.

**Secure Sockets Layer (SSL)**: A security protocol that supports confidentiality and integrity of messages in client and server applications that communicate over open networks. SSL supports server and, optionally, client authentication using X.509 certificates [X509] and [RFC5280]. SSL is superseded by Transport Layer Security (TLS). TLS version 1.0 is based on SSL version 3.0 [SSL3].

**tenant**: A protocol client or protocol server that accesses a partition in a shared service database.

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

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

### 1.2 References

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

#### 1.2.1 Normative References

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


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


1.2.2 Informative References

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

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


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.
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:
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].

<table>
<thead>
<tr>
<th>Web Application Open Platform Interface</th>
<th>This Protocol</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP</td>
<td>HTTPS</td>
</tr>
<tr>
<td>TCP</td>
<td>IP</td>
</tr>
</tbody>
</table>

![Figure 3: This protocol in relation to other protocols](image)

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-FSSHTTP]). 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.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>IANA assigned port for HTTP</td>
<td>80</td>
<td><a href="http://www.iana.org/assignments/part-number">http://www.iana.org/assignments/part-number</a></td>
</tr>
<tr>
<td>IANA assigned port for SSL</td>
<td>443</td>
<td><a href="http://www.iana.org/assignments/part-number">http://www.iana.org/assignments/part-number</a></td>
</tr>
</tbody>
</table>
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.

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authorization</td>
<td>Defined in [RFC2616]. This header MUST have the value &quot;Bearer &quot; + &lt;token&gt; as specified in section 2.2.2. Note that there MUST be a space between &quot;Bearer&quot; and &lt;token&gt;.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-AppEndpoint</td>
<td>A string specifying the endpoint of the WOPI client sending the request. This string MUST NOT be used for anything other than logging.</td>
</tr>
<tr>
<td>X-WOPI-ClientVersion</td>
<td>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.</td>
</tr>
<tr>
<td>X-WOPI-MachineName</td>
<td>A string indicating the name of the machine making the call, which MUST NOT be used for anything other than logging.</td>
</tr>
<tr>
<td>X-WOPI-PerfTraceRequested</td>
<td>A Boolean value that indicates that the WOPI client has requested the WOPI server to return a value for X-WOPI-PerfTrace.</td>
</tr>
<tr>
<td>X-WOPI-CorrelationId</td>
<td>A string that the WOPI server uses when logging server activity to correlate that activity with WOPI client activity.</td>
</tr>
<tr>
<td>X-WOPI-UsingRestrictedScenario</td>
<td>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. 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.</td>
</tr>
<tr>
<td>X-WOPI-Proof</td>
<td>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 values provided in ct_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:</td>
</tr>
</tbody>
</table>
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 `$ct_proof-key$`.

### 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 values provided in `$ct_proof-key$` (section 3.1.5.1.2.2.5) in Discovery (section 3.1.5.1) as the public key.

The value of X-WOPI-ProofOld 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

This data is then signed as described above.

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 `$ct_proof-key$`.

### 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.

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-HostEndpoint</td>
<td>A string specifying the endpoint of the WOPI server handling the request. This string MUST NOT be used for anything other than logging.</td>
</tr>
<tr>
<td>X-WOPI-ServerVersion</td>
<td>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.</td>
</tr>
<tr>
<td>X-WOPI-MachineName</td>
<td>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.</td>
</tr>
</tbody>
</table>
### 2.2.2 Common URI Parameters

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

<table>
<thead>
<tr>
<th>URI Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;token&gt;</code></td>
<td>A <em>string</em> used by the WOPI server to determine the identity and permissions of the issuer of the WOPI request. The <em>token</em> is generated by the WOPI server and learned by the WOPI client before the first WOPI request. The WOPI server passes the <em>token</em> to the WOPI client in the initiating request described in section 3.1.5.1.1.2.3.3. The WOPI client requires no understanding of the format or content of the <em>token</em>. The <em>token</em> SHOULD expire after a length of time deemed reasonable by the implementer of the WOPI server. The <em>token</em> SHOULD only provide access to resources associated with the WOPI request. That is, the <em>token</em> SHOULD be scoped to a specific user and set of resources.</td>
</tr>
<tr>
<td><code>&lt;id&gt;</code></td>
<td>The <em>id</em> represents the file or folder being operated on by the WOPI client. A WOPI server MUST issue a unique <em>id</em> for any resource that a WOPI client operates on.</td>
</tr>
</tbody>
</table>

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 can be included with all WOPI requests.

<table>
<thead>
<tr>
<th>URI Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;sc&gt;</code></td>
<td>The <em>sc</em> is a <em>string</em> that the WOPI server MAY include with a WOPI request. The WOPI client MAY in turn return the value of <em>sc</em> as the value of the X-WOPI-SessionContext HTTP header when making the <strong>CheckFileInfo</strong> (section 3.3.5.1.1) and <strong>CheckFolderInfo</strong> (section 3.3.5.2.1) WOPI requests.</td>
</tr>
</tbody>
</table>
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.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="HTTP://server/hosting/discovery">HTTP://server/hosting/discovery</a></td>
<td>Provides access to information about the capabilities of a WOPI client and the mechanisms for invoking those abilities through URIs.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discovery</td>
<td>Returns data that describes the supported abilities of the WOPI client and how to call these abilities through URIs.</td>
</tr>
</tbody>
</table>

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 3.1.5.1.1.2.3) 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 uses 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.

```xml
<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 3.1.5.1.2.3.1) 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 3.1.5.1.2.3.2) that specifies the required capabilities of the WOPI server to be able to use this action.

**urlsrc**: A st_wopi-url-source attribute (section 3.1.5.1.2.3.3) 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 3.1.5.1.2.3.1) 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 3.1.5.1.2.3.1) 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.

**targetext**: A **string** attribute that specifies the file extension of the target object. Used by actions that require both source and target host objects.

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

```xml
<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="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:attribute name="targetext" type="xs:string" use="optional"/>
</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.

The public keys are specified in two different formats.

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 **base64-encoded** value representing public key information that is compatible with the unmanaged Microsoft Cryptographic API ([MSDN-CAPI]).

**value**: A base64-encoded value representing public key information that is compatible with the unmanaged Microsoft Cryptographic API ([MSDN-CAPI]).

**modulus**: A base64-encoded value representing the RSA modulus as described in [RFC3447] section A.1.1. This modulus value MUST only be used with the value of the **exponent** attribute to produce a complete public key.

**oldmodulus**: A base64-encoded value representing the RSA modulus as described in [RFC3447] section A.1.1. This modulus value MUST only be used with the value of the **oldexponent** attribute to produce a complete public key.

**exponent**: A base64-encoded ([RFC4648]) value representing the RSA publicExponent as described in [RFC3447] section A.1.1. This exponent value MUST only be used with the value of the **modulus** attribute to produce a complete public key.

**oldexponent**: A base64-encoded ([RFC4648]) value representing the RSA publicExponent as described in [RFC3447] section A.1.1. This exponent value MUST only be used with the value of the **oldmodulus** attribute to produce a complete public key.

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

embededit: An action that indicates that the WOPI client supports an editable 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.

**formsubmit**: 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 **formsubmit** 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.

**preloadview**: An action that indicates that the WOPI client supports preloading static content prior to using the **view** action.

**preloadedit**: An action that indicates that the WOPI client supports preloading static content prior to using the **edit** action.

**rtc**: An action that indicates the location of a real-time communication endpoint for the file type.

**getinfo**: An action that indicates that the WOPI client supports retrieving additional information about the WOPI client in relation to the file type.

**convert**: An action that indicates that the WOPI client supports converting a document in a binary format, such as doc, into a modern format, like docx.

**syndicate**: An action that indicates that the WOPI client supports a view of the file type in web part mode.

**legacywebservice**: An action that indicates that the WOPI client supports a view of the file type in legacy web service mode.

**collab**: An action that indicates that the WOPI client supports joining collaboration service.

**formpreview**: An action that indicates that the WOPI client supports previewing the file type via a form-style interface.

**documentchat**: An action that indicates that the WOPI client supports chatting with other coauthors in this document.

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

```xml
<xs:simpleType name="st_wopi-action-values">
<xs:restriction base="xs:string">
<xs:enumeration value="view" />
<xs:enumeration value="mobileView" />
<xs:enumeration value="embedview" />
<xs:enumeration value="embededit" />
<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 3.3.5.1.8), **Unlock** (see section 3.3.5.1.15), **RefreshLock** (see section 3.3.5.1.12), and **UnlockAndRelock** (see section 3.3.5.1.16).
- **cobalt**: Indicates that the WOPI server MUST support the system described in [MS-FSSHTTP] through **ExecuteCellStorageRequest** (see section 3.3.5.1.4) and **ExecuteCellStorageRelativeRequest** (see section 3.3.5.1.3).
- **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 3.3.5.1.2).

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

```xml
<xs:simpleType name="st_wopi-action-requirements">
  <xs:restriction base="xs:string">
    <xs:pattern value="/^(update|locks|cobalt|containers)s*\s*,\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 receives a request from the browser that is required to use a WOPI action, an HTTP POST request MUST be sent from the WOPI server that navigates the browser along with a query string **WOPISrc** parameter and the **access_token** access token as the request body.

A WOPI server can also send an **access_token_TTL** as the request body.

- **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** containing the number of milliseconds since January 1, 1970 UTC and representing the expiration date and time stamp of the **access_token**.

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 could 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 includes the preferred UI language in the format described in [RFC1766].

**DC_LLCC**: Indicates that the WOPI server includes preferred data language in the format described in [RFC1766] for cases where language can affect data calculation.

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

**DISABLE_ASYNC**: Indicates that the WOPI server includes the value "true" to prevent the attendee from navigating a file. For example, when using the attendee action (see **st_wopi-action-values** in section 3.1.5.1.1.2.3.1).

**DISABLE_BROADCAST**: Indicates that the WOPI server includes 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.

**FULLSCREEN**: Indicates that the WOPI server includes the value "true" to load the file type in full-screen mode.

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

**THEME_ID**: Indicates that the WOPI server includes a value to designate the theme used. Current values are "1" to indicate a light-colored theme and "2" to indicate a darker colored theme.

**BUSINESS_USER**: Indicates that the WOPI server includes the value "1" to indicate that the user is a business user.
**DISABLE_CHAT**: Indicates that the WOPI server includes the value "1" to load a view of the document that does not create or join a chat session.

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

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

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**.

### 3.1.5.1.1.2.3.4 **st_wopi-zone**

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

**Values:**

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

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

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

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

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

```xml
<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 3.3.5.1.4 and section 3.3.5.1.3 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

<table>
<thead>
<tr>
<th>Resource</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="HTTP://server/">HTTP://server/</a>&lt;...&gt;/wopi*/files/&lt;id&gt;</td>
<td>Provides access to information about a file and allows for file-level operations.</td>
</tr>
<tr>
<td><a href="HTTP://server/">HTTP://server/</a>&lt;...&gt;/wopi*/folders/&lt;id&gt;</td>
<td>Provides access to information about a folder and allows for folder level operations.</td>
</tr>
<tr>
<td><a href="HTTP://server/">HTTP://server/</a>&lt;...&gt;/wopi*/files/&lt;id&gt;/contents</td>
<td>Provides access to operations that get and update the contents of a file.</td>
</tr>
<tr>
<td><a href="HTTP://server/">HTTP://server/</a>&lt;...&gt;/wopi*/folders/&lt;id&gt;/children</td>
<td>Provides access to the files and folders in a folder.</td>
</tr>
</tbody>
</table>

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: 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.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CheckFileInfo</td>
<td>Returns information about a file.</td>
</tr>
<tr>
<td>DeleteFile</td>
<td>Removes a file from the WOPI server.</td>
</tr>
<tr>
<td>ExecuteCellStorageRelativeRequest</td>
<td>Changes the contents of the file in accordance with [MS-FSSHTTP].</td>
</tr>
<tr>
<td>ExecuteCellStorageRequest</td>
<td>Changes the contents of the file in accordance with [MS-FSSHTTP].</td>
</tr>
<tr>
<td>GetLock</td>
<td>Retrieves a lock for editing a file.</td>
</tr>
<tr>
<td>GetRestrictedLink</td>
<td>Gets a link to a file through which a user is able to operate on a file in a limited way.</td>
</tr>
<tr>
<td>GetShareUrl</td>
<td>Gets a URI to the file that is suitable for sharing with other users.</td>
</tr>
<tr>
<td>Lock</td>
<td>Takes a lock for editing a file.</td>
</tr>
<tr>
<td>PutRelativeFile</td>
<td>Creates a copy of a file on the WOPI server.</td>
</tr>
<tr>
<td>PutUserInfo</td>
<td>Stores user information on the WOPI server.</td>
</tr>
<tr>
<td>ReadSecureStore</td>
<td>Accesses the WOPI server's implementation of a secure store.</td>
</tr>
<tr>
<td>RefreshLock</td>
<td>Refreshes a lock for editing a file.</td>
</tr>
<tr>
<td>RenameFile</td>
<td>Renames a file.</td>
</tr>
<tr>
<td>RevokeRestrictedLink</td>
<td>Revokes all links to a file through which a number of users are able to operate on a file in a limited way.</td>
</tr>
<tr>
<td>Unlock</td>
<td>Releases a lock for editing a file.</td>
</tr>
<tr>
<td>UnlockAndRelock</td>
<td>Releases and then retakes a lock for editing a file.</td>
</tr>
</tbody>
</table>

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 can be included with this request.

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-SessionContext</td>
<td>The value of the &lt;sc&gt; URI parameter (section 2.2.2).</td>
</tr>
</tbody>
</table>
The response message for this operation can result in the following status codes.

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
<tr>
<td>401</td>
<td>Token is invalid</td>
</tr>
<tr>
<td>404</td>
<td>File unknown/User Unauthorized</td>
</tr>
<tr>
<td>500</td>
<td>Server error</td>
</tr>
</tbody>
</table>

### 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

{
    "AllowAdditionalMicrosoftServices":{"type":bool,"default":false,"optional":true},
    "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},
    "ClientId":{"type":string,"default":"","optional":true},
    "CloseButtonClosesWindow":{"type":bool,"default":false,"optional":true},
    "ClosePostMessage":{"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},
    "EditAndReplyUrl":{"type":string,"default":"","optional":true},
    "EditModePostMessage":{"type":bool,"default":false,"optional":true},
    "EditNotificationPostMessage":{"type":bool,"default":false,"optional":true},
    "FileExtension":{"type":string,"default":"","optional":true},
    "FileNameMaxLength":{"type":integer,"default":250,"optional":true},
    "FileSharingPostMessage":{"type":bool,"default":false,"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},
    "PostMessageOrigin":{"type":string,"default":"","optional":true},
    "PresenceProvider":{"type":string,"default":"","optional":true},
    "IrmPolicyDescription":{"type":string,"default":"","optional":true},
    "IrmPolicyTitle":{"type":string,"default":"","optional":true},
    "LicenseCheckForEditIsEnabled":{"type":bool,"default":false,"optional":true},
    "OwnerId":{"type":string,"optional":false},
    "PostMessageOrigin":{"type":string,"default":"","optional":true},
    "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},
"SignInUrl": {"type": "string", "default": ",", "optional": true},
"SignoutUrl": {"type": "string", "default": ",", "optional": true},
"Size": {"type": "int", "optional": false},
"SupportedShareUrlTypes": {"type": "string[]", "default": [], "optional": true},
"SupportsCoauth": {"type": "bool", "default": false, "optional": true},
"SupportsCobalt": {"type": "bool", "default": false, "optional": true},
"SupportsExtendedLockLength": {"type": "bool", "default": false, "optional": true},
"SupportsFileCreation": {"type": "bool", "default": false, "optional": true},
"SupportsFolders": {"type": "bool", "default": false, "optional": true},
"SupportGetLock": {"type": "bool", "default": false, "optional": true},
"SupportsLocks": {"type": "bool", "default": false, "optional": true},
"SupportsRename": {"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},
"SupportsUserInfo": {"type": "bool", "default": false, "optional": true},
"TenantId": {"type": "string", "default": ",", "optional": true},
"TermsOfUseUrl": {"type": "string", "default": ",", "optional": true},
"TimeZone": {"type": "string", "default": ",", "optional": true},
"UniqueContentId": {"type": "string", "default": ",", "optional": true},
"UserCanAttend": {"type": "bool", "default": false, "optional": true},
"UserCanNotWriteRelative": {"type": "bool", "default": false, "optional": true},
"UserCanRename": {"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},
"UserInfo": {"type": "string", "default": ",", "optional": true},
"WebEditingDisabled": {"type": "bool", "default": false, "optional": true},
"WorkflowType": {"type": "string[]", "default": [], "optional": true},
"WorkflowUrl": {"type": "string", "default": ",", "optional": true} }

AllowAdditionalMicrosoftServices: A Boolean value that indicates the WOPI client allows connections to external Microsoft services to provide additional functionality. If this value is false, then the WOPI client MUST NOT allow such connections.

AllowExternalMarketplace: A Boolean value that indicates the WOPI client allows 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 displays to the user that indicates the brand name of the WOPI server.

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

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

BreadcrumbDocUrl: A URI to a web page that the WOPI client navigates to when the user clicks on UI that displays BreadcrumbDocName.
BreadcrumbFolderName: A string that the WOPI client displays 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 navigates 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.

ClosePostMessage: A Boolean value that indicates that the WOPI client SHOULD notify the WOPI server in the event that the user closes the rendering or editing client currently using this file. <1>

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.

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

EditModePostMessage: A Boolean value that indicates that the WOPI client SHOULD notify the WOPI server in the event that the user attempts to edit the file.

EditNotificationPostMessage: A Boolean value that indicates that the WOPI client SHOULD notify the WOPI server in the event that the user attempts to edit the file.

FileExtension: A string specifying the file extension of the file. This value MUST begin with a ".". <2>

FileNameMaxLength: An integer indicating the maximum length for file names, including the file extension, supported by the WOPI server.

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

FileSharingPostMessage: A Boolean value that indicates that the WOPI client SHOULD notify the WOPI server in the event that the user attempts 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**.

**LicenseCheckInEditIsEnabled:** A **Boolean** value that indicates that the WOPI client SHOULD take measures to ensure the user is properly licensed prior to allowing editing of the file.

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

**PostMessageOrigin:** A **string** that the WOPI client MUST use as the **targetOrigin** parameter when sending messages as described in [W3C-HTML5WEBMSG].

**PresenceProvider:** A **string** that identifies the provider of information that a WOPI client uses 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. <3>

**SignInUrl:** A URI that will sign the current user into the WOPI server supported authentication system.

**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.

SupportedShareUrlTypes: An array of strings indicating the share URL types supported by the WOPI server.

SupportsCoauth: A Boolean value that indicates that the WOPI server supports multiple users making changes to this file simultaneously.<4>

SupportsCobalt: A Boolean value that indicates that the WOPI server supports ExecuteCellStorageRequest (see section 3.3.5.1.4) and ExecuteCellStorageRelativeRequest (see section 3.3.5.1.3) operations for this file.

SupportsExtendedLockLength: A Boolean value that indicates that the WOPI server supports lock IDs up to 1024 ASCII characters in length. <5>

SupportsFileCreation: A Boolean value that indicates that the WOPI server supports creating new files using the WOPI client.<6>

SupportsFolders: A Boolean value that indicates that the WOPI server supports EnumerateChildren (see section 3.3.5.4.1) and DeleteFile (see section 3.3.5.1.2) operations for this file.

SupportsGetLock: A Boolean value that indicates that the WOPI server supports GetLock (see section 3.3.5.1.5).<7>

SupportsLocks: A Boolean value that indicates that the WOPI server supports Lock (see section 3.3.5.1.8), Unlock (see section 3.3.5.1.15), RefreshLock (see section 3.3.5.1.12), and UnlockAndRelock (see section 3.3.5.1.16) operations for this file.

SupportsRename: A Boolean value that indicates that the WOPI server supports RenameFile (see section 3.3.5.1.13).<8>

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 3.3.5.1.9) operations for this file.

SupportsUserInfo: A Boolean value that indicates that the WOPI server supports PutUserInfo (see section 3.3.5.1.10).<9>

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

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.

UniqueContentId: A string that uniquely represents the file contents.<10> This value MUST change when the file content changes. This value MUST also be equivalent for all files with equivalent contents.

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 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.

UserCanRename: A Boolean value that indicates the user has permissions to rename the file.

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.

UserInfo: A string that the WOPI client SHOULD use to verify the user’s licensing status. The WOPI client MAY ignore this string if it does not recognize the contents.

Version: A string representing the current version of the file based on the WOPI server’s file versioning schema. This value MUST change when the file content 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.

WorkflowPostMessage: A Boolean value that indicates that the WOPI client SHOULD notify the WOPI server in the event that the user attempts to trigger a workflow on the file.

WorkflowType: An array of strings representing the workflow types that are available for the file.

WorkflowUrl: A string representing the current version of the file based on the WOPI server’s file versioning schema. This value MUST change when the file content changes, and version values MUST never repeat for a given file.

3.3.5.1.1.3 Processing Details

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

3.3.5.1.2 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.

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Override</td>
<td>A string specifying requested operation from the WOPI server. Required.</td>
<td>The string &quot;DELETE&quot;.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
</tbody>
</table>
### 3.3.5.1.2.1 Request Body

None.

### 3.3.5.1.2.2 Response Body

None.

### 3.3.5.1.2.3 Processing Details

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

### 3.3.5.1.3 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.

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Override</td>
<td>A string specifying the requested operation from the WOPI server. Required.</td>
<td>The string &quot;COBALT&quot;.</td>
</tr>
<tr>
<td>X-WOPI-Size</td>
<td>An integer specifying the size of the request body.</td>
<td>An integer.</td>
</tr>
<tr>
<td>X-WOPI-RelativeTarget</td>
<td>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.</td>
<td>A string.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
<tr>
<td>404</td>
<td>File unknown/User unauthorized</td>
</tr>
<tr>
<td>500</td>
<td>Server error</td>
</tr>
<tr>
<td>501</td>
<td>Unsupported</td>
</tr>
</tbody>
</table>
3.3.5.1.3.1 Request Body

Content that conforms to the Body element specified in [MS-FSSHTTP] section 2.2.2.1.

3.3.5.1.3.2 Response Body

Content that conforms to the Body element specified in [MS-FSSHTTP] section 2.2.2.2.

3.3.5.1.3.3 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.

3.3.5.1.4 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.

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Override</td>
<td>A string specifying requested operation from the WOPI server. Required.</td>
<td>The string &quot;COBALT&quot;.</td>
</tr>
<tr>
<td>X-WOPI-Size</td>
<td>An integer specifying the size of the request body.</td>
<td>An integer.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
<tr>
<td>404</td>
<td>File unknown/User unauthorized</td>
</tr>
<tr>
<td>500</td>
<td>Server error</td>
</tr>
<tr>
<td>501</td>
<td>Unsupported</td>
</tr>
</tbody>
</table>

3.3.5.1.4.1 Request Body

Content that conforms to the Body element specified in [MS-FSSHTTP] section 2.2.2.1.

3.3.5.1.4.2 Response Body

Content that conforms to the Body element specified in [MS-FSSHTTP] section 2.2.2.2.

3.3.5.1.4.3 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.

3.3.5.1.5 GetLock
Retrieve a lock for editing a file.<11>

HTTP Verb: POST

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

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

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Override</td>
<td>A string specifying the requested operation from the WOPI server. Required.</td>
<td>The string &quot;GET_LOCK&quot;.</td>
</tr>
</tbody>
</table>

The response message for this operation can contain the following HTTP headers.

<table>
<thead>
<tr>
<th>Response header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Lock</td>
<td>A string provided by the WOPI client that the WOPI server MUST use to identify the lock on the file. This header MUST be included when responding with the following status codes:</td>
<td>A string.</td>
</tr>
<tr>
<td></td>
<td>200</td>
<td></td>
</tr>
<tr>
<td></td>
<td>409</td>
<td></td>
</tr>
<tr>
<td>X-WOPI-LockFailureReason</td>
<td>A string indicating why the operation failed. There is no standard for how this string is to be formatted. This string MUST NOT be used for anything other than logging. This header MAY only be included when responding with the 409 status code.</td>
<td>A string.</td>
</tr>
<tr>
<td>X-WOPI-LockedByOtherInterface</td>
<td>A string indicating that the file is currently locked by another client. This header SHOULD only be included when responding with the 409 status code.</td>
<td>The string &quot;true&quot;.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
<tr>
<td>401</td>
<td>Token is invalid</td>
</tr>
<tr>
<td>404</td>
<td>File unknown/User unauthorized</td>
</tr>
<tr>
<td>409</td>
<td>Lock mismatch/locked by another interface; an X-WOPI-Lock response header containing the value of the current lock on the file MUST be included when using this response code</td>
</tr>
<tr>
<td>500</td>
<td>Server error</td>
</tr>
<tr>
<td>501</td>
<td>Unsupported</td>
</tr>
</tbody>
</table>

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
Retrieves an existing lock on a file.

### 3.3.5.1.6 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.

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Override</td>
<td>A <code>string</code> specifying requested operation from the WOPI server. Required.</td>
<td>The <code>string</code> &quot;GET_RESTRICTED_LINK&quot;.</td>
</tr>
<tr>
<td>X-WOPI-RestrictedLink</td>
<td>A <code>string</code> specifying the type of restricted link being requested by the WOPI client. The valid values of this <code>string</code> are determined through convention. The only valid value currently is &quot;FORMS&quot;. Required.</td>
<td>A <code>string</code> that is a valid value for a restricted link type. For example, &quot;FORMS&quot;.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Response header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-RestrictedUseLink</td>
<td>A <code>string</code> specifying the URL that the WOPI client requested. For example, the URL that allows the user to fill out a form. Required.</td>
<td>A <code>string</code> that is a valid URL.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
<tr>
<td>404</td>
<td>File unknown/User unauthorized</td>
</tr>
<tr>
<td>500</td>
<td>Server error</td>
</tr>
<tr>
<td>501</td>
<td>Unsupported</td>
</tr>
</tbody>
</table>

### 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

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

3.3.5.1.7 GetShareUrl

Gets a URI to the file that is suitable for sharing with other users.

HTTP Verb: POST

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

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

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Override</td>
<td>A string specifying the requested operation from the WOPI server. Required.</td>
<td>The string &quot;GET_SHARE_URL&quot;.</td>
</tr>
<tr>
<td>X-WOPI-UrlType</td>
<td>A string indicating what share URL type to return. Required.</td>
<td>A string.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
<tr>
<td>401</td>
<td>Token is invalid</td>
</tr>
<tr>
<td>404</td>
<td>File unknown/User unauthorized</td>
</tr>
<tr>
<td>500</td>
<td>Server error</td>
</tr>
<tr>
<td>501</td>
<td>Unsupported</td>
</tr>
</tbody>
</table>

3.3.5.1.7.1 Request Body

None.

3.3.5.1.7.2 Response Body

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

JSON:

```json
{
  "ShareUrl":{"type":"string","default":"","optional":true}
}
```

ShareUrl: A URI to a location that allows authorized users to view the file.

3.3.5.1.7.3 Processing Details
If the X-WOPI-UrlType header is not present, or the WOPI server does not recognize the value, then the WOPI server MUST respond with a 501 status code.

### 3.3.5.1.8 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.

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Override</td>
<td>A string specifying the requested operation from the WOPI server. Required.</td>
<td>The string &quot;LOCK&quot;.</td>
</tr>
<tr>
<td>X-WOPI-Lock</td>
<td>A string provided by the WOPI client that the WOPI server MUST use to identify the lock on the file. Required.</td>
<td>A string.</td>
</tr>
</tbody>
</table>

The response message for this operation can contain the following HTTP headers.

<table>
<thead>
<tr>
<th>Response header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Lock</td>
<td>A string provided by the WOPI client that the WOPI server MUST use to identify the lock on the file. This header SHOULD be included when responding with the 409 status code if a different shared lock exists on the file. This header SHOULD NOT be included when responding with the 200 status code.</td>
<td>A string.</td>
</tr>
<tr>
<td>X-WOPI-LockFailureReason</td>
<td>A string indicating why the operation failed. There is no standard for how this string is to be formatted. This string MUST NOT be used for anything other than logging. This header SHOULD only be included when responding with the 409 status code.</td>
<td>A string.</td>
</tr>
<tr>
<td>X-WOPI-LockedByOtherInterface</td>
<td>A string indicating that the file is currently locked by another client. This header SHOULD only be included when responding with the 409 status code.</td>
<td>The string &quot;true&quot;.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
<tr>
<td>401</td>
<td>Token is invalid</td>
</tr>
<tr>
<td>404</td>
<td>File unknown/User unauthorized</td>
</tr>
<tr>
<td>409</td>
<td>Lock mismatch/locked by another interface;</td>
</tr>
<tr>
<td>500</td>
<td>Server error</td>
</tr>
<tr>
<td>501</td>
<td>Unsupported</td>
</tr>
</tbody>
</table>
3.3.5.1.8.1 Request Body
None.

3.3.5.1.8.2 Response Body
None.

3.3.5.1.8.3 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.

3.3.5.1.9 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.

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Override</td>
<td>A string specifying the requested operation from the WOPI server. Required.</td>
<td>The string &quot;PUT_RELATIVE&quot;.</td>
</tr>
<tr>
<td>X-WOPI-SuggestedTarget</td>
<td>A UTF-7 string specifying either a file extension or a full file name. If only the extension is provided, the name of the initial file without extension SHOULD be combined with the extension to create the proposed name. The WOPI server MUST modify the proposed name as needed 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.</td>
<td>A string that is a file extension or a full file name.</td>
</tr>
<tr>
<td>X-WOPI-RelativeTarget</td>
<td>A UTF-7 string that specifies a full file name. The WOPI server MUST NOT modify the name to fulfill the request. When a file with the specified name already exists, if the X-WOPI-OverwriteRelativeTarget request header is set to false, or if the X-WOPI-OverwriteRelativeTarget request header is set to true and the file is locked, the host MUST respond with a 409 status code.</td>
<td>A string that is a full file name.</td>
</tr>
<tr>
<td>X-WOPI-OverwriteRelativeTarget</td>
<td>A Boolean value that specifies whether the host MUST overwrite the file name if it exists.</td>
<td>A Boolean value.</td>
</tr>
<tr>
<td>X-WOPI-Size</td>
<td>An integer that specifies the size of the file in bytes.</td>
<td>An integer.</td>
</tr>
</tbody>
</table>

The response message for this operation can contain the following HTTP headers.

<table>
<thead>
<tr>
<th>Response header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-</td>
<td>A UTF-7 string provided by the WOPI server that the</td>
<td>A string.</td>
</tr>
</tbody>
</table>
### Response header

<table>
<thead>
<tr>
<th>Response header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ValidRelativeTarget</td>
<td>WOPI client can use as the value of X-WOPI-RelativeTarget in a future PutRelativeFile operation. This header SHOULD only be included when responding with the 409 status code if a file with the specified name already exists.</td>
<td></td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
<tr>
<td>400</td>
<td>Specified name is illegal</td>
</tr>
<tr>
<td>401</td>
<td>Token is invalid</td>
</tr>
<tr>
<td>404</td>
<td>File unknown/User unauthorized</td>
</tr>
<tr>
<td>409</td>
<td>Target file already exists or the file is locked;</td>
</tr>
<tr>
<td>413</td>
<td>File is too large. The maximum size is implementation-specific.</td>
</tr>
<tr>
<td>500</td>
<td>Server error</td>
</tr>
<tr>
<td>501</td>
<td>Unsupported</td>
</tr>
</tbody>
</table>

#### 3.3.5.1.9.1 Request Body

The binary contents of the file.

#### 3.3.5.1.9.2 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":null,"optional":false},
    "HostViewUrl":{"type":"string","default":null,"optional":true},
    "HostEditUrl":{"type":"string","default":null,"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 \textit{id} is the WOPI server’s \textit{id} for the new file, and \textit{token} is the WOPI server’s credentials token for accessing the new file.

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

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

\subsection*{3.3.5.1.9.3 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.

\subsection*{3.3.5.1.10 PutUserInfo}

Stores user information on the WOPI server.\footnote{12}

HTTP Verb: POST

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

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

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Override</td>
<td>A \textit{string} specifying the requested operation from the WOPI server. Required.</td>
<td>The \textit{string} &quot;PUT_USER_INFO&quot;.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
<tr>
<td>401</td>
<td>Token is invalid</td>
</tr>
<tr>
<td>404</td>
<td>File unknown/User unauthorized</td>
</tr>
<tr>
<td>500</td>
<td>Server error</td>
</tr>
<tr>
<td>501</td>
<td>Unsupported</td>
</tr>
</tbody>
</table>

\subsection*{3.3.5.1.10.1 Request Body}

A \textit{string} that the WOPI server MUST store. If stored, the WOPI server MUST associate it with the current user. The WOPI server SHOULD include this \textit{string} in the UserInfo property.

\subsection*{3.3.5.1.10.2 Response Body}

None.

\subsection*{3.3.5.1.10.3 Processing Details}

The WOPI server MUST store the string passed in the request body and MUST associate it with the current user. The WOPI server MUST include this \textit{string} in the \textit{UserInfo} property when responding to \textit{CheckFileInfo} (see section 3.3.5.1.1) requests for this user.
3.3.5.1.11 ReadSecureStore

Access the WOPI server's implementation of a secure store as specified in [MS-SSWPS]<13>. 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.

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Override</td>
<td>A string specifying requested operation from the WOPI server. Required.</td>
<td>The string &quot;READ_SECURE_STORE&quot;.</td>
</tr>
<tr>
<td>X-WOPI-ApplicationId</td>
<td>[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.</td>
<td>The value of the Application ID.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
<tr>
<td>401</td>
<td>Token is invalid</td>
</tr>
<tr>
<td>404</td>
<td>File unknown/User unauthorized</td>
</tr>
<tr>
<td>500</td>
<td>Server error</td>
</tr>
<tr>
<td>501</td>
<td>Unsupported</td>
</tr>
</tbody>
</table>

3.3.5.1.11.1 Request Body

None.

3.3.5.1.11.2 Response Body

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

JSON:

```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).

### 3.3.5.1.11.3 Processing Details

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

### 3.3.5.1.12 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.

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Override</td>
<td>A string specifying the requested operation from the WOPI server. Required.</td>
<td>The string &quot;REFRESH_LOCK&quot;.</td>
</tr>
<tr>
<td>X-WOPI-Lock</td>
<td>A string provided by the WOPI client that the WOPI server MUST use to identify the lock on the file. Required.</td>
<td>The string provided by the WOPI client through Lock (see section 3.3.5.1.8).</td>
</tr>
</tbody>
</table>

The response message for this operation can contain the following HTTP headers.

<table>
<thead>
<tr>
<th>Response header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Lock</td>
<td>A string provided by the WOPI client that the WOPI server uses to identify the lock on the file. This header MUST be included when responding with the 409 status code if no lock or a different shared lock exists on the file. This header SHOULD NOT be included when responding with the 200 status code.</td>
<td>A string.</td>
</tr>
<tr>
<td>X-WOPI-LockedByOtherInterface</td>
<td>A string indicating that the file is currently locked by another client. This header SHOULD be included when responding with the 409 status code if the file is locked by another client.</td>
<td>The string &quot;true&quot;.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
<tr>
<td>401</td>
<td>Token is invalid</td>
</tr>
<tr>
<td>404</td>
<td>File unknown/User unauthorized</td>
</tr>
<tr>
<td>409</td>
<td>Lock mismatch/locked by another interface;</td>
</tr>
</tbody>
</table>
### Status code

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>Server error</td>
</tr>
<tr>
<td>501</td>
<td>Unsupported</td>
</tr>
</tbody>
</table>

#### 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

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.

### 3.3.5.1.13 RenameFile

Renames a file. [<14>](#)

HTTP Verb: POST

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

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

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Override</td>
<td>A string specifying the requested operation from the WOPI server. Required.</td>
<td>The string &quot;RENAME_FILE&quot;.</td>
</tr>
<tr>
<td>X-WOPI-Lock</td>
<td>A string provided by the WOPI client that the WOPI server uses to identify the lock on the file. Required.</td>
<td>The string provided by the WOPI client through Lock (see section 3.3.5.1.8).</td>
</tr>
<tr>
<td>X-WOPI-RequestedName</td>
<td>A UTF-7 string that is a full file name. Required.</td>
<td>A string.</td>
</tr>
</tbody>
</table>

The response message for this operation can contain the following HTTP headers.

<table>
<thead>
<tr>
<th>Response header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-InvalidFileNameError</td>
<td>A string indicating why the operation failed. There is no standard for how this string is to be formatted. This string MUST NOT be used for anything other than logging. This header SHOULD only be included when responding with the 400 status code.</td>
<td>A string.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
</tbody>
</table>
3.3.5.1.13.1 Request Body

None.

3.3.5.1.13.2 Response Body

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

JSON:

```json
{
    "Name":{"type":"string","optional":false}
}
```

**Name**: The name of the renamed file without the path or file extension.

3.3.5.1.13.3 Processing Details

None.

3.3.5.1.14 RevokeRestrictedLink

Revoke 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.

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Override</td>
<td>A string specifying requested operation from the WOPI server. Required.</td>
<td>The string &quot;REVOKE_RESTRICTED_LINK&quot;.</td>
</tr>
<tr>
<td>X-WOPI-RestrictedLink</td>
<td>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 &quot;FORMS&quot;.</td>
<td>A string that is a valid value for a restricted link type. For example, &quot;FORMS&quot;.</td>
</tr>
</tbody>
</table>
The response message for this operation can result in the following status codes.

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
<tr>
<td>404</td>
<td>File unknown/User unauthorized</td>
</tr>
<tr>
<td>500</td>
<td>Server error</td>
</tr>
<tr>
<td>501</td>
<td>Unsupported</td>
</tr>
</tbody>
</table>

### 3.3.5.1.14.1 Request Body

None.

### 3.3.5.1.14.2 Response Body

None.

### 3.3.5.1.14.3 Processing Details

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

### 3.3.5.1.15 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.

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Override</td>
<td>A <strong>string</strong> specifying the requested operation from the WOPI server. Required.</td>
<td>The <strong>string</strong> &quot;UNLOCK&quot;.</td>
</tr>
<tr>
<td>X-WOPI-Lock</td>
<td>A <strong>string</strong> provided by the WOPI client that the WOPI server MUST use to identify the lock on the file. Required.</td>
<td>The <strong>string</strong> provided by the WOPI client through Lock (see section 3.3.5.1.8).</td>
</tr>
</tbody>
</table>

The response message for this operation can contain the following HTTP headers.

<table>
<thead>
<tr>
<th>Response header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Lock</td>
<td>A <strong>string</strong> provided by the WOPI client that the WOPI server uses to identify the lock on the file. This header SHOULD be included when responding with the 409 status code if no lock or a different shared lock exists on the file. This header SHOULD NOT be included when responding with the 200 status code.</td>
<td>A <strong>string</strong>.</td>
</tr>
<tr>
<td>X-WOPI-LockedByOtherInterface</td>
<td>A <strong>string</strong> indicating that the file is currently locked by another client. This header SHOULD be included when responding with the 409 status code if the file is locked</td>
<td>The **string&quot;true&quot;.</td>
</tr>
</tbody>
</table>
The response message for this operation can result in the following status codes.

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
<tr>
<td>401</td>
<td>Token is invalid</td>
</tr>
<tr>
<td>404</td>
<td>File unknown/User unauthorized</td>
</tr>
<tr>
<td>409</td>
<td>Lock mismatch/locked by another interface;</td>
</tr>
<tr>
<td>500</td>
<td>Server error</td>
</tr>
<tr>
<td>501</td>
<td>Unsupported</td>
</tr>
</tbody>
</table>

### 3.3.5.1.15.1 Request Body

None.

### 3.3.5.1.15.2 Response Body

None.

### 3.3.5.1.15.3 Processing Details

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

### 3.3.5.1.16 UnlockAndRelock

Release and retake a lock for editing a file. This MUST be an Atomic operation.

HTTP Verb: POST

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

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

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Override</td>
<td>A string specifying the requested operation from the WOPI server. Required.</td>
<td>The string &quot;LOCK&quot;.</td>
</tr>
<tr>
<td>X-WOPI-Lock</td>
<td>A string provided by the WOPI client that the WOPI server MUST use to identify the lock on the file. Required.</td>
<td>A string.</td>
</tr>
<tr>
<td>X-WOPI-OldLock</td>
<td>A string previously provided by the WOPI client that the WOPI server MUST have used to identify the lock on the file. Required.</td>
<td>The string provided by the WOPI client through Lock (see section 3.3.5.1.8).</td>
</tr>
</tbody>
</table>

The response message for this operation can contain the following HTTP headers.
<table>
<thead>
<tr>
<th>Response header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Lock</td>
<td>A string provided by the WOPI client that the WOPI server MUST use to identify the lock on the file. This header SHOULD be included when responding with the 409 status code if no lock or a different shared lock exists on the file. This header SHOULD NOT be included when responding with the 200 status code.</td>
<td>A string.</td>
</tr>
<tr>
<td></td>
<td><strong>X-WOPI-LockFailureReason</strong>: A string indicating why the operation failed. There is no standard for how this string is to be formatted. This string MUST NOT be used for anything other than logging. This header SHOULD only be included when responding with the 409 status code.</td>
<td>A string.</td>
</tr>
<tr>
<td>X-WOPI-LockedByOtherInterface</td>
<td>A string indicating that the file is currently locked by another client. This header SHOULD only be included when responding with the 409 status code.</td>
<td>The string &quot;true&quot;.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
<tr>
<td>401</td>
<td>Token is invalid</td>
</tr>
<tr>
<td>404</td>
<td>File unknown/User unauthorized</td>
</tr>
<tr>
<td>409</td>
<td>Lock mismatch/locked by another interface;</td>
</tr>
<tr>
<td>500</td>
<td>Server error</td>
</tr>
<tr>
<td>501</td>
<td>Unsupported</td>
</tr>
</tbody>
</table>

### 3.3.5.16.1 Request Body

None.

### 3.3.5.16.2 Response Body

None.

### 3.3.5.16.3 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.

### 3.3.5.2 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**: A unique identifier of a folder managed by the WOPI server.

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

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CheckFolderInfo</td>
<td>Returns information about a folder.</td>
</tr>
</tbody>
</table>

### 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 can be included with this request.

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-SessionContext</td>
<td>The value of the <code>&lt;sc&gt;</code> URI parameter (section 2.2.2).</td>
</tr>
</tbody>
</table>

#### 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**:

```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},
  "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},
  "PresenceUserRole":{"type":"string","default":"","optional":true},
  "PresenceUserURL":{"type":"string","default":"","optional":true},
  "PresenceUserUrl":{"type":"string","default":"","optional":true},
  "StartUrl":{"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 displays to the user as the branding image of the WOPI server.

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

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

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

**BreadcrumbDocUrl:** A URI to a web page that the WOPI client navigates to when the user clicks on the UI that displays **BreadcrumbDocName**.

**BreadcrumbFolderName:** A string that the WOPI client displays 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 navigates 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 uses 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 tenant.

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.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetFile</td>
<td>Request message to retrieve a file for the <a href="HTTP://server/">HTTP://server/</a>&lt;...&gt;/wopi*/files/&lt;id&gt;/contents operation.</td>
</tr>
</tbody>
</table>
### Operation Description

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PutFile</td>
<td>Request message to update a file for the <code>HTTP://server/&lt;...&gt;/wopi*/files/&lt;id&gt;/contents</code> operation.</td>
</tr>
</tbody>
</table>

### 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.

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-MaxExpectedSize</td>
<td>An <code>integer</code> 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 <code>integer</code> if this value is not set in the request.</td>
<td>An <code>integer</code>.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success.</td>
</tr>
<tr>
<td>401</td>
<td>Token is invalid</td>
</tr>
<tr>
<td>404</td>
<td>File unknown/User unauthorized.</td>
</tr>
<tr>
<td>500</td>
<td>Server error.</td>
</tr>
</tbody>
</table>

#### 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.

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-Override</td>
<td>A <em>string</em> specifying the requested operation from the WOPI server. Required.</td>
<td>The <em>string</em> &quot;PUT&quot;.</td>
</tr>
<tr>
<td>X-WOPI-Lock</td>
<td>A <em>string</em> provided by the WOPI client that the WOPI server used to identify the lock on the file. Required.</td>
<td>The <em>string</em> provided by the WOPI client through Lock (see section 3.3.5.1.8).</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success</td>
</tr>
<tr>
<td>401</td>
<td>Token is invalid</td>
</tr>
<tr>
<td>404</td>
<td>File unknown/User unauthorized</td>
</tr>
<tr>
<td>409</td>
<td>Lock mismatch/locked by another interface;</td>
</tr>
<tr>
<td>413</td>
<td>File is too large. The maximum file size is implementation-specific.</td>
</tr>
<tr>
<td>500</td>
<td>Server error</td>
</tr>
<tr>
<td>501</td>
<td>Unsupported</td>
</tr>
</tbody>
</table>

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 3.3.5.1.8) or the UnlockAndRelock operation (see section 3.3.5.1.16) 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.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EnumerateChildren</td>
<td>Returns a set of URIs that provides access to resources in the folder.</td>
</tr>
</tbody>
</table>

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 request message for this operation MAY contain the following HTTP headers.

<table>
<thead>
<tr>
<th>Request header</th>
<th>Usage</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-WOPI-RelativeTarget</td>
<td>A UTF-7 string that specifies the full file path and file name from the root folder. The WOPI server MUST NOT modify the name to fulfill the request.</td>
<td>A string that is a full file path and file name.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Status code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>Success.</td>
</tr>
<tr>
<td>404</td>
<td>Folder unknown/User unauthorized.</td>
</tr>
<tr>
<td>500</td>
<td>Server error.</td>
</tr>
<tr>
<td>501</td>
<td>Unsupported.</td>
</tr>
</tbody>
</table>

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:

```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.

When the X-WOPI-RelativeTarget header is supplied, only the target name will be returned by this endpoint. If omitted, all children for the root folder will be returned.

### 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 Additional WOPI Server Documentation

Additional documentation describing how to implement a WOPI server can be found at [WOPIOfficeOnline].

4.2 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?ui=UI_LLCC&amp;gt;&amp;rs=DC_LLCC&amp;amp;gt;&amp;showpagestats=PERFSTATS&amp;amp;gt;" /><action name="edit" ext="docx" requires="locks, cobalt, update" urlsrc="http://WOPIClient/wv/wordeditorframe.aspx?ui=UI_LLCC&amp;gt;&amp;rs=DC_LLCC&amp;amp;gt;&amp;showpagestats=PERFSTATS&amp;amp;gt;" /><action name="editnew" ext="docx" requires="locks, cobalt, update" urlsrc="http://WOPIClient/wv/wordeditorframe.aspx?new=1&amp;amp;ui=UI_LLCC&amp;gt;&amp;rs=DC_LLCC&amp;amp;gt;&amp;showpagestats=PERFSTATS&amp;amp;gt;" /><action name="imagepreview" ext="docx" urlsrc="http://WOPIClient/wv/WordPreviewHandler.ashx?ui=UI_LLCC&amp;amp;gt;&amp;rs=DC_LLCC&amp;amp;gt;&amp;showpagestats=PERFSTATS&amp;amp;gt;" /><action name="interactivepreview" ext="docx" urlsrc="http://WOPIClient/wv/wordviewerframe.aspx?embed=1&amp;amp;ui=UI_LLCC&amp;amp;gt;&amp;rs=DC_LLCC&amp;amp;gt;&amp;showpagestats=PERFSTATS&amp;amp;gt;" /><action name="mobileView" ext="docx" urlsrc="http://WOPIClient/wv/wordviewerframe.aspx?ui=UI_LLCC&amp;amp;gt;&amp;rs=DC_LLCC&amp;amp;gt;&amp;showpagestats=PERFSTATS&amp;amp;gt;" />
4.3 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 could introduce undesirable behaviors.

GetFile request from WOPI client:

GET http://WOPIClient/winResReader.ashx?n=p_1_10.xml&WOPIsrc=http://WOPIServer/~~vti_bin/wopi.ashx/files/cc8e6af1e0554661b71faeaa0d45fdd7&access_token=eyJ0eXAiOiJKV1QiLCJhbiBpZCI6IjAwMDAwMDAyLTAwMDAtMGZmM1VjOTM2NyIiLCJpYXQiOjE1MjcyNjIwMzEwMDU5fQ%3D%3D&v=...
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="",?;"> </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="",?;"> </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;,?;">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
<?xml version="1.0" encoding="utf-8"?>
<xs:schema id="Discovery"
  targetNamespace="http://tempuri.org/Discovery.xsd"
  elementFormDefault="qualified"
  xmlns:ms="http://tempuri.org/Discovery.xsd"
  xmlns:msxmlns="http://tempuri.org/Discovery.xsd"
  xmlns:xsi="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="embededit" />
    <xs:enumeration value="mobileclient" />
    <xs:enumeration value="viewNew" />
    <xs:enumeration value="present" />
    <xs:enumeration value="getinfo" />
    <xs:enumeration value="convert" />
    <xs:enumeration value="formsubmit" />
    <xs:enumeration value="formedit" />
    <xs:enumeration value="rest" />
    <xs:enumeration value="preloadview" />
    <xs:enumeration value="preloadedit" />
    <xs:enumeration value="rtc" />
    <xs:enumeration value="getinfo" />
    <xs:enumeration value="convert" />
    <xs:enumeration value="formsubmit" />
    <xs:enumeration value="formedit" />
    <xs:enumeration value="rest" />
    <xs:enumeration value="preloadview" />
    <xs:enumeration value="preloadedit" />
    <xs:enumeration value="rtc" />
    <xs:enumeration value="getinfo" />
    <xs:enumeration value="convert" />
    <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*(,\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="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="urisrc" 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>
```
<xs:attribute name="targetext" 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_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" default="false"/>
<xs:attribute name="checkLicense" type="xs:boolean" use="optional" default="false"/>
</xs:complexType>

<xs:complexType name="ct_proof-key">
<xs:attribute name="exponent" type="xs:string" use="required"/>
<xs:attribute name="modulus" type="xs:string" use="required"/>
<xs:attribute name="oldexponent" type="xs:string" use="required"/>
<xs:attribute name="oldmodulus" type="xs:string" use="required"/>
<xs:attribute name="oldvalue" 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.

<table>
<thead>
<tr>
<th>Schema name</th>
<th>Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>CheckFileInfo JSON</td>
<td>7.1</td>
</tr>
<tr>
<td>CheckFolderInfo JSON</td>
<td>7.2</td>
</tr>
<tr>
<td>EnumerateChildren JSON</td>
<td>7.3</td>
</tr>
<tr>
<td>PutRelativeFile JSON</td>
<td>7.4</td>
</tr>
</tbody>
</table>

7.1 CheckFileInfo JSON

```json
{
  "AllowAdditionalMicrosoftServices": {"type": "bool", "default": false, "optional": true},
  "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},
  "CloseButtonClosesWindow": {"type": "bool", "default": false, "optional": true},
  "ClosePostMessage": {"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},
  "EditModePostMessage": {"type": "bool", "default": false, "optional": true},
  "EditModePostMessage": {"type": "bool", "default": false, "optional": true},
  "FileExtension": {"type": "string", "default": "", "optional": true},
  "FileNameMaxLength": {"type": "integer", "default": 250, "optional": true},
  "FileSharingPostMessage": {"type": "bool", "default": false, "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},
  "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},
  "LicenseCheckForEditIsEnabled": {"type": "bool", "default": false, "optional": true},
  "OwnerId": {"type": "string", "optional": false},
  "PostMessageOrigin": {"type": "string", "default": "", "optional": true},
  "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},
  "SignInUrl": {"type": "string", "default": "", "optional": true},
  "SignoutUrl": {"type": "string", "default": "", "optional": true},
  "signInUrl": {"type": "string", "default": "", "optional": true},
  "signoutUrl": {"type": "string", "default": "", "optional": true},
}
```
7.2 CheckFolderInfo JSON

```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", "default": "", "optional": true},
    "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 updates to those products.

- Microsoft Lync Server 2013
- Microsoft Exchange Server 2013
- Microsoft Office Web Apps Server
- Microsoft SharePoint Server 2013
- Microsoft SharePoint Foundation 2013
- Office Online Server
- Microsoft Skype for Business Server 2015
- Microsoft SharePoint Server 2016
- Microsoft Skype for Business Server 2019
- Microsoft SharePoint Server 2019

Exceptions, if any, are noted in this section. If an update version, service pack or Knowledge Base (KB) number appears with a product name, the behavior changed in that update. The new behavior also applies to subsequent updates unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

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

<1> Section 3.3.5.1.1.2: SharePoint Server 2013 does not support the ClosePostMessage field.

<2> Section 3.3.5.1.1.2: SharePoint Server 2013 and SharePoint Server 2016 do not support the FileExtension field.

<3> Section 3.3.5.1.1.2: SharePoint Server 2013 and SharePoint Server 2016 do not provide the SHA256 field and set this field to empty.

<4> Section 3.3.5.1.1.2: SharePoint Server 2013 returns the value false for the SupportsCoauth field.

<5> Section 3.3.5.1.1.2: SharePoint Server 2013 and SharePoint Server 2016 do not support the SupportsExtendedLockLength field.

<6> Section 3.3.5.1.1.2: SharePoint Server 2013 does not support the SupportsFileCreation field.

<7> Section 3.3.5.1.1.2: SharePoint Server 2013 and SharePoint Server 2016 do not support the SupportsGetLock field.

<8> Section 3.3.5.1.1.2: SharePoint Server 2013 does not support the SupportsRename field.

<9> Section 3.3.5.1.1.2: SharePoint Server 2013 and SharePoint Server 2016 do not support the SupportsUserInfo field.

<10> Section 3.3.5.1.1.2: SharePoint Server 2013 and SharePoint Server 2016 do not provide the UniqueContentId field and set this field to empty.
<11> Section 3.3.5.1.5: SharePoint Server 2013 and SharePoint Server 2016 do not support the GetLock operation.

<12> Section 3.3.5.1.10: SharePoint Server 2013 and SharePoint Server 2016 do not support the PutUserInfo operation.

<13> Section 3.3.5.1.11: SharePoint Foundation 2013 does not support the ReadSecureStore operation.

<14> Section 3.3.5.1.13: SharePoint Server 2013 does not support the RenameFile operation.
9 Change Tracking

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

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

- A document revision that incorporates changes to interoperability requirements.
- A document revision that captures changes to protocol functionality.

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

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

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

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Revision class</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1 CheckFileInfo JSON</td>
<td>Added AllowAdditionalMicrosoftServices.</td>
<td>Minor</td>
</tr>
<tr>
<td>7.2 CheckFolderInfo JSON</td>
<td>Added FileSharingUrl.</td>
<td>Minor</td>
</tr>
</tbody>
</table>
10 Index

A
Abstract data model
  WOPI protocol server details 25
Accessing discovery XML example 56
Applicability 10

C
Capability negotiation 10
Change tracking 67
CheckFileInfo JSON 62
CheckFolderInfo JSON 63
Common URI parameters 14
Custom HTTP headers 12

D
Data model – abstract
  WOPI protocol server details 25

E
EnumerateChildren JSON 64
Examples
  accessing discovery XML 56
  Accessing Discovery XML example 56
  Additional WOPI Server Documentation example
    56
  overview 56
  viewing a document 57
  Viewing a Document example 57

F
Fields – vendor-extensible 10
Full JSON schema 62
  CheckFileInfo 62
  CheckFolderInfo 63
  EnumerateChildren 64
  PutRelativeFile 64
Full XML schema 60

G
Glossary 6

H
Headers
  custom HTTP 12

I
Implementer – security considerations 59
  Index of security parameters 59
  Informative references 8
  Introduction 6

J
JSON schema 62
  CheckFileInfo 62
  CheckFolderInfo 63
  EnumerateChildren 64
  PutRelativeFile 64

M
Message processing
  WOPI discovery details 15
  WOPI protocol server details 25
Messages
  syntax 12
  transport 12

N
Normative references 7

O
Overview (synopsis) 8

P
Parameters
  common URI 14
Parameters – security index 59
  Preconditions 10
  Prerequisites 10
  Product behavior 65
Protocol Details
  WOPI Discovery 15
  WOPI Protocol Client 24
  WOPI Protocol Server 24
Protocol examples
  Accessing Discovery XML 56
  Additional WOPI Server Documentation 56
  Viewing a Document 57
  PutRelativeFile JSON 64

R
References
  informative 8
  normative 7
  Relationship to other protocols 10

S
Security
  implementer considerations 59
  parameter index 59
Standards assignments 10
Syntax
  messages – overview 12

T
Tracking changes 67