

[MS-XWDSEARCH]: Web Distributed Authoring and Versioning (WebDAV) Extensions for Search

Intellectual Property Rights Notice for Open Specifications Documentation

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

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

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

Revision Summary

Date	Revision History	Revision Class	Comments
12/03/2008	1.0		Initial Release.
03/04/2009	1.01		Revised and edited technical content.
04/10/2009	2.0		Deprecated for Exchange 2010.
07/15/2009	3.0	Major	Changes made for template compliance.
11/04/2009	3.1.0	Minor	Updated the technical content.
02/10/2010	3.2.0	Minor	Updated the technical content.
05/05/2010	3.3.0	Minor	Updated the technical content.
08/04/2010	3.3.0	No change	No changes to the meaning, language, or formatting of the technical content.
11/03/2010	3.3.0	No change	No changes to the meaning, language, or formatting of the technical content.
03/18/2011	3.4	Minor	Clarified the meaning of the technical content.

Table of Contents

1	Introduction	6
1.1	Glossary	6
1.2	References.....	6
1.2.1	Normative References.....	6
1.2.2	Informative References	7
1.3	Overview	7
1.4	Relationship to Other Protocols.....	8
1.5	Prerequisites/Preconditions	8
1.6	Applicability Statement.....	8
1.7	Versioning and Capability Negotiation.....	8
1.8	Vendor-Extensible Fields.....	8
1.9	Standards Assignments	8
2	Messages.....	9
2.1	Transport.....	9
2.2	Message Syntax	9
2.2.1	Headers	9
2.2.1.1	Accept-Ranges	9
2.2.1.2	Range	9
2.2.1.3	Content-Range	10
2.2.2	XML Element	10
2.2.2.1	DAV:contentrange	10
2.2.3	Properties.....	11
2.2.3.1	PidNameDavSearchRequest	11
2.2.3.2	PidNameDavSearchType	11
2.2.3.2.1	Dynamic	11
2.2.3.2.2	Static.....	11
2.2.3.3	PidNameDavResourceType.....	11
2.2.4	Query Grammar	11
2.2.4.1	select-clause.....	13
2.2.4.1.1	Syntax.....	13
2.2.4.1.1.1	list.....	13
2.2.4.2	from-predicate	14
2.2.4.2.1	Syntax.....	14
2.2.4.2.1.1	scope.....	14
2.2.4.2.1.2	url	14
2.2.4.2.2	scope-specification	14
2.2.4.2.2.1	Syntax	14
2.2.4.2.2.1.1	depth	14
2.2.4.2.2.1.2	url.....	15
2.2.4.2.2.1.3	view	15
2.2.4.3	where-clause	15
2.2.4.3.1	Syntax.....	15
2.2.4.3.1.1	expression.....	15
2.2.4.3.2	comparison-predicate	15
2.2.4.3.2.1	Syntax	15
2.2.4.3.2.1.1	property-name-expression	15
2.2.4.3.2.1.2	operator	16
2.2.4.3.2.1.3	value-expression.....	16
2.2.4.3.2.1.4	property-cast.....	16

2.2.4.3.2.1.4.1	Syntax	16
2.2.4.3.2.1.4.1.1	property	16
2.2.4.3.2.1.4.1.2	value	17
2.2.4.3.2.1.4.1.3	datatype	17
2.2.4.3.3	null-predicate	17
2.2.4.3.3.1	Syntax	17
2.2.4.3.3.1.1	propertyname	17
2.2.4.3.4	like-predicate	17
2.2.4.3.4.1	Syntax	17
2.2.4.3.4.1.1	propertyname	17
2.2.4.3.4.1.2	wildcard-value	17
2.2.4.4	order-by-clause	17
2.2.4.4.1	Syntax	17
2.2.4.4.1.1	propertyname	18
2.2.4.4.1.2	Sortorder	18
2.2.4.5	group-by-predicate	18
2.2.4.5.1	Syntax	18
2.2.4.5.1.1	propertyname	18
3	Protocol Details	19
3.1	Client Details	19
3.1.1	Abstract Data Model	19
3.1.2	Timers	19
3.1.3	Initialization	19
3.1.4	Higher-Layer Triggered Events	19
3.1.5	Message Processing Events and Sequencing Rules	19
3.1.5.1	Accept-Ranges	19
3.1.5.2	Range	19
3.1.5.3	Content-Range	19
3.1.5.4	DAV:contentrange	19
3.1.5.5	PidNameDavSearchRequest	20
3.1.5.6	PidNameDavSearchType	20
3.1.5.7	PidNameDavResourceType	20
3.1.6	Timer Events	20
3.1.7	Other Local Events	20
3.2	Server Details	20
3.2.1	Abstract Data Model	20
3.2.2	Timers	20
3.2.3	Initialization	20
3.2.4	Higher-Layer Triggered Events	20
3.2.5	Message Processing Events and Sequencing Rules	21
3.2.5.1	Accept-Ranges	21
3.2.5.2	Range	21
3.2.5.3	Content-Range	21
3.2.5.4	DAV:contentrange	21
3.2.5.5	PidNameDavSearchRequest	21
3.2.5.6	PidNameDavSearchType	21
3.2.5.7	PidNameDavResourceType	22
3.2.6	Timer Events	22
3.2.7	Other Local Events	22
4	Protocol Examples	23
4.1	Requesting a Range of Rows in a SEARCH Request	23

4.2	Creating a Search Folder	23
4.3	Verifying Search Type	24
5	Security	26
5.1	Security Considerations for Implementers	26
5.2	Index of Security Parameters	26
6	Appendix A: Product Behavior	27
7	Change Tracking.....	28
8	Index	30

1 Introduction

The Web Distributed Authoring and Versioning (WebDAV) Extensions for Search extend the **Web Distributed Authoring and Versioning Protocol (WebDAV)**, which is described in the HTTP Extensions for Distributed Authoring -- WEBDAV ([\[RFC2518\]](#)), and the Web Distributed Authoring and Versioning (WebDAV) Protocol: Server Extensions, as described in [\[MS-WDVSE\]](#), to allow clients to request ranges for server-side searches of content. The WebDAV Extensions for Search also allow clients to create persistent **search folders** on the server.

1.1 Glossary

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

Augmented Backus-Naur Form (ABNF)
Hypertext Transfer Protocol (HTTP)
XML

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

Folder object
search folder
soft delete
Uniform Resource Locator (URL)
Web Distributed Authoring and Versioning Protocol (WebDAV)
WebDAV client
WebDAV server

The following terms are specific to this document:

query grammar: A set of definitions of XML elements, attributes (1), and constraints on their relations and values that defines a set of queries and the intended semantics.

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

1.2 References

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information. Please check the archive site, <http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624>, as an additional source.

[MS-OXCDATA] Microsoft Corporation, "[Data Structures](#)", April 2008.

[MS-OXPROPS] Microsoft Corporation, "[Exchange Server Protocols Master Property List](#)", April 2008.

[MS-WDVSE] Microsoft Corporation, "[Web Distributed Authoring and Versioning \(WebDAV\) Protocol: Server Extensions](#)", September 2007.

[MS-XWDFOLD] Microsoft Corporation, "[Web Distributed Authoring and Versioning \(WebDAV\) Extensions for Folders Support](#)", December 2008.

[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.ietf.org/rfc/rfc2119.txt>

[RFC2518] Goland, Y., Whitehead, E., Faizi, A., et al., "HTTP Extensions for Distributed Authoring - WebDAV", RFC 2518, February 1999, <http://www.ietf.org/rfc/rfc2518.txt>

[RFC5234] Crocker, D., Ed., and Overell, P., "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008, <http://www.ietf.org/rfc/rfc5234.txt>

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

1.2.2 Informative References

[MS-GLOS] Microsoft Corporation, "[Windows Protocols Master Glossary](#)", March 2007.

[MS-OXGLOS] Microsoft Corporation, "[Exchange Server Protocols Master Glossary](#)", April 2008.

[MS-XWDEXT] Microsoft Corporation, "[Web Distributed Authoring and Versioning \(WebDAV\) Core Extensions](#)", July 2009.

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

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

1.3 Overview

The WebDAV Extensions for Search allow clients to request that the server perform a search on the contents of the server and return a specific result set. The extensions also allow the client to create a search folder on the server.

This protocol specifies the following extensions to [\[RFC2518\]](#), [\[MS-WDVSE\]](#), and [\[RFC2068\]](#):

- An extension to the **Accept-Ranges** header, as described in [\[RFC2068\]](#), to indicate support for the extensions specified in this document.
- Extensions to the **HTTP Range** and **Content-Range** headers, as described in [\[RFC2068\]](#) to allow clients to request a range of rows.
- An XML element in the DAV: namespace that allows the server to indicate the range of rows that were returned.
- A property that can be set on a search folder to indicate the search that is to be performed.
- A new value for the **PidNameDavResourceType** property ([\[MS-OXPROPS\]](#) section 2.447) to indicate that a **Folder object** is a search folder.
- A **query grammar** that is used to express search requests.

1.4 Relationship to Other Protocols

The WebDAV Extensions for Search rely on the Web Distributed Authoring and Versioning (WebDAV) Core extensions, as described in [\[MS-XWDEXT\]](#); the HTTP Extensions for Distributed Authoring – WEBDAV, as described in [\[RFC2518\]](#); the Web Distributed Authoring and Versioning (WebDAV) Protocol: Server Extensions, as described in [\[MS-WDVSE\]](#), and the Hypertext Transfer Protocol – HTTP/1.1, as described in [\[RFC2068\]](#). It also relies on HTTP Over TLS, as described in [\[RFC2818\]](#), for data protection services.

All properties are listed in [\[MS-OXPROPS\]](#) and are formatted as described in [\[MS-OXCDATA\]](#) for use with WebDAV.

1.5 Prerequisites/Preconditions

The WebDAV Extensions for Search require a **WebDAV server** as described in [\[RFC2518\]](#). The extensions also require that **WebDAV clients** have a **Uniform Resource Locator (URL)** that points to the WebDAV server.

1.6 Applicability Statement

The WebDAV Extensions for Search can be used to control what results are returned from requests using the **SEARCH** or **PROPFIND** methods and can also be used to create and modify persistent searches in the form of search folders.

1.7 Versioning and Capability Negotiation

Clients can determine whether a server supports the extensions to the **Range** and **Content-Range** headers specified in this document by sending an **OPTIONS** method command, as described in [\[RFC2068\]](#), to the server and examining the response. If the server supports the WebDAV Extensions for Search, it returns an **Accept-Ranges** header set to "rows" in the **OPTIONS** method response.

Clients can determine whether a server supports the query grammar specified in this document by sending an **OPTIONS** command to the server and examining the response. If the server supports the WebDAV Extensions for Search, it returns a **DASL** header, as described in [\[RFC5323\]](#) section 3.2, set to "<DAV:sql>" in the **OPTIONS** method response.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

The WebDAV Extensions for Search use the same transport mechanisms as those specified in [\[RFC2518\]](#) and [\[MS-WDVSE\]](#).

2.2 Message Syntax

2.2.1 Headers

The **Augmented Backus-Naur Form (ABNF)** notation, as specified in [\[RFC5234\]](#), is used to specify the format of the following headers.

2.2.1.1 Accept-Ranges

The **Accept-Ranges** header is specified in [\[RFC2068\]](#). The WebDAV Extensions for Search extend the **Accept-Ranges** header by defining a new range unit. The following ABNF notation specifies the format of the new range unit.

```
row-range-unit = "rows"
```

2.2.1.2 Range

The **Range** header is optional in a **SEARCH** method request. The WebDAV Extensions for Search extend the ABNF specified in [\[RFC2068\]](#) for the **Range** header by defining a new row-ranges-specifier rule. The byte-ranges-specifier rule is specified in [\[RFC2068\]](#).

```
ranges-specifier = byte-ranges-specifier
                  / row-ranges-specifier

row-ranges-specifier = row-range-unit "=" [row-range-list]

row-range-list = row-range ["," row-range-list]

row-range = row-range-spec / suffix-row-range-spec

row-range-spec = first-row-num "-" [last-row-num]

first-row-num  = 1*DIGIT
last-row-num   = 1*DIGIT
suffix-row-range-spec = "-" suffix-length
suffix-length  = 1*DIGIT
row-range-unit = "rows"
```

Examples:

```
Range: rows=20-39
Range: rows=0-5, 10-15, 20-25, 30-35
Range: rows=-50 (the last 50 rows)
Range: rows=228- (all rows from the 228th to the end of the result set)
```

Range: rows=0-4,-5 (the first and last five rows)

2.2.1.3 Content-Range

The **Content-Range** header is specified in [RFC2068](#) and extended here. The byte-content-range-spec rule is specified in [RFC2068](#).

```
Content-Range = "Content-Range" ":" content-range-spec

content-range-spec = byte-content-range-spec / row-content-range-spec

row-content-range-spec = row-range-unit row-range-list [";" "total=" total-rows] / "total="
total-rows

row-range-list = row-range ["," row-range-list]

row-range = row-range-spec / suffix-row-range-spec

row-range-spec = first-row-num "-" [last-row-num]

suffix-row-range-spec = "-" suffix-length

first-row-num = 1*DIGIT
last-row-num = 1*DIGIT
suffix-length = 1*DIGIT
total-rows = 1*DIGIT / "*"
row-range-unit = "rows"
```

Examples:

```
Content-Range: rows 0-5; total=200
Content-Range: rows 0-5, 7-10; total=*
Content-Range: rows 0-50; total=51
Content-Range: total=0
```

2.2.2 XML Element

2.2.2.1 DAV:contentrange

The **DAV:contentrange** XML element is similar in format and use to the **Content-Range** header, as specified in section [2.2.1.3](#).

```
DAV:contentrange = "<DAV:contentrange>" content-range-spec "</DAV:contentrange>"

content-range-spec = first-row-num "-" last-row-num

first-row-num = 1*DIGIT
last-row-num = 1*DIGIT
```

2.2.3 Properties

2.2.3.1 PidNameDavSearchRequest

DAV Property name: DAV:searchrequest

Data type: PtypString ([\[MS-OXCDATA\]](#) section 2.11.1.5)

The **PidNameDavSearchRequest** property ([\[MS-OXPROPS\]](#) section 2.448) indicates the search that the search folder will perform. The syntax of the **PidNameDavSearchRequest** property is specified in section [2.2.4](#). This property MUST be present on search folders and MUST NOT be present on other types of resources.

2.2.3.2 PidNameDavSearchType

DAV Property name: DAV:searchtype

Data type: PtypString ([\[MS-OXCDATA\]](#) section 2.11.1.5)

The **PidNameDavSearchType** property ([\[MS-OXPROPS\]](#) section 2.449) indicates what type of search the search folder is performing. The value MUST be either "dynamic" or "static". The property is set by the server and is read-only to the client. This property MUST be present on search folders and MUST NOT be present on other types of resources.

2.2.3.2.1 Dynamic

The server MUST set the **PidNameDavSearchType** property ([\[MS-OXPROPS\]](#) section 2.449) to "dynamic" when the server updates the contents of the search folder as content changes on the server.

2.2.3.2.2 Static

The server MUST set the **PidNameDavSearchType** property ([\[MS-OXPROPS\]](#) section 2.449) to "static" when the server does not update the contents of the search folder as content changes on the server.

2.2.3.3 PidNameDavResourceType

The WebDAV Extensions for Search specify a new value for the **PidNameDavResourceType** property ([\[MS-OXPROPS\]](#) section 2.447). This value is "<DAV:searchresults/>". If this value and "<DAV:collection/>" are present in the **PidNameDavResourceType** property on a Folder object, that Folder object is a search folder. The **PidNameDavResourceType** property is specified in [\[RFC2518\]](#) as **DAV:resourcetype**.

2.2.4 Query Grammar

The grammar required for the **PidNameDavSearchRequest** property ([\[MS-OXPROPS\]](#) section 2.448) is defined by the following ABNF notation.

```
search-request = select-clause
                from-predicate
                [where-clause]
                [( order-by-clause [group-by-predicate] ) /
                 (group-by-predicate [order-by-clause])]
```

```

select-clause = "SELECT" select-items
select-items = ["ALL" / "DISTINCT"] ( "*" / select-specific )
select-specific = property-expression *["," property-expression]
property-expression = property-name / property-name " AS " alias
property-name = DQUOTE 1*VCHAR DQUOTE
alias = DQUOTE 1*VCHAR DQUOTE

from-predicate = "FROM" from-clause
from-clause = scope-specification / target-url

scope-specification = "SCOPE" "(" ["" scope-arguments "" ] *["," "" scope-arguments "" ]
)" ["AS" view-spec]
scope-arguments = [traversal-depth] target-url
traversal-depth = (("SHALLOW" / "DEEP" / "HIERARCHICAL" / "SOFTDELETED") "TRAVERSAL OF")
target-url = (DQUOTE 1*VCHAR DQUOTE) / ("" 1*VCHAR "")
view-spec = DQUOTE 1*VCHAR DQUOTE

where-clause = "WHERE" condition-expression
condition-expression = boolean-expression / ( condition-expression "OR" boolean-expression )
boolean-expression = boolean-factor / ( boolean-expression "AND" boolean-factor )
boolean-factor = ["NOT"] boolean-item [test]
test = "IS" ["NOT"] evaluation
evaluation = "TRUE" / "FALSE"
boolean-item = predicate / ( "(" condition-expression ")" )
predicate = comparison-predicate / null-predicate / like-predicate

null-predicate = property-name "IS" ["NOT"] "NULL"

like-predicate = propertyname-expression ["NOT"] "LIKE" wildcard-search-pattern
wildcard-search-pattern = like-prefix / like-contains
like-prefix = "" 1*VCHAR "%'"
like-contains = "'%" 1*VCHAR "%'"

comparison-predicate = propertyname-expression operator value-expression
propertyname-expression = property-name / property-cast
property-cast = "CAST" "(" property-name "AS" data-type ")"
value-expression = property-value / value-cast
value-cast = "CAST" "(" property-value "AS" data-type ")"
operator = "=" / "<>" / "!=" / "<" / ">" / "<=" / ">="

order-by-clause = "ORDER BY" order-clause
order-clause = sort-specification *["," sort-specification]
sort-specification = property-name [sort-order]
sort-order = "ASC" / "DESC"

group-by-predicate = "GROUP BY" group-clause
group-clause = group-spec *["," group-spec]
group-spec = property-name

property-value = evaluation / 1*DIGIT / ("" 1*VCHAR "")

data-type = ( [WebDAV-type-modifier] WebDAV-Type ) / DBType

WebDAV-type-modifier = "limited." /
                        "mv."

WebDAV-Type = "boolean" /
              "bin.hex" /
              "bin.base64" /

```

```

        "char" /
        "dateTime" /
        "dateTime.tz" /
        "dateTime.rfc1123" /
        "fixed.l4.4" /
        "float" /
        "i1" /
        "i2" /
        "i4" /
        "i8" /
        "int" /
        "number" /
        "r4" /
        "r8" /
        "string" /
        "ui1" /
        "ui2" /
        "ui4" /
        "ui8" /
        "uri" /
        "uuid"

DBType = "DBTYPE_BOOL" /
        "DBTYPE_BSTR" /
        "DBTYPE_BYTES" /
        "DBTYPE_CY" /
        "DBTYPE_DATE" /
        "DBTYPE_FILETIME" /
        "DBTYPE_GUID" /
        "DBTYPE_I1" /
        "DBTYPE_I2" /
        "DBTYPE_I4" /
        "DBTYPE_I8" /
        "DBTYPE_R4" /
        "DBTYPE_R8" /
        "DBTYPE_STR" /
        "DBTYPE_UI1" /
        "DBTYPE_UI2" /
        "DBTYPE_UI4" /
        "DBTYPE_UI8" /
        "DBTYPE_WSTR"

```

2.2.4.1 select-clause

The select-clause clause specifies what properties are to be returned in a search. It **MUST** be present in a search request.

2.2.4.1.1 Syntax

```
SELECT [ALL | DISTINCT] list
```

The server **MUST** ignore the ALL and DISTINCT prefixes.

2.2.4.1.1.1 list

The *list* parameter can have the following values.

Value name	Description
*	The server MUST return the properties defined in the PidNameExchDataExpectedContentClass property ([MS-OXPROPS] section 2.465) on the Folder object being searched.
A comma-delimited list of WebDAV properties.	The server MUST return the requested properties in the result set.

2.2.4.2 from-predicate

The from-predicate predicate specifies the Folder objects on which the search will be performed. It MUST be present in a search request.

2.2.4.2.1 Syntax

```
FROM scope | url
```

2.2.4.2.1.1 scope

The *scope* parameter can be used to control the location and depth of a search. If the *scope* parameter is used, the *url* parameter, as specified in section [2.2.4.2.1.2](#), MUST NOT be used. The syntax for the *scope* parameter is specified in section [2.2.4.2.2](#).

2.2.4.2.1.2 url

The *url* parameter can be used to control the location of a search. The search that will be performed will be a SHALLOW search, as specified in section [2.2.4.2.2.1.1](#). The *url* parameter is a URL of the Folder object to be searched. If this parameter is omitted, then the Request-URI, as specified in [\[RFC2068\]](#), is used in its place.

2.2.4.2.2 scope-specification

The scope-specification element is an optional part of the from-predicate predicate and is used to specify the location and depth of a search.

2.2.4.2.2.1 Syntax

```
SCOPE('[depth TRAVERSAL OF] "url"') [AS view]
```

2.2.4.2.2.1.1 depth

The *depth* parameter specifies the depth of the search. It can be one of the values in the following table.

Value name	Description
SHALLOW	The requested search will be performed on items contained in the Folder object being searched, but not on items contained in subfolders within that Folder object.
DEEP	The requested search will be performed on items contained in the Folder object being searched and on items contained in subfolders, all the way to the bottom of the Folder

Value name	Description
	object hierarchy. <1>
HIERARCHICAL	The requested search will be performed on folder items in the Folder object being searched.
SOFTDELETED	The requested search will be performed on the soft deleted contents of the Folder object.

If no depth is specified, the server MUST treat the request as though DEEP TRAVERSAL OF was specified.

2.2.4.2.2.1.2 url

The *url* parameter can be used to control the location of a search. The *url* parameter is a URL of the Folder object to be searched. If this parameter is omitted, the Request-URI, as specified in [\[RFC2068\]](#), is used in its place.

2.2.4.2.2.1.3 view

The server MUST ignore the *view* parameter.

2.2.4.3 where-clause

The where-clause element specifies search criteria for the search request. It is optional in a search request. If the where-clause clause is included in a search request, the server MUST only return results that match the search criteria.

2.2.4.3.1 Syntax

```
WHERE expression AND | OR expression
```

2.2.4.3.1.1 expression

The *expression* parameter can take the form of any of the following predicates: comparison-predicate, null-predicate, or like-predicate.

2.2.4.3.2 comparison-predicate

The comparison-predicate predicate is used to do a simple comparison of values.

2.2.4.3.2.1 Syntax

```
property-name-expression operator value-expression
```

2.2.4.3.2.1.1 property-name-expression

The *property-name-expression* parameter can take two forms. It can be either a property name or a **property-cast** function, as specified in section [2.2.4.3.2.1.4](#), using the *property* parameter specified in section [2.2.4.3.2.1.4.1.1](#).

2.2.4.3.2.1.2 operator

The *operator* parameter can be one of the values from the following table.

Operator value	Description
=	The item on the left of the operator is tested for equality with the item on the right of the operator. If the items are equal, the criteria is satisfied.
<>	The item on the left of the operator is tested for equality with the item on the right of the operator. If the items are not equal, the criteria is satisfied. Note When this operator is used in XML , the '<' is encoded as '<' as specified in [XML] .
!=	The item on the left of the operator is tested for equality with the item on the right of the operator. If the items are not equal, the criteria is satisfied.
<	The relative value of the item on the left of the operator is compared to the relative value of the item on the right of the operator. If the value of the item on the left is less than the value of the item on the right, the criteria is satisfied. Note When this operator is used in XML, the '<' is encoded as '<' as specified in [XML] .
>	The relative value of the item on the left of the operator is compared to the relative value of the item on the right of the operator. If the value of the item on the left is greater than the value of the item on the right, the criteria is satisfied.
<=	The relative value of the item on the left of the operator is compared to the relative value of the item on the right of the operator. If the value of the item on the left is less than or equal to the value of the item on the right, the criteria is satisfied. Note When this operator is used in XML, the '<' is encoded as '<' as specified in [XML] .
>=	The relative value of the item on the left of the operator is compared to the relative value of the item on the right of the operator. If the value of the item on the left is greater than or equal to the value of the item on the right, the criteria is satisfied.

2.2.4.3.2.1.3 value-expression

The *value-expression* parameter can take two forms. It can be either a literal value or a **property-cast** function, as specified in section [2.2.4.3.2.1.4](#), using the *value* parameter specified in section [2.2.4.3.2.1.4.1.2](#).

2.2.4.3.2.1.4 property-cast

A **property-cast** function is used to specify a specific data type to be used to interpret a value.

2.2.4.3.2.1.4.1 Syntax

```
CAST("property | value" AS "datatype")
```

2.2.4.3.2.1.4.1.1 property

The *property* parameter is used to specify a property name. When this parameter is used, the value of that property will be cast to the data type specified in the *datatype* parameter, as specified in section [2.2.4.3.2.1.4.1.3](#).

2.2.4.3.2.1.4.1.2 value

The *value* parameter is used to specify a literal value. When this parameter is used, the literal value will be cast to the data type specified in the *datatype* parameter, as specified in section [2.2.4.3.2.1.4.1.3](#).

2.2.4.3.2.1.4.1.3 datatype

The *datatype* parameter can be any of the alternate names for the property value types specified in [\[MS-OXCDATA\]](#) section 2.11.1, or any of the WebDAV property value types specified in [\[MS-OXCDATA\]](#) section 2.11.1.5. See the "data-type" ABNF rule in section [2.2.4](#).

2.2.4.3.3 null-predicate

The null-predicate predicate specifies that a property's value is to be tested to determine whether it is null or, alternatively, not null.

2.2.4.3.3.1 Syntax

propertyname IS [NOT] NULL

2.2.4.3.3.1.1 propertyname

The *propertyname* parameter specifies which property's value is to be tested. This parameter is a WebDAV property name.

2.2.4.3.4 like-predicate

The like-predicate predicate specifies a string to check for within a specific property. The search does a character-by-character comparison.

2.2.4.3.4.1 Syntax

propertyname LIKE wildcard-value

2.2.4.3.4.1.1 propertyname

The *propertyname* parameter specifies which property value is to be included in the search. It is a WebDAV property name surrounded by double-quotes.

2.2.4.3.4.1.2 wildcard-value

The *wildcard-value* parameter specifies the character pattern to be searched for. It MUST be surrounded by single quotes, and the last character in the pattern MUST be a percent-sign (%).

2.2.4.4 order-by-clause

The order-by-clause clause specifies what properties are to be used to sort the search results. It is optional in search requests.

2.2.4.4.1 Syntax

ORDER BY propertyname sortorder [, propertyname sortorder]

2.2.4.4.1.1 **propertyname**

The *propertyname* parameter specifies which property value is to be used to sort the results. It is a WebDAV property name surrounded by double-quotes.

2.2.4.4.1.2 **Sortorder**

The *sortorder* parameter specifies the order the results are sorted in. It MUST be one of the following values:

Value name	Description
ASC	The results are sorted in ascending order. For string properties, this is A to Z; for numerical properties, this is lesser values to greater values.
DESC	The results are sorted in descending order. For string properties, this is Z to A; and for numerical properties, this is greater values to lesser values.

2.2.4.5 **group-by-predicate**

The group-by-predicate predicate specifies what properties are to be used to group the search results. It is optional in search requests.

2.2.4.5.1 **Syntax**

```
GROUP BY propertyname [, propertyname]
```

2.2.4.5.1.1 **propertyname**

The *propertyname* parameter specifies which property value is to be used to group the results. It is a WebDAV property name surrounded by double-quotes.

3 Protocol Details

3.1 Client Details

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

3.1.5.1 Accept-Ranges

The client can send an **OPTIONS** method request, as specified in [\[RFC2068\]](#) section 9.2, to the server and examine the response for the **Accept-Ranges** header, as specified in section [2.2.1.1](#). The client MUST NOT use the extensions to the **Range** header specified in section [2.2.1.2](#) unless the **Accept-Ranges** header contains the value "rows".

3.1.5.2 Range

The client can send an optional **Range** header, as specified in section [2.2.1.2](#), in a **SEARCH** method request, as specified in [\[MS-WDVSE\]](#), or a **PROPFIND** method request, as specified in [\[RFC2518\]](#) section 8.1, to limit the results that are returned by the server.

3.1.5.3 Content-Range

The client can examine server responses to the **SEARCH** method request, as specified in [\[MS-WDVSE\]](#), and the **PROPFIND** method request, as specified in [\[RFC2518\]](#) section 8.1, for the **Content-Range** header, as specified in section [2.2.1.3](#), to determine what rows were returned. This information can alternatively be found in the **DAV:contentrange** XML element, as specified in section [2.2.2.1](#), in the XML body of the response.

3.1.5.4 DAV:contentrange

The client can examine the XML body returned in the server response to the **SEARCH** method request, as specified in [\[MS-WDVSE\]](#), or the **PROPFIND** method request, as specified in [\[RFC2518\]](#) section 8.1, for the **DAV:contentrange** XML element, as specified in section [2.2.2.1](#), to determine what rows were returned. This information can alternatively be found in the **Content-Range** header, as specified in section [2.2.2.1](#), of the response.

3.1.5.5 PidNameDavSearchRequest

The client can send a **MKCOL** method command, as specified in [\[RFC2518\]](#) section 8.3, to create a search folder. The request MUST include an XML body with a **propertyupdate** element, as specified in [\[RFC2518\]](#) section 12.13, for the **PROPPATCH** method, as specified in [\[RFC2518\]](#) section 8.2. The **propertyupdate** element MUST set the **PidNameDavSearchRequest** property ([\[MS-OXPROPS\]](#) section 2.448) with the desired search.

The client can update the **PidNameDavSearchRequest** property to modify the search that a search folder will perform.

The client can query the **PidNameDavSearchRequest** property to determine what search a search folder is performing.

3.1.5.6 PidNameDavSearchType

The client can query the **PidNameDavSearchType** property ([\[MS-OXPROPS\]](#) section 2.449) on a search folder to determine whether the server will update the contents of the search folder updates dynamically.

3.1.5.7 PidNameDavResourceType

The client can query the **DAV:resourcetype** property, as specified in [\[RFC2518\]](#), on a Folder object to determine whether the Folder object is a search folder. The client can check for the values "<DAV:collection/>" and "<DAV:searchresults/>", which indicate that the Folder object is a search folder.

3.1.6 Timer Events

None.

3.1.7 Other Local Events

None.

3.2 Server Details

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

3.2.5.1 Accept-Ranges

If the server supports the extensions to the **Range** and **Content-Range** headers specified in section 2.2.1.2 and section 2.2.1.3 of this document, then it MUST return an **Accept-Ranges** header, as specified in section 2.2.1.1, set to "rows" in response to an **OPTIONS** method request, as specified in [RFC2068] section 9.2.

3.2.5.2 Range

If the server receives a **SEARCH** method request, as specified in [MS-WDVSE], or a **PROPFIND** method request, as specified in [RFC2518] section 8.1, with a **Range** header that uses "rows" as the unit, as specified in section 2.2.1.2, then it MUST limit the results returned in the response accordingly. The server MUST add a **DAV:contentrange** element, as specified in section 2.2.2.1, to the XML response indicating the rows that were returned, and it MUST return a **Content-Range** header, as specified in section 3.2.5.3.

If the contents of the **Range** header, as specified in section 2.2.1.2, are incorrectly formed, then the server MUST return an HTTP status 400 Bad Request error code, as specified in [RFC2068] section 10.4.1.

3.2.5.3 Content-Range

The server MUST return a **Content-Range** header, as specified in section 2.2.1.3, in response to a **SEARCH** method request, as specified in [MS-WDVSE], or a **PROPFIND** method request, as specified in [RFC2518] section 8.1, if the request contained a **Range** header, as specified in section 2.2.1.2.

3.2.5.4 DAV:contentrange

The server MUST add a **DAV:contentrange** XML element, as specified in section 2.2.2.1, to the XML body in the response to a **SEARCH** method request, as specified in [MS-WDVSE], or a **PROPFIND** method request, as specified in [RFC2518] section 8.1, when the request contained a **Range** header, as specified in section 2.2.1.2.

3.2.5.5 PidNameDavSearchRequest

If the server receives a request to create a Folder object, as specified in [MS-XWDFOLD], that includes a request to set the **PidNameDavSearchRequest** property ([MS-OXPROPS] section 2.448), then it MUST create a search folder to execute the requested search or return an error indicating the reason it cannot fulfill the request.

The **PidNameDavSearchRequest** property is read-only. If the server receives a request to update the **PidNameDavSearchRequest** property on a search folder, then it MUST return a 403 Forbidden error code, as specified in [RFC2068] section 10.4.4.

3.2.5.6 PidNameDavSearchType

The server MUST set the **PidNameDavSearchType** property ([MS-OXPROPS] section 2.449) to indicate whether it will update the contents of the search folder dynamically as content changes on the server.

The server MUST NOT allow clients to set this property and SHOULD return an HTTP 403 Forbidden error code, as specified in [\[RFC2068\]](#) section 10.4.4, within an HTTP 207 Multi-Status response, as specified in [\[RFC2518\]](#) section 10.2.

3.2.5.7 PidNameDavResourceType

The server MUST include the values "<DAV:searchresults/>" and "<DAV:collection/>" in the **PidNameDavResourceType** property ([\[MS-OXPROPS\]](#) section 2.447) on search folders.

3.2.6 Timer Events

None.

3.2.7 Other Local Events

None.

4 Protocol Examples

4.1 Requesting a Range of Rows in a SEARCH Request

Alex J. Simmons is using a WebDAV client to search his Inbox. He wants to limit the search results to the third and fourth items only.

The client begins by creating a **SEARCH** method request, as described in [\[MS-WDVSE\]](#). Then, the client adds the **Range** header, as described in section [2.2.1.2](#), with a value of "rows=2-3" to limit the results to the desired subset.

```
SEARCH /exchange/alex/inbox HTTP/1.1
Range: rows=2-3
Content-Type: text/xml

<?xml version="1.0"?>
<g:searchrequest xmlns:g="DAV:">
  <g:sql>
    SELECT "DAV:displayname" FROM SCOPE('SHALLOW TRAVERSAL OF "/exchange/alex/inbox"')
  </g:sql>
</g:searchrequest>
```

The server processes the **SEARCH** method request and returns the results. In this case, there were nine total items in Alex's Inbox. The server adds the **Content-Range** header, as specified in section [2.2.1.3](#), with a value of "rows 2-3; total=9," indicating that row numbers 2 and 3 were returned, and that there are nine total rows that could have been returned. The server also adds the **DAV:contentrange** XML element to the XML body of the response, with a value of "2-3."

```
HTTP/1.1 207 Multi-Status
Cache-Control: no-cache
Transfer-Encoding: chunked
Content-Type: text/xml
Content-Range: rows 2-3; total=9
Accept-Ranges: rows
Date: Tue, 07 Oct 2008 04:22:48 GMT

<?xml version="1.0"?>
<a:multistatus xmlns:a="DAV:">
  <a:contentrange>2-3</a:contentrange>
  <a:response><a:href>http://server/exchange/alex/Inbox/Test3.EML
  </a:href><a:propstat><a:status>HTTP/1.1 200 OK</a:status>
  <a:prop><a:displayname>Test3.EML</a:displayname></a:prop></a:propstat></a:response>
  <a:response>
    <a:href> http://server/exchange/alex/Inbox/Test4.EML </a:href><a:propstat><a:status>HTTP/1.1
    200 OK</a:status>
    <a:prop><a:displayname> Test4.EML</a:displayname></a:prop></a:propstat>
  </a:response>
</a:multistatus>
```

4.2 Creating a Search Folder

Alex J. Simmons wants to create a persistent search folder inside his Inbox that will find all messages with "ACTION REQUIRED" in the subject. The client forms a **MKCOL** method request, as

described in [RFC2518](#), and sets the **PidNameDavSearchRequest** property ([\[MS-OXPROPS\]](#) section 2.448) to an appropriate value.

```
MKCOL /exchange/alex/inbox/Action HTTP/1.1
Content-type: text/xml

<?xml version="1.0"?>
<a:propertyupdate xmlns:a="DAV:">
  <a:set>
    <a:prop>
      <a:searchrequest>
        <a:sql>
          SELECT * FROM scope('shallow traversal of "http://server/exchange/alex/inbox"')
          WHERE "urn:schemas:httpmail:subject" LIKE '%ACTION REQUIRED%'
        </a:sql>
      </a:searchrequest>
    </a:prop>
  </a:set>
</a:propertyupdate>
```

The server processes the request and successfully creates the search folder.

```
HTTP/1.1 207 Multi-Status
Date: Thu, 16 Oct 2008 16:04:10 GMT
Content-Type: text/xml
Content-Length: 293

<?xml version="1.0" ?>
<a:multistatus xmlns:a="DAV:">
  <a:response>
    <a:href>http://server/exchange/alex/Inbox/Action</a:href>
    <a:status>HTTP/1.1 201 Created</a:status>
    <a:propstat>
      <a:status>HTTP/1.1 200 OK</a:status>
      <a:prop>
        <a:searchrequest />
      </a:prop>
    </a:propstat>
  </a:response>
</a:multistatus>
```

4.3 Verifying Search Type

The client checks to see whether the contents of the newly created search folder will be updated dynamically. The client creates a **PROPFIND** method request, as described in [RFC2518](#) to check the value of the **PidNameDavSearchType** property ([\[MS-OXPROPS\]](#) section 2.449).

```
PROPFIND /exchange/alex/inbox/Action HTTP/1.1
Content-type: text/xml
Translate: f
Depth: 0

<?xml version="1.0"?>
<a:propfind xmlns:a="DAV:">
  <a:prop><a:searchtype/></a:prop>
```


</a:propfind>

The server is dynamically updating the contents of the search folder as new content arrives. It returns a response indicating that the search type is dynamic.

```
HTTP/1.1 207 Multi-Status
Date: Thu, 16 Oct 2008 16:12:45 GMT
Content-Type: text/xml

<?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://server/exchange/alex/Inbox/Action/</a:href>
    <a:propstat>
      <a:status>HTTP/1.1 200 OK</a:status>
      <a:prop>
        <a:searchtype>dynamic</a:searchtype>
      </a:prop>
    </a:propstat>
  </a:response>
</a:multistatus>
```

5 Security

5.1 Security Considerations for Implementers

There are no special security concerns specific to the WebDAV Extensions for Search. General security considerations pertaining to the underlying transport apply, as described in [\[RFC2518\]](#) and [\[MS-WDVSE\]](#).

5.2 Index of Security Parameters

None.

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:

- Microsoft® Exchange Server 2003
- Microsoft® Exchange Server 2007

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

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

[<1> Section 2.2.4.2.2.1.1:](#) Exchange 2003 and Exchange 2007 do not support DEEP traversals in the public folder store.

7 Change Tracking

This section identifies changes that were made to the [MS-XWDSEARCH] protocol document between the November 2010 and March 2011 releases. Changes are classified as New, Major, Minor, Editorial, or No change.

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

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

- A document revision that incorporates changes to interoperability requirements or functionality.
- An extensive rewrite, addition, or deletion of major portions of content.
- The removal of a document from the documentation set.
- Changes made for template compliance.

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

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

The revision class **No change** means that no new technical or language changes were introduced. The technical content of the document is identical to the last released version, but minor editorial and formatting changes, as well as updates to the header and footer information, and to the revision summary, may have been made.

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

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

- Protocol syntax updated due to protocol revision.
- Protocol syntax removed due to protocol revision.
- New content added for template compliance.
- Content updated for template compliance.
- Content removed for template compliance.
- Obsolete document removed.

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

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

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

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

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
1 Introduction	Removed normative citations.	N	Content updated for template compliance.
1.3 Overview	Updated reference to [RFC4918] to [RFC2518].	N	Content updated.
1.4 Relationship to Other Protocols	Updated reference to [RFC4918] to [RFC2518].	N	Content updated.
1.5 Prerequisites/Preconditions	Updated reference to [RFC4918] to [RFC2518].	N	Content updated.
2.1 Transport	Updated reference to [RFC4918] to [RFC2518].	N	Content updated.
2.2.3.3 PidNameDavResourceType	Updated reference to [RFC4918] to [RFC2518].	N	Content updated.
3.1.5.5 PidNameDavSearchRequest	Updated reference to [RFC4918] to [RFC2518].	N	Content updated.
5.1 Security Considerations for Implementers	Updated reference to [RFC4918] to [RFC2518].	N	Content updated.

8 Index

A

Abstract data model
 [client](#) 19
 [server](#) 20
[Accept-Ranges header](#) 9
[Applicability](#) 8

C

[Capability negotiation](#) 8
[Change tracking](#) 28
Client
 [abstract data model](#) 19
 [higher-layer triggered events](#) 19
 [initialization](#) 19
 [other local events](#) 20
 [timer events](#) 20
 [timers](#) 19
Client - message processing
 [Accept-Ranges](#) 19
 [Content-Range](#) 19
 [DAV:contentrange](#) 19
 [PidNameDavResourceType](#) 20
 [PidNameDavSearchRequest](#) 20
 [PidNameDavSearchType](#) 20
 [Range](#) 19
Client - sequencing rules
 [Accept-Ranges](#) 19
 [Content-Range](#) 19
 [DAV:contentrange](#) 19
 [PidNameDavResourceType](#) 20
 [PidNameDavSearchRequest](#) 20
 [PidNameDavSearchType](#) 20
 [Range](#) 19
[Content-Range header](#) 10
[Creating a search folder example](#) 23

D

Data model - abstract
 [client](#) 19
 [server](#) 20
[DAV:contentrange XML element](#) 10

E

Examples
 [creating a search folder](#) 23
 [requesting a range of rows in a SEARCH request](#) 23
 [verifying search type](#) 24

F

[Fields - vendor-extensible](#) 8
[from-predicate query grammar element](#) 14

G

[Glossary](#) 6
[group-by-predicate query grammar](#) 18

H

Headers
 [Accept-Ranges](#) 9
 [Content-Range](#) 10
 [Range](#) 9
[Headers message](#) 9
Higher-layer triggered events
 [client](#) 19
 [server](#) 20

I

[Implementer - security considerations](#) 26
[Index of security parameters](#) 26
[Informative references](#) 7
Initialization
 [client](#) 19
 [server](#) 20
[Introduction](#) 6

M

Message processing - client
 [Accept-Ranges](#) 19
 [Content-Range](#) 19
 [DAV:contentrange](#) 19
 [PidNameDavResourceType](#) 20
 [PidNameDavSearchRequest](#) 20
 [PidNameDavSearchType](#) 20
 [Range](#) 19
Message processing - server
 [Accept-Ranges](#) 21
 [Content-Range](#) 21
 [DAV:contentrange](#) 21
 [PidNameDavResourceType](#) 22
 [PidNameDavSearchRequest](#) 21
 [PidNameDavSearchType](#) 21
 [Range](#) 21
Messages
 [Headers](#) 9
 [Query Grammar](#) 11
 [transport](#) 9

N

[Normative references](#) 6

O

[order-by-clause query grammar](#) 17
Other local events

[client](#) 20
[server](#) 22
[Overview](#) 7

P

[Parameters - security index](#) 26
[PidNameDavResourceType property](#) 11
[PidNameDavSearchRequest property](#) 11
[PidNameDavSearchType property](#) 11
[Preconditions](#) 8
[Prerequisites](#) 8
[Product behavior](#) 27
Properties
 [PidNameDavResourceType](#) 11
 [PidNameDavSearchRequest](#) 11
 [PidNameDavSearchType](#) 11

Q

Query grammar
 [from-predicate element](#) 14
 [group-by-predicate](#) 18
 [order-by-clause](#) 17
 [select-clause element](#) 13
 [where-clause element](#) 15
[Query Grammar message](#) 11

R

[Range header](#) 9
References
 [informative](#) 7
 [normative](#) 6
[Relationship to other protocols](#) 8
[Requesting a range of rows in a SEARCH request example](#) 23

S

Security
 [implementer considerations](#) 26
 [parameter index](#) 26
 [select-clause query grammar element](#) 13
Sequencing rules - client
 [Accept-Ranges](#) 19
 [Content-Range](#) 19
 [DAV:contentrange](#) 19
 [PidNameDavResourceType](#) 20
 [PidNameDavSearchRequest](#) 20
 [PidNameDavSearchType](#) 20
 [Range](#) 19
Sequencing rules - server
 [Accept-Ranges](#) 21
 [Content-Range](#) 21
 [DAV:contentrange](#) 21
 [PidNameDavResourceType](#) 22
 [PidNameDavSearchRequest](#) 21
 [PidNameDavSearchType](#) 21
 [Range](#) 21
Server
 [abstract data model](#) 20

[higher-layer triggered events](#) 20
[initialization](#) 20
[other local events](#) 22
[timer events](#) 22
[timers](#) 20

Server - message processing
 [Accept-Ranges](#) 21
 [Content-Range](#) 21
 [DAV:contentrange](#) 21
 [PidNameDavResourceType](#) 22
 [PidNameDavSearchRequest](#) 21
 [PidNameDavSearchType](#) 21
 [Range](#) 21
Server - sequencing rules
 [Accept-Ranges](#) 21
 [Content-Range](#) 21
 [DAV:contentrange](#) 21
 [PidNameDavResourceType](#) 22
 [PidNameDavSearchRequest](#) 21
 [PidNameDavSearchType](#) 21
 [Range](#) 21
[Standards assignments](#) 8

T

Timer events
 [client](#) 20
 [server](#) 22
Timers
 [client](#) 19
 [server](#) 20
[Tracking changes](#) 28
[Transport](#) 9
Triggered events - higher-layer
 [client](#) 19
 [server](#) 20

V

[Vendor-extensible fields](#) 8
[Verifying search type example](#) 24
[Versioning](#) 8

W

[where-clause query grammar element](#) 15

X

[XML element - DAV:contentrange](#) 10