

[MS-WDVSE]:

Web Distributed Authoring and Versioning (WebDAV) Protocol: Server Extensions

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation (“this documentation”) for protocols, file formats, data portability, computer languages, and standards support. Additionally, overview documents cover inter-protocol relationships and interactions.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you can make copies of it in order to develop implementations of the technologies that are described in this documentation and can distribute portions of it in your implementations that use these technologies or in your documentation as necessary to properly document the implementation. You can also distribute in your implementation, with or without modification, any schemas, IDLs, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications documentation.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that might cover your implementations of the technologies described in the Open Specifications documentation. Neither this notice nor Microsoft's delivery of this documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open Specifications document might be covered by the Microsoft [Open Specifications Promise](#) or the [Microsoft Community Promise](#). If you would prefer a written license, or if the technologies described in this documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **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

Date	Revision History	Revision Class	Comments
7/20/2007	0.1	Major	MCPP Milestone 5 Initial Availability
9/28/2007	0.2	Minor	Clarified the meaning of the technical content.
10/23/2007	0.2.1	Editorial	Changed language and formatting in the technical content.
11/30/2007	0.2.2	Editorial	Changed language and formatting in the technical content.
1/25/2008	0.2.3	Editorial	Changed language and formatting in the technical content.
3/14/2008	0.2.4	Editorial	Changed language and formatting in the technical content.
5/16/2008	0.2.5	Editorial	Changed language and formatting in the technical content.
6/20/2008	1.0	Major	Updated and revised the technical content.
7/25/2008	1.0.1	Editorial	Changed language and formatting in the technical content.
8/29/2008	1.0.2	Editorial	Changed language and formatting in the technical content.
10/24/2008	1.0.3	Editorial	Changed language and formatting in the technical content.
12/5/2008	1.0.4	Editorial	Changed language and formatting in the technical content.
1/16/2009	2.0	Major	Updated and revised the technical content.
2/27/2009	3.0	Major	Updated and revised the technical content.
4/10/2009	3.0.1	Editorial	Changed language and formatting in the technical content.
5/22/2009	3.0.2	Editorial	Changed language and formatting in the technical content.
7/2/2009	4.0	Major	Updated and revised the technical content.
8/14/2009	4.0.1	Editorial	Changed language and formatting in the technical content.
9/25/2009	4.1	Minor	Clarified the meaning of the technical content.
11/6/2009	4.1.1	Editorial	Changed language and formatting in the technical content.
12/18/2009	4.1.2	Editorial	Changed language and formatting in the technical content.
1/29/2010	4.1.3	Editorial	Changed language and formatting in the technical content.
3/12/2010	4.1.4	Editorial	Changed language and formatting in the technical content.
4/23/2010	4.1.5	Editorial	Changed language and formatting in the technical content.
6/4/2010	4.1.6	Editorial	Changed language and formatting in the technical content.
7/16/2010	4.1.6	None	No changes to the meaning, language, or formatting of the technical content.
8/27/2010	4.1.6	None	No changes to the meaning, language, or formatting of the technical content.
10/8/2010	4.1.6	None	No changes to the meaning, language, or formatting of the technical content.
11/19/2010	4.1.6	None	No changes to the meaning, language, or formatting of the

Date	Revision History	Revision Class	Comments
			technical content.
1/7/2011	5.0	Major	Updated and revised the technical content.
2/11/2011	5.0	None	No changes to the meaning, language, or formatting of the technical content.
3/25/2011	5.0	None	No changes to the meaning, language, or formatting of the technical content.
5/6/2011	5.0	None	No changes to the meaning, language, or formatting of the technical content.
6/17/2011	5.1	Minor	Clarified the meaning of the technical content.
9/23/2011	6.0	Major	Updated and revised the technical content.
12/16/2011	7.0	Major	Updated and revised the technical content.
3/30/2012	7.0	None	No changes to the meaning, language, or formatting of the technical content.
7/12/2012	7.0	None	No changes to the meaning, language, or formatting of the technical content.
10/25/2012	7.0	None	No changes to the meaning, language, or formatting of the technical content.
1/31/2013	7.1	Minor	Clarified the meaning of the technical content.
8/8/2013	8.0	Major	Updated and revised the technical content.
11/14/2013	8.1	Minor	Clarified the meaning of the technical content.
2/13/2014	8.1	None	No changes to the meaning, language, or formatting of the technical content.
5/15/2014	8.1	None	No changes to the meaning, language, or formatting of the technical content.
6/30/2015	9.0	Major	Significantly changed the technical content.
10/16/2015	9.0	None	No changes to the meaning, language, or formatting of the technical content.
7/14/2016	9.0	None	No changes to the meaning, language, or formatting of the technical content.
6/1/2017	9.0	None	No changes to the meaning, language, or formatting of the technical content.

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	8
1.5	Prerequisites/Preconditions	8
1.6	Applicability Statement	8
1.7	Versioning and Capability Negotiation	9
1.8	Vendor-Extensible Fields	9
1.9	Standards Assignments.....	9
2	Messages.....	10
2.1	Transport.....	10
2.2	Message Syntax.....	10
2.2.1	Translate Request Header.....	10
2.2.2	MS-Author-Via Response Header	11
2.2.3	Depth "noroot" Request Header Extension	12
2.2.4	SEARCH Method	12
2.2.5	Deviations from RFC4918	12
2.2.5.1	Additional "Live" Properties	12
2.2.5.2	Collections and Locking.....	12
2.2.5.3	Multiple Property Elements.....	13
3	Protocol Details	14
3.1	WebDAV Client Details.....	14
3.1.1	Abstract Data Model.....	14
3.1.2	Timers	14
3.1.3	Initialization.....	14
3.1.4	Higher-Layer Triggered Events	14
3.1.5	Processing Events and Sequencing Rules	14
3.1.5.1	Translate Request Header	14
3.1.5.2	MS-Author-Via Response Header.....	14
3.1.5.3	Depth "noroot" Request Header Extension.....	15
3.1.5.4	SEARCH Method.....	15
3.1.6	Timer Events.....	15
3.1.7	Other Local Events.....	15
3.2	WebDAV Server Details.....	15
3.2.1	Abstract Data Model.....	15
3.2.2	Timers	15
3.2.3	Initialization.....	15
3.2.4	Higher-Layer Triggered Events	15
3.2.5	Processing Events and Sequencing Rules	15
3.2.5.1	Translate Request Header	16
3.2.5.2	MS-Author-Via Response Header.....	16
3.2.5.3	Depth "noroot" Request Header Extension.....	16
3.2.5.4	SEARCH Method.....	16
3.2.6	Timer Events.....	16
3.2.7	Other Local Events.....	16
4	Protocol Examples	17
4.1	MS-Author-Via Response Header.....	17
4.2	Translate Request Header	17
4.3	Depth "noroot" Request Header Extension.....	18

5	Security	21
5.1	Security Considerations for Implementers	21
5.2	Index of Security Parameters	21
6	Appendix A: Product Behavior	22
7	Change Tracking	24
8	Index	25

1 Introduction

The web-based Distributed Authoring and Versioning (WebDAV) Protocol defined in [\[RFC4918\]](#) extends the standard Hypertext Transfer Protocol (HTTP) mechanisms defined in [\[RFC2068\]](#) to provide file access and content management over the Internet. The WebDAV Protocol enables an Internet-based file system. However, some types of files (for example, files with programmatically derived content) are not easily managed by WebDAV and some protocol interactions are suboptimal for file system usage.

This specification extends WebDAV by introducing new HTTP request and response headers that both enable the file types that are not currently manageable and optimize protocol interactions for file system clients. This specification also introduces a new WebDAV method that is used to send search queries to disparate search providers.

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:

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 [\[RFC5234\]](#).

collection: A resource that contains a set of URIs that identify member resources. Use of this term is consistent with what is specified in [\[RFC4918\]](#) section 5.2.

entity: Any document on a server that is accessible by using a **Hypertext Transfer Protocol (HTTP)** URL.

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 [\[RFC5246\]](#).

locking: A mechanism that is used for overwrite protection. **Locking** can be applied to individual resources or to entire collection hierarchies (see [\[RFC4918\]](#) sections 6 and 7).

property: A name/value pair that associates metadata with a resource. This term is used as specified in [\[RFC4918\]](#) section 4.

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 uses two keys to encrypt data—a public key known to everyone and a private or secret key known only to the recipient of the message. SSL supports server and, optionally, client authentication using X.509 certificates. For more information, see [\[X509\]](#). The SSL protocol is precursor to **Transport Layer Security (TLS)**. The TLS version 1.0 specification is based on SSL version 3.0 [\[SSL3\]](#).

Transport Layer Security (TLS): A security protocol that supports confidentiality and integrity of messages in client and server applications communicating over open networks. **TLS** supports

server and, optionally, client authentication by using X.509 certificates (as specified in [X509]). **TLS** is standardized in the IETF TLS working group.

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

Web Distributed Authoring and Versioning Protocol (WebDAV): The Web Distributed Authoring and Versioning Protocol, as described in [\[RFC2518\]](#) or [RFC4918].

web server: A server computer that hosts websites and responds to requests from applications.

WebDAV client: A computer that uses **WebDAV**, as described in [RFC2518] or [RFC4918], to retrieve data from a **WebDAV server**.

WebDAV server: A computer that supports **WebDAV**, as described in [RFC2518] or [RFC4918], and responds to requests from **WebDAV clients**.

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.

[RFC2068] Fielding, R., Gettys, J., Mogul, J., et al., "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2068, January 1997, <http://www.ietf.org/rfc/rfc2068.txt>

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

[RFC2246] Dierks, T., and Allen, C., "The TLS Protocol Version 1.0", RFC 2246, January 1999, <http://www.rfc-editor.org/rfc/rfc2246.txt>

[RFC2291] Slein, J., Vitali, F., Whitehead, E., et al., "Requirements for a Distributed Authoring and Versioning Protocol for the World Wide Web", RFC 2291, February 1998, <http://www.rfc-editor.org/rfc/rfc2291.txt>

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

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

[RFC4234] Crocker, D., Ed., and Overell, P., "Augmented BNF for Syntax Specifications: ABNF", RFC 4234, October 2005, <http://www.rfc-editor.org/rfc/rfc4234.txt>

[RFC4918] Dusseault, L, Ed., "HTTP Extensions for Web Distributed Authoring and Versioning (WebDAV)", RFC 4918, June 2007, <http://www.rfc-editor.org/rfc/rfc4918.txt>

1.2.2 Informative References

[MSASP] Microsoft Corporation, "Active Server Pages", <http://msdn.microsoft.com/en-us/library/aa286483.aspx>

[MSDN-FP] Microsoft Corporation, "FrontPage", [http://msdn.microsoft.com/en-us/library/aa167865\(office.11\).aspx](http://msdn.microsoft.com/en-us/library/aa167865(office.11).aspx)

1.3 Overview

WebDAV is a set of methods, headers, and content types that extend the HTTP 1.1 Protocol, as specified in [\[RFC2068\]](#). WebDAV allows data to be written to Internet servers and is an Internet standard for collaborative authoring, as specified in [\[RFC4918\]](#).

WebDAV expands the basic support in HTTP 1.1 for content authoring by introducing additional methods and headers to provide support for resource **properties** and other base functions, such as resource **locking**. These new capabilities make the WebDAV Protocol suitable for basic remotely mountable file systems. A resource is an entity that can be identified by a URI. A resource is used as specified in [\[RFC2616\]](#) section 1.3.

This document specifies the following extensions to the base WebDAV Protocol, as specified in [\[RFC4918\]](#).

- An **HTTP** method that serves as the transport vehicle for search functionality in external processes.
- An HTTP request header that indicates whether an **entity** is to be returned as-is, or whether any associated programmatic processing is to be performed and the result returned.
- The [MS-Author-Via Response header](#), which specifies to the client application what the preferred protocol mechanism is for authoring documents in a particular namespace. The namespace is the entire collection (as specified in [\[RFC4918\]](#) section 5.2) of items under a request URI. For example, a client might have multiple authoring tools available to it, each possibly supported by a different network authoring protocol.
- An extension to the existing Depth HTTP request header that indicates whether the root of a resource is to be included in the results.
- Additional properties in the DAV namespace that describe additional file characteristics.

1.4 Relationship to Other Protocols

The WebDAV Protocol: Server Extensions rely on **WebDAV** which, in turn, relies on HTTP 1.1 as defined in [\[RFC2068\]](#). These extensions can use **HTTPS** for data protection services, as defined in [\[RFC2818\]](#).

1.5 Prerequisites/Preconditions

This specification requires a **WebDAV server**, as defined in [\[RFC2291\]](#), that supports the OPTIONS command.

This specification also requires that **WebDAV clients** have **URLs** that point to WebDAV servers. The client has to obtain the URLs through some out-of-band mechanism.

1.6 Applicability Statement

WebDAV Protocol: Server Extensions applies in scenarios that require efficient file operations. Note that this document specifies only those extensions specific to enabling efficient file system clients.

These extensions do not add any functionality. They instead help reduce the network traffic and increase the performance of clients that use the **WebDAV** Protocol.

1.7 Versioning and Capability Negotiation

Supported Transports: WebDAV Protocol: Server Extensions use **HTTP** as the only transport.

Versioning: No new versioning mechanisms have been introduced beyond those that already exist in **WebDAV** and HTTP, as specified in [\[RFC4918\]](#) and [\[RFC2068\]](#).

Capability Negotiation: Negotiation of WebDAV (as specified in [\[RFC4918\]](#) sections 10.1 and 18) and HTTP capabilities (as specified in [\[RFC2068\]](#) section 9.2) is via the OPTIONS method. This specification extends the OPTIONS method that uses an HTTP response header to indicate which authoring tools are to be used. For more information about authoring, see section [2.2.2](#).

1.8 Vendor-Extensible Fields

This protocol can be extended by adding new tokens to the [MS-Author-Via](#) field value (see section 2.2.2).

This protocol uses **HTTP** status codes as defined in [\[RFC2068\]](#) section 10 and [\[RFC4918\]](#) section 11.

1.9 Standards Assignments

No standards body has approved or governs this document or its header names, values, methods, and deviations. This specification conforms to the form and behavior of other custom **HTTP** headers, as specified in [\[RFC2068\]](#) section 4.2.

2 Messages

The following section describes transport requirements and the syntax of the WebDAV Protocol: Server Extensions.

2.1 Transport

Messages are transported using **HTTP** as specified in [\[RFC4918\]](#) and [\[RFC2068\]](#).

This protocol can be used with **Secure Sockets Layer (SSL)** or **Transport Layer Security (TLS)**, as specified in [\[RFC2246\]](#).

Port 80 is the standard port assignment for HTTP, and port 443 is the standard port assignment for HTTP over SSL or TLS; however, individual implementations can support other ports.

2.2 Message Syntax

The extension headers in this protocol conform to the form and behavior of other custom **HTTP** headers, as specified in [\[RFC2068\]](#) section 4.2, and are consistent with the **WebDAV** verbs and headers as defined in [\[RFC4918\]](#) sections 9 and 10.

This section specifies the following header extensions:

- The [Translate request header \(section 2.2.1\)](#) allows the **WebDAV client** to request the source of an **entity**.
- The header field [MS-Author-Via](#) is returned as a response-header field to a client that has issued an **OPTIONS** command to the server. The syntax of the **OPTIONS** command is specified in [\[RFC2616\]](#) section 9.2. The syntax of the MS-Author-Via header is specified using the **Augmented Backus-Naur Form (ABNF)**, as defined in [\[RFC2616\]](#) section 2.1.
- The "[noroot \(section 2.2.3\)](#)" extension to the Depth request header extends the functionality that is specified in [\[RFC4918\]](#) section 10.2, allowing a WebDAV client to request the children of a requested entity, but not to include the requested entity itself.

This section specifies the following method:

- The [SEARCH method \(section 2.2.4\)](#) is used to transport search-related commands to external search providers.

This section specifies the following deviations from [\[RFC4918\]](#) in the **WebDAV server** implementation:

- WebDAV **property** retrieval, as specified in [\[RFC4918\]](#) section 9.1, can include additional "live" property settings, as specified in section [2.2.5.1](#).
- WebDAV **locks**, as specified in [\[RFC4918\]](#) sections 6 and 7, might not be supported on WebDAV **collections**, as specified in section [2.2.5.2](#) and in [\[RFC4918\]](#) section 7.4.
- The WebDAV property setting, as specified in [\[RFC4918\]](#) section 9.2, can include multiple un-nested property settings, as specified in section [2.2.5.3](#).

2.2.1 Translate Request Header

Many resources obtained from a **WebDAV server** are returned exactly as-is. However, some resources are programmatically interpreted by the **web server** and the result of that interpretation is returned instead of the source representation. For instance, a request to retrieve an Active Server Page (ASP) (as described in [\[MSASP\]](#)) from the web server would return the processed HTML file

rather than the actual source of the ASP page. A resource is an entity that can be identified by a URI. A resource is used as specified in [\[RFC2616\]](#) section 1.3.

In order for a **WebDAV client** to indicate the required representation, WebDAV Protocol: Server Extensions introduces a new Translate request header. This header requests that the web server perform "translation" (programmatic interpretation) of the file.

This new request header is defined as follows, using the ABNF syntax, as specified in [\[RFC2068\]](#) section 2.1.

```
Translate-header = "Translate" ":" Translate-value
Translate-value = "t" / "f" / "F"
```

If the Translate value is "t", the web server is to process the content before returning it to the WebDAV client; if "f" | "F", the web server is to return the unprocessed (or source) content to the WebDAV client. The values "f" and "F" are synonymous. All other values SHOULD be ignored by the web server. For more details about processing of this request header, see section [3.1.5.<1>](#)

This request header MUST be supported on the GET verb and MAY be supported on other verbs. [<2>](#)

2.2.2 MS-Author-Via Response Header

This extension MUST have the following format:

```
MS-Author-Via = "MS-Author-Via" ":" ("MS-FP/4.0" / "MS-FP/4.0,DAV" / "DAV" / "DAV,MS-FP/4.0" / token ",DAV")
```

This header field indicates to the issuer of an **HTTP OPTIONS** command what protocol mechanism is preferred for authoring documents in this particular namespace. The preference MUST be ordered so the first mechanism listed is the one most preferred by the server.

Note The "token" definition in the preceding grammar is specified in [\[RFC2616\]](#) section 2.2. Its purpose in the grammar is to allow for future extensibility. The WebDAV Server Extensions protocol MAY be extended by adding new tokens to the MS-Author-Via field. Clients MUST ignore tokens in this field that they do not recognize.

The following table lists the values that are currently defined for this header.

Value	Meaning
MS-FP/4.0	The server's preferred method for authoring is the FrontPage Server Extensions.
MS-FP/4.0,DAV	The server's protocol preference for authoring is first the FrontPage Server Extensions (for more information, see [MSDN-FP]), and then WebDAV .
DAV	The server's preferred method for authoring is WebDAV.
DAV,MS-FP/4.0	The server's protocol preference for authoring is first WebDAV, and then the FrontPage Server Extensions (for more information, see [MSDN-FP]).
Token,DAV	The server's protocol preference for authoring is an extensible token, and then WebDAV.

- MS-FP/4.0 refers to Microsoft FrontPage Server protocol version 4.0 enabled on the **web server**. Some applications look for this string and use that protocol to communicate with this web server.
- DAV is used to indicate that WebDAV is enabled on the **WebDAV server**, and a **WebDAV client** MAY use WebDAV commands to communicate with this WebDAV server.

2.2.3 Depth "noroot" Request Header Extension

This header extension is defined as follows, using the ABNF syntax defined in [\[RFC4234\]](#) section 2.1:

```
Depth = "Depth" ":" " ("0" / "1" / "infinity" / "1,noroot" / "infinity,noroot")
```

The following values are associated with the Depth header.

Value	Meaning
0	The command applies only to the specified resource.
1	The command applies to the specified resource and the next level of resources that it contains.
infinity	The command applies to the specified resource and all the resources that it contains.
1,noroot	The command applies to the next level of resources in the container but not to the container itself.
infinity,noroot	The command applies to all the resources in the container but not to the container itself.

The noroot extension is present if the "1,noroot" value is set or if the "infinity,noroot" value is set. [<3>](#)

2.2.4 SEARCH Method

This method extension allows a **WebDAV client** to request search results from an external search provider. A **WebDAV server** SHOULD advertise support for the SEARCH method in the response to an OPTIONS request. The SEARCH method is simply a delivery mechanism for passing queries to external search providers. [<4>](#)

The following example shows the syntax for the SEARCH method, which conforms to the **HTTP** syntax as defined in [\[RFC2068\]](#).

```
SEARCH /[resource] HTTP/1.1
Host: www.example.com
Content-type: text/xml; charset="utf-8"
Content-Length: nnn
<?xml version="1.0" encoding="utf-8" ?>
<D:searchrequest xmlns:D = "DAV:">
  <D:sql>
    [implementation-specific content for a search provider]
  </D:sql>
</D:searchrequest>
```

2.2.5 Deviations from RFC4918

2.2.5.1 Additional "Live" Properties

[\[RFC4918\]](#) sections 4 and 15 define the "live" **properties** that a **WebDAV server** SHOULD implement. [<5>](#)

2.2.5.2 Collections and Locking

[\[RFC4918\]](#) sections 7 and 9 define how **WebDAV servers** SHOULD implement **locking** for **collections**. [<6>](#)

2.2.5.3 Multiple Property Elements

[RFC4918] section 9.2 defines the message syntax for setting **properties** by using the PROPPATCH method. A property or group of properties for a resource MAY be enclosed within a single <d:prop></d:prop> element, <7> as shown in the following example.

```
PROPPATCH /[resource] HTTP/1.1
Host: www.example.com
Content-type: text/xml; charset="utf-8"
Content-Length: nnn
<?xml version="1.0" encoding="utf-8" ?>
<d:propertyupdate xmlns:d='DAV:'>
  <d:set>
    <d:prop>
      <d:owner>Owner's Name</d:owner>
    </d:prop>
  </d:set>
</d:propertyupdate>
```

3 Protocol Details

As specified in [\[RFC4918\]](#), **WebDAV** operates between an initiator (a **WebDAV client**) and a responder (a **WebDAV server**). This section specifies the client and the server behaviors with respect to the WebDAV extensions.

3.1 WebDAV Client Details

3.1.1 Abstract Data Model

A **WebDAV client** SHOULD maintain a variable for each server with which it communicates that contains the value of the [MS-Author-Via response header](#).

3.1.2 Timers

No new timers are required beyond those specified in [\[RFC2068\]](#).

3.1.3 Initialization

Prior to general interaction with a **WebDAV server**, the **WebDAV client** SHOULD perform an OPTIONS request to determine the current authoring implementation using the [MS-Author-Via response header](#).[<8>](#)

3.1.4 Higher-Layer Triggered Events

No new events are triggered beyond those specified in [\[RFC2068\]](#).

3.1.5 Processing Events and Sequencing Rules

A **WebDAV client** SHOULD send an OPTIONS request to the server and query the response headers for the presence of the **WebDAV server's** supported features. When sending an OPTIONS request, the WebDAV client SHOULD send the OPTIONS request before sending any other requests to the WebDAV server. The information returned SHOULD be used by the WebDAV client to specialize requests to this server.

3.1.5.1 Translate Request Header

The **WebDAV client** SHOULD add the [Translate request header](#) with an "f" or "F" flag to a request if the WebDAV client needs the source of a file without any translation; otherwise this header SHOULD be omitted.

3.1.5.2 MS-Author-Via Response Header

A **WebDAV client** MAY send an OPTIONS request to the server and query the response headers for the presence of the MS-Author-Via header. The WebDAV client SHOULD use the following information to determine the best method to send requests to the server:

- The WebDAV client SHOULD NOT use **WebDAV** commands when communicating with a server that responds to the OPTIONS command with only the MS-Author-Via: MS-FP/4.0 response header.
- The WebDAV client MAY use WebDAV commands when communicating with a server that responds to the OPTIONS command with any of the following [MS-Author-Via response header](#) values:

- MS-Author-Via: DAV
- MS-Author-Via: DAV, MS-FP/4.0
- MS-Author-Via: MS-FP/4.0, DAV
- MS-Author-Via: MS-FP
- MS-Author-Via: Token,DAV

3.1.5.3 Depth "noroot" Request Header Extension

The **WebDAV client** SHOULD add the [Depth request header](#) with a value of '1,noroot' or 'infinity,noroot' to a request if the WebDAV client needs the resources that are in the container, but not in the container itself; otherwise this header SHOULD be omitted. <9> A resource is an entity that can be identified by a URI. A resource is used as specified in [\[RFC2616\]](#) section 1.3.

3.1.5.4 SEARCH Method

A **WebDAV client** SHOULD send a SEARCH request to a **WebDAV server** that advertises support for the SEARCH method through the response to an OPTIONS request from the WebDAV client. If the WebDAV server does not advertise support for the SEARCH method, the WebDAV client SHOULD NOT send a SEARCH request.

3.1.6 Timer Events

No new timers are required beyond those in the base protocol.

3.1.7 Other Local Events

There are no new local events other than those described in the base protocol.

3.2 WebDAV Server Details

3.2.1 Abstract Data Model

No new abstract data model is needed other than that described in the base protocol.

3.2.2 Timers

No new timers are required beyond those in the base protocol.

3.2.3 Initialization

No initialization is required beyond that in the base protocol.

3.2.4 Higher-Layer Triggered Events

No new events are triggered beyond that in the base protocol.

3.2.5 Processing Events and Sequencing Rules

A **WebDAV server** can advertise support for **WebDAV** authoring by adding the MS-Author-Via: DAV response header to the response to the OPTIONS command. If the server does not support WebDAV

authoring, the OPTIONS response SHOULD omit the MS-Author-Via: DAV response header. A WebDAV server MAY add the MS-Author-Via: DAV header to other responses.

3.2.5.1 Translate Request Header

The **WebDAV server** MUST respond with the actual source of the file if the translate request header exists with a flag value starting with "f" or "F". The WebDAV server SHOULD perform access checks when processing a command with a [Translate request header.<10>](#)

To maintain consistency with web browsers, the default behavior if this request header is omitted is to translate the file; omission of this header SHOULD be the same as sending Translate: t.<11>

3.2.5.2 MS-Author-Via Response Header

The **WebDAV server** SHOULD respond with the [MS-Author-Via response header](#) when a **WebDAV client** sends an OPTIONS request.<12> The WebDAV server responds with the appropriate information when the following conditions are true:

- "MS-Author-Via: DAV" The WebDAV server responds with this header when **WebDAV** is enabled and the FrontPage Server Extensions protocol is not enabled.
- "MS-Author-Via: MS-FP/4.0" The **web server** responds with this header when the FrontPage Server Extensions protocol is enabled and WebDAV is not enabled.
- "MS-Author-Via: DAV,MS-FP/4.0" The WebDAV server responds with this header when both WebDAV and the FrontPage Server Extensions protocol are enabled, and WebDAV is preferred over the FrontPage Server Extensions protocol.
- "MS-Author-Via: MS-FP/4.0,DAV" The WebDAV server responds with this header when both the FrontPage Server Extensions protocol and WebDAV are enabled, and the FrontPage Server Extensions protocol is preferred over WebDAV.
- "MS-Author-Via: Token,DAV" The WebDAV server responds with this header when both the WebDAV and an extensible token are enabled.

3.2.5.3 Depth "noroot" Request Header Extension

If the **WebDAV server** supports the [Depth "noroot" request header extension](#), the WebDAV server SHOULD respond appropriately depending on the **WebDAV** command that the **WebDAV client** is sending.<13>

3.2.5.4 SEARCH Method

A **WebDAV server** advertises support for the SEARCH method through the response to an OPTIONS request from the **WebDAV client**. If a WebDAV client sends a SEARCH request, the WebDAV server SHOULD send the request to the appropriate search provider on the server.<14>

3.2.6 Timer Events

No new timers are required beyond those in the base protocol.

3.2.7 Other Local Events

There are no new local events other than those described in the base protocol.

4 Protocol Examples

This section provides examples of the protocol extensions.

4.1 MS-Author-Via Response Header

The following example shows an OPTIONS request from a **WebDAV client** and the response from the **WebDAV server** that contains the [MS-Author-Via response header](#).

Request:

```
OPTIONS / HTTP/1.1
Host: localhost
Accept: */*
```

Response:

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/6.0
Date: Wed, 28 Jun 2006 00:06:21 GMT
MS-Author-Via: DAV
Allow: OPTIONS, TRACE, GET, HEAD, POST, LOCK, UNLOCK, MKCOL,
      PROPFIND, PROPPATCH, COPY, MOVE
Public: OPTIONS, TRACE, GET, HEAD, POST, LOCK, UNLOCK, MKCOL,
       PROPFIND, PROPPATCH, COPY, MOVE
MS-Author-Via: DAV
Content-Length: 0
```

4.2 Translate Request Header

The following examples show the difference between requesting an **entity** and the source of an entity. The first example is a typical **HTTP** GET command as issued by a browser such as Internet Explorer.

Request:

```
GET /Temp/world.asp HTTP/1.1
Translate: t
Host: localhost
Accept: */*
```

Response:

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/5.1
Date: Wed, 28 Jun 2006 00:06:21 GMT
Content-Length: 129
Content-Type: text/html
Set-Cookie: ASPSESSIONIDCSSTSCQB=IEEJDPNAAIJECIOBLMMGDJM; path=/
Cache-control: private

<FONT SIZE="1">Hello World</FONT><BR>
<FONT SIZE="2">Hello World</FONT><BR>
<FONT SIZE="3">Hello World</FONT><BR>
```

An authoring application might want to retrieve the source of an entity, and it would issue the same request asking for the source of the entity as follows:

Request:

```
GET /Temp/world.asp HTTP/1.1
Translate: f
Host: localhost
Accept: */*
```

Response:

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/5.1
Date: Wed, 28 Jun 2006 00:16:34 GMT
Content-Type: text/plain
Content-Length: 497
ETag: "22a87614489ac61:c02"
Last-Modified: Wed, 28 Jun 2006 00:16:19 GMT
Accept-Ranges: bytes
<%
!*****
!*      Sample ASP Code      *
!*      *                      *
!*****

' Declare loop variable.
Dim I

' Loop 3 times, adjusting the
' font size in each loop.
For I = 1 To 3 Step 1
    ' Output our HTML and text using the value
    ' of I as the FONT TAG's SIZE attribute.
%>
<FONT SIZE="<%=I%>">Hello World</FONT><BR>
<%
Next ' continue looping
%>
```

The difference between these two requests is that the second one is requesting the source of an entity. This is a typical example of how the [Translate request header](#) is used.

4.3 Depth "noroot" Request Header Extension

The following example shows a PROPFIND request from a **WebDAV client** and the response from the **WebDAV server** that contains the [Depth "noroot" Request Header Extension](#).

PROPFIND request with Depth: 1

```
PROPFIND /dav/ HTTP/1.1
Depth: 1
Content-Type: text/xml
Host: localhost
Accept: */*
Content-Length:104
```

```
<?xml version="1.0"?>
<D:propfind xmlns:D="DAV:">
  <D:prop>
    <D:displayname/>
  </D:prop>
</D:propfind>
```

Response:

```
HTTP/1.1 207 Multi-Status
Date: Fri, 17 Aug 2007 20:24:20 GMT
Server: Microsoft-IIS/6.0
X-Powered-By: ASP.NET
Content-Type: text/xml
Transfer-Encoding: chunked
```

```
<?xml version="1.0"?>
<a:multistatus xmlns:b="urn:uuid:c2f41010-65b3-11d1-a29f-00aa00c14882/"
xmlns:c="xml:" xmlns:a="DAV:">
  <a:response>
    <a:href>http://localhost/dav/</a:href>
    <a:propstat>
      <a:status>HTTP/1.1 200 OK</a:status>
      <a:prop>
        <a:displayname>dav</a:displayname>
      </a:prop>
    </a:propstat>
  </a:response>
  <a:response>
    <a:href>http://localhost/dav/pagerror.gif</a:href>
    <a:propstat>
      <a:status>HTTP/1.1 200 OK</a:status>
      <a:prop>
        <a:displayname>pagerror.gif</a:displayname>
      </a:prop>
    </a:propstat>
  </a:response>
  <a:response>
    <a:href>http://localhost/dav/script.asp</a:href>
    <a:propstat>
      <a:status>HTTP/1.1 200 OK</a:status>
      <a:prop>
        <a:displayname>script.asp</a:displayname>
      </a:prop>
    </a:propstat>
  </a:response>
  <a:response>
    <a:href>http://localhost/dav/textfile.txt</a:href>
    <a:propstat>
      <a:status>HTTP/1.1 200 OK</a:status>
      <a:prop>
        <a:displayname>textfile.txt</a:displayname>
      </a:prop>
    </a:propstat>
  </a:response>
</a:multistatus>
```

PROPFIND request with Depth: 1,noroot

```
PROPFIND /dav/ HTTP/1.1
Depth: 1,noroot
Content-Type: text/xml
Host: localhost
Accept: */*\r\n
```

Content-Length:104

```
<?xml version="1.0"?>
<D:propfind xmlns:D="DAV:">
  <D:prop>
    <D:displayname/>
  </D:prop>
</D:propfind>
```

Response:

```
HTTP/1.1 207 Multi-Status
Date: Fri, 17 Aug 2007 20:28:15 GMT
Server: Microsoft-IIS/6.0
X-Powered-By: ASP.NET
Content-Type: text/xml
Transfer-Encoding: chunked
```

```
<?xml version="1.0"?>
<a:multistatus xmlns:b="urn:uuid:c2f41010-65b3-11d1-a29f-00aa00c14882/"
xmlns:c="xml:" xmlns:a="DAV:">
  <a:response>
    <a:href>http://localhost/dav/pagerror.gif</a:href>
    <a:propstat>
      <a:status>HTTP/1.1 200 OK</a:status>
      <a:prop>
        <a:displayname>pagerror.gif</a:displayname>
      </a:prop>
    </a:propstat>
  </a:response>
  <a:response>
    <a:href>http://localhost/dav/script.asp</a:href>
    <a:propstat>
      <a:status>HTTP/1.1 200 OK</a:status>
      <a:prop>
        <a:displayname>script.asp</a:displayname>
      </a:prop>
    </a:propstat>
  </a:response>
  <a:response>
    <a:href>http://localhost/dav/textfile.txt</a:href>
    <a:propstat>
      <a:status>HTTP/1.1 200 OK</a:status>
      <a:prop>
        <a:displayname>textfile.txt</a:displayname>
      </a:prop>
    </a:propstat>
  </a:response>
</a:multistatus>
```

5 Security

5.1 Security Considerations for Implementers

WebDAV servers that support the [Translate request header](#) can perform access checks before returning the source of the file, as specified in section [3.2.5.1](#), in order to protect any source content (for example, database passwords).<15>

5.2 Index of Security Parameters

No new security parameters are required beyond those in the base protocol.

6 Appendix A: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs.

- Windows 2000 operating system
- Windows XP operating system
- Windows Server 2003 operating system
- Windows Vista operating system
- Windows Server 2008 operating system
- Windows 7 operating system
- Windows Server 2008 R2 operating system
- Windows 8 operating system
- Windows Server 2012 operating system
- Microsoft SharePoint Foundation 2010
- Windows 8.1 operating system
- Windows Server 2012 R2 operating system
- Windows 10 operating system
- Windows Server 2016 operating system

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

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

[<1> Section 2.2.1](#): The **web server** in IIS for Windows 2000, Windows XP, Windows Server 2003, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016 can accept the [Translate request header](#). This implementation accepts "f" and "F" as FALSE. Everything else, including omission of the header, is accepted as TRUE.

[<2> Section 2.2.1](#): The web server in IIS for Windows 2000, Windows XP, Windows Server 2003, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016 can accept this header on all commands, while only honoring this header on GET commands.

[<3> Section 2.2.3](#): This request header can be supported by the **WebDAV server** in IIS for Windows 2000, Windows XP, and Windows Server 2003.

[<4> Section 2.2.4](#): This method can be supported by the WebDAV server in IIS for Windows 2000, Windows XP, and Windows Server 2003, and is used to pass queries to Windows Index Server.

[<5> Section 2.2.5.1](#): The WebDAV server in IIS for Windows 2000, Windows XP, Windows Server 2003, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016 return the additional iscollection and ishidden **properties** as "live" properties.

<6> [Section 2.2.5.2](#): The WebDAV server in IIS for Windows 2000, Windows XP, and Windows Server 2003 does not support **locking collections**.

<7> [Section 2.2.5.3](#): The WebDAV server in IIS for Windows 2000, Windows XP, Windows Server 2003, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016 allows the use of multiple <d:prop></d:prop> elements to encapsulate multiple properties.

<8> [Section 3.1.3](#): The WebDAV server in Windows 2000, Windows XP, Windows Server 2003, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016 can return the MS-Author-Via: DAV response header if **WebDAV** is enabled on the server.

<9> [Section 3.1.5.3](#): The Depth header field is used with several WebDAV commands. However, Windows implementations only support the "noroot" extension on two verbs: **DELETE** and **PROPFIND**. "1,noroot" is supported for **PROPFIND** and "infinity,noroot" is supported on **DELETE**. Attempting to specify "noroot" on other verbs that support the [Depth header](#) field will result in a "400 Bad Request" error being returned.

<10> [Section 3.2.5.1](#): The WebDAV server in IIS for Windows 2000, Windows XP, and Windows Server 2003 requires WRITE access to the file to return the source of the file.

<11> [Section 3.2.5.1](#): The WebDAV server in IIS for Windows 2000, Windows XP, Windows Server 2003, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016 assumes the header as Translate: t by default.

<12> [Section 3.2.5.2](#): The WebDAV server in IIS for Windows 2000, Windows XP, Windows Server 2003, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, and Windows Server 2016 can return the [MS-Author-Via response header](#) in response to an OPTIONS request.

<13> [Section 3.2.5.3](#): The WebDAV server in IIS for Windows 2000, Windows XP, and Windows Server 2003 supports the Depth "noroot" request header extension.

<14> [Section 3.2.5.4](#): The WebDAV server in IIS for Windows 2000, Windows XP, and Windows Server 2003 supports the SEARCH method.

<15> [Section 5.1](#): For a Translate header value of FALSE, the WebDAV server in IIS for Windows 2000, Windows XP, and Windows Server 2003 requires WRITE access to return the source of the file.

7 Change Tracking

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

8 Index

"

["Live" properties](#) 12

[

[\[RFC4918\]](#) 12

A

Abstract data model

[client](#) 14

[server](#) 15

[WebDAV client details](#) 14

[WebDAV server details](#) 15

[Additional "Live" properties](#) 12

[Applicability](#) 8

C

[Capability negotiation](#) 9

[Change tracking](#) 24

Client

[abstract data model](#) 14

[higher-layer triggered events](#) 14

[initialization](#) 14

[other local events](#) 15

[timer events](#) 15

[timers](#) 14

[Collections](#) 12

D

Data model - abstract

[client](#) 14

[server](#) 15

[WebDAV client details](#) 14

[WebDAV server details](#) 15

Depth "noroot" Request header extension

[overview](#) 12

[WebDAV client details](#) 15

[WebDAV server details](#) 16

[Depth "noroot" Request Header Extension example](#) 18

[Depth "noroot" Request Header Extension message](#) 12

[Deviations from \[RFC4918\]](#) 12

E

Examples

[Depth "noroot" Request Header Extension](#) 18

[MS-Author-Via Response Header](#) 17

[overview](#) 17

[Translate Request Header](#) 17

F

[Fields - vendor-extensible](#) 9

G

[Glossary](#) 6

H

Higher-layer triggered events

[client](#) 14

[server](#) 15

[WebDAV client details](#) 14

[WebDAV server details](#) 15

I

[Implementer - security considerations](#) 21

[Index of security parameters](#) 21

[Informative references](#) 8

Initialization

[client](#) 14

[server](#) 15

[WebDAV client details](#) 14

[WebDAV server details](#) 15

[Introduction](#) 6

L

Local events

[WebDAV client details](#) 15

[WebDAV server details](#) 16

[Locking](#) 12

M

Message processing

[WebDAV client details](#) 14

[WebDAV server details](#) 15

Messages

[Depth "noroot" Request Header Extension](#) 12

[MS-Author-Via Response Header](#) 11

[overview](#) 10

[SEARCH Method](#) 12

[syntax](#) 10

[Translate Request Header](#) 10

[transport](#) 10

MS-Author-Via Response header

[overview](#) 11

[WebDAV client details](#) 14

[WebDAV server details](#) 16

[MS-Author-Via Response Header example](#) 17

[MS-Author-Via Response Header message](#) 11

[Multiple Property elements](#) 13

N

[Normative references](#) 7

O

Other local events

[client](#) 15

[server](#) 16

[Overview](#) 8

[Overview \(synopsis\)](#) 8

P

[Parameters - security index](#) 21
[Preconditions](#) 8
[Prerequisites](#) 8
[Product behavior](#) 22
Protocol Details
 [overview](#) 14

R

[References](#) 7
 [informative](#) 8
 [normative](#) 7
[Relationship to other protocols](#) 8

S

SEARCH method
 [overview](#) 12
 [WebDAV client details](#) 15
 [WebDAV server details](#) 16
[SEARCH Method message](#) 12
Security
 [implementer considerations](#) 21
 [parameter index](#) 21
Sequencing rules
 [WebDAV client details](#) 14
 [WebDAV server details](#) 15
Server
 [abstract data model](#) 15
 [higher-layer triggered events](#) 15
 [initialization](#) 15
 [other local events](#) 16
 [timer events](#) 16
 [timers](#) 15
[Standards assignments](#) 9
[Syntax](#) 10

T

Timer events
 [client](#) 15
 [server](#) 16
 [WebDAV client details](#) 15
 [WebDAV server details](#) 16
Timers
 [client](#) 14
 [server](#) 15
 [WebDAV client details](#) 14
 [WebDAV server details](#) 15
[Tracking changes](#) 24
Translate Request header
 [example](#) 17
 [overview](#) 10
 [WebDAV client details](#) 14
 [WebDAV server details](#) 16
[Translate Request Header message](#) 10
[Transport](#) 10
Triggered events - higher-layer
 [client](#) 14
 [server](#) 15
 [WebDAV client details](#) 14
 [WebDAV server details](#) 15

V

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

W

WebDAV client details
 [abstract data model](#) 14
 [Depth "noroot" Request header extension](#) 15
 [higher-layer triggered events](#) 14
 [initialization](#) 14
 [local events](#) 15
 [message processing](#) 14
 [MS-Author-Via Response header](#) 14
 [SEARCH method](#) 15
 [sequencing rules](#) 14
 [timer events](#) 15
 [timers](#) 14
 [Translate Request header](#) 14
WebDAV server details
 [abstract data model](#) 15
 [Depth "noroot" Request header extension](#) 16
 [higher-layer triggered events](#) 15
 [initialization](#) 15
 [local events](#) 16
 [message processing](#) 15
 [MS-Author-Via Response header](#) 16
 [SEARCH method](#) 16
 [sequencing rules](#) 15
 [timer events](#) 16
 [timers](#) 15
 [Translate Request header](#) 16