

Extended MKCOL for WebDAV
draft-daboo-webdav-mkcol-00

Status of This Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with [Section 6 of BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on May 14, 2008.

Copyright Notice

Copyright (C) The IETF Trust (2007).

Abstract

This specification extends the WebDAV MKCOL method to allow collections and resources of arbitrary resourcetype to be created and to allow properties to be set at the same time.

Table of Contents

1.	Introduction	3
2.	Conventions Used in This Document	3
3.	WebDAV extended MKCOL	3
3.1.	Extended MKCOL Support	4
3.1.1.	Example: Using OPTIONS for the Discovery of Support for extended MKCOL	4
3.2.	Status Codes	5
3.3.	Example: Successful extended MKCOL Request	5
4.	Replacing existing MKxxx methods	6
4.1.	MKCALENDAR replacement	6
4.1.1.	Example: Replacing MKCALENDAR with MKCOL	6
4.2.	MKWORKSPACE replacement	8
4.2.1.	Example: Replacing MKWORKSPACE with MKCOL	9
4.3.	MKACTIVITY replacement	11
4.3.1.	Example: Replacing MKACTIVITY with MKCOL	11
5.	XML Element Definitions	14
5.1.	mkcol XML Element	14
5.2.	mkcol-response XML Element	14
6.	Security Considerations	14
7.	IANA Considerations	14
8.	Acknowledgments	15
9.	Normative References	15

1. Introduction

WebDAV [[RFC4918](#)] defines an HTTP [[RFC2616](#)] method MKCOL. This method is used to create WebDAV collections on the server. However, several WebDAV-based specifications (e.g., DeltaV [[RFC3253](#)], CalDAV [[RFC4791](#)]) define "special" collections or resources - ones which are identified by additional values in the DAV:resourcetype property assigned to the collection resource, or through other means. These "special" collections are created by new methods (e.g., MKACTIVITY, MKWORKSPACE, MKCALENDAR). The addition of a new MKxxx method for each new "special" collection or resource adds to server complexity and is detrimental to overall reliability due to the need to make sure intermediaries are aware of these methods.

This specification proposes an extension to the WebDAV MKCOL method that adds a request body allowing a client to specify WebDAV properties to be set on the newly created collection or resource. In particular, the DAV:resourcetype property can be used to create a "special" collection, or other properties used to create a "special" resource. This avoids the need to invent new MKxxx methods.

2. Conventions Used in This Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [[RFC2119](#)].

When XML element types in the namespace "DAV:" are referenced in this document outside of the context of an XML fragment, the string "DAV:" will be prefixed to the element type names.

3. WebDAV extended MKCOL

The WebDAV MKCOL request is extended to allow the inclusion of a request body. The request body is an XML document containing a single DAV:mkcol XML element at the top-level. One or more DAV:set XML elements MAY be included in the DAV:mkcol XML element to allow setting properties on the collection as it is created. In particular, to create a collection of a particular type, the DAV:resourcetype XML element MUST be included in a DAV:set XML element and MUST specify the correct resource type elements for the new resource.

As per the PROPPATCH method, servers MUST process any DAV:set instructions in document order (an exception to the normal rule that ordering is irrelevant). Instructions MUST either all be executed or none executed. Thus, if any error occurs during processing, all executed instructions MUST be undone and a proper error result

returned. Failure to set a property value on the collection MUST result in a failure of the overall MKCOL request.

Additional XML elements, specific to the type of resource being created, MAY be included in the DAV:mkcol XML element. For example, the DAV:mkactivity XML element can be included to provide the same behavior that it would have done if used in an MKACTIVITY request.

If a server attempts to make any of the property changes in an extended MKCOL request (i.e., the request is not rejected for high-level errors before processing the body), the response MUST be an XML document containing a single DAV:mkcol-response XML element, which MUST contain DAV:propstat XML elements with the status of each property.

Additional XML elements, specific to the type of resource being created, MAY be included in the DAV:mkcol-response XML element. For example, the DAV:mkactivity-response XML element can be included to provide the same behavior that it would have done if used in an MKACTIVITY response.

In all other respects the behavior of the extended MKCOL request follows that of the standard MKCOL request.

3.1. Extended MKCOL Support

A server supporting the features described in this document, MUST include "extended-mkcol" as a field in the DAV response header from an OPTIONS request on any resource that supports use of the extended MKCOL method.

3.1.1. Example: Using OPTIONS for the Discovery of Support for extended MKCOL

>> Request <<

```
OPTIONS /addressbooks/users/ HTTP/1.1
Host: addressbook.example.com
```

>> Response <<

```
HTTP/1.1 200 OK
Allow: OPTIONS, GET, HEAD, POST, PUT, DELETE, TRACE, COPY, MOVE
Allow: MKCOL, PROPFIND, PROPPATCH, LOCK, UNLOCK, REPORT, ACL
DAV: 1, 2, 3, access-control, extended-mkcol
Date: Sat, 11 Nov 2006 09:32:12 GMT
Content-Length: 0
```

Daboo

Expires May 14, 2008

[Page 4]

3.2. Status Codes

As per [Section 9.3.1 of \[RFC4918\]](#).

3.3. Example: Successful extended MKCOL Request

This example shows how the extended MKCOL request is used to create a collection of a fictitious type "special-resource".

>> Request <<

```
MKCOL /home/special/ HTTP/1.1
Host: special.example.com
Content-Type: application/xml; charset="utf-8"
Content-Length: xxxx

<?xml version="1.0" encoding="utf-8" ?>
<D:mkcol xmlns:D="DAV:"
          xmlns:E="http://example.com/ns/">
  <D:set>
    <D:prop>
      <D:resourcetype>
        <D:collection/>
        <E:special-resource/>
      </D:resourcetype>
      <D:displayname>Special Resource</D:displayname>
    </D:prop>
  </D:set>
</D:mkcol>
```


>> Response <<

```
HTTP/1.1 201 Created
Cache-Control: no-cache
Date: Sat, 11 Nov 2006 09:32:12 GMT
Content-Type: application/xml; charset="utf-8"
Content-Length: xxxx
```

```
<?xml version="1.0" encoding="utf-8" ?>
<D:mkcol-response xmlns:D="DAV:">
  <D:propstat>
    <D:prop>
      <D:resourcetype/>
      <D:displayname/>
    </D:prop>
    <D:status>HTTP/1.1 200 OK</D:status>
  </D:propstat>
</D:mkcol-response>
```

4. Replacing existing MKxxx methods

One of the goals of this extension is to eliminate the need for other extensions to define their own variant of MKCOL to create the special collections they need. This extension can be used to replace existing MKxxx methods in other extensions as detailed below. If a server supports this extension and the other extension listed, then the server **MUST** support use of the extended MKCOL method to achieve the same result as the MKxxx method of the other extension.

4.1. MKCALENDAR replacement

CalDAV [[RFC4791](#)] defines the MKCALENDAR method to create a calendar collection as well as set properties during creation.

The extended MKCOL method can be used instead by specifying both DAV:collection and CALDAV:calendar-collection XML elements in the DAV:resourcetype property, set during the extended MKCOL request.

4.1.1. Example: Replacing MKCALENDAR with MKCOL

The first example below shows an MKCALENDAR request containing a CALDAV:mkcalendar XML element in the request body, and returning a CALDAV:mkcalendar-response XML element in the response body. The second example shows the equivalent extended MKCOL request with the same request and response XML elements.

>> MKCALENDAR Request <<

MKCALENDAR /home/lisa/calendars/events/ HTTP/1.1
Host: calendar.example.com
Content-Type: application/xml; charset="utf-8"
Content-Length: xxxx

```
<?xml version="1.0" encoding="utf-8" ?>
<C:mkcalendar xmlns:D="DAV:"
               xmlns:C="urn:ietf:params:xml:ns:caldav">
  <D:set>
    <D:prop>
      <D:displayname>Lisa's Events</D:displayname>
    </D:prop>
  </D:set>
</C:mkcalendar>
```

>> MKCALENDAR Response <<

HTTP/1.1 201 Created
Cache-Control: no-cache
Date: Sat, 11 Nov 2006 09:32:12 GMT
Content-Type: application/xml; charset="utf-8"
Content-Length: xxxx

```
<?xml version="1.0" encoding="utf-8" ?>
<C:mkcalendar-response xmlns:D="DAV:"
                      xmlns:C="urn:ietf:params:xml:ns:caldav">
  <D:propstat>
    <D:prop>
      <D:displayname/>
    </D:prop>
    <D:status>HTTP/1.1 200 OK</D:status>
  </D:propstat>
</C:mkcalendar-response>
```


>> MKCOL Request <<

```
MKCOL /home/cyrus/calendars/events/ HTTP/1.1
Host: calendar.example.com
Content-Type: application/xml; charset="utf-8"
Content-Length: xxxx
```

```
<?xml version="1.0" encoding="utf-8" ?>
<D:mkcol xmlns:D="DAV:"
          xmlns:C="urn:ietf:params:xml:ns:caldav">
  <D:set>
    <D:prop>
      <D:resourcetype>
        <D:collection/>
        <C:calendar-collection/>
      </D:resourcetype>
      <D:displayname>Cyrus' Events</D:displayname>
    </D:prop>
  </D:set>
</D:mkcol>
```

>> MKCOL Response <<

```
HTTP/1.1 201 Created
Cache-Control: no-cache
Date: Sat, 11 Nov 2006 09:32:12 GMT
Content-Type: application/xml; charset="utf-8"
Content-Length: xxxx
```

```
<?xml version="1.0" encoding="utf-8" ?>
<D:mkcol-response xmlns:D="DAV:"
                  xmlns:C="urn:ietf:params:xml:ns:caldav">
  <D:propstat>
    <D:prop>
      <D:resourcetype/>
      <D:displayname/>
    </D:prop>
    <D:status>HTTP/1.1 200 OK</D:status>
  </D:propstat>
</D:mkcol-response>
```

4.2. MKWORKSPACE replacement

Versioning Extensions to WebDAV [[RFC3253](#)] defines the MKWORKSPACE method to create a collection that is used as a versioning workspace.

The extended MKCOL method can be used instead by specifying the DAV:collection XML element in the DAV:resourcetype property, and

specifying the DAV:workspace property (set to the URI of the workspace resource itself) during the extended MKCOL request.

In addition, any DAV:mkworkspace XML element that would have been sent in the MKWORKSPACE request can also be included in the DAV:mkcol XML element.

Also, the DAV:mkcol-response XML element in the response to the extended MKCOL request can also include the DAV:mkworkspace-response element, if that would have been sent in the response to the equivalent MKWORKSPACE request.

4.2.1. Example: Replacing MKWORKSPACE with MKCOL

The first example below shows an MKWORKSPACE request containing a DAV:mkworkspace XML element in the request body, and returning a DAV:mkworkspace-response XML element in the response body. The second example shows the equivalent extended MKCOL request with the same request and response XML elements.

>> MKWORKSPACE Request <<

```
MKWORKSPACE /repository/project/caldav/cyrus-branch HTTP/1.1
Host: source.example.com
Content-Type: application/xml; charset="utf-8"
Content-Length: xxxx
```

```
<?xml version="1.0" encoding="utf-8" ?>
<D:mkworkspace xmlns:D="DAV:"
    xmlns:E="http://source.example.com/ns/">
  <E:branch-owner>Cyrus Daboo</E:branch-owner>
  <E:message>Creating branch for server-to-server work.</E:message>
</D:mkworkspace>
```

>> MKACTION Response <<

```
HTTP/1.1 201 Created
Cache-Control: no-cache
Date: Sat, 11 Nov 2006 09:32:12 GMT
Content-Type: application/xml; charset="utf-8"
Content-Length: xxxx
```

```
<?xml version="1.0" encoding="utf-8" ?>
<D:mkworkspace-response xmlns:D="DAV:"
    xmlns:E="http://source.example.com/ns/">
  <E:revision>r12345</E:revision>
</D:mkworkspace-response>
```


>> MKCOL Request <<

MKCOL /repository/project/caldav/cyrus-branch HTTP/1.1

Host: source.example.com

Content-Type: application/xml; charset="utf-8"

Content-Length: xxxx

<?xml version="1.0" encoding="utf-8" ?>

<D:mkcol xmlns:D="DAV:"

xmlns:C="urn:ietf:params:xml:ns:caldav">

<D:set>

<D:prop>

<D:resourcetype>

<D:collection/>

</D:resourcetype>

<D:workspace>

<D:href>/repository/project/caldav/cyrus-branch</D:href>

</D:workspace>

</D:prop>

</D:set>

<D:mkworkspace xmlns:D="DAV:"

xmlns:E="http://source.example.com/ns/">

<E:branch-owner>Cyrus Daboo</E:branch-owner>

<E:message>Creating branch for server-to-server work.</E:message>

</D:mkworkspace>

</D:mkcol>

>> MKCOL Response <<

```
HTTP/1.1 201 Created
Cache-Control: no-cache
Date: Sat, 11 Nov 2006 09:32:12 GMT
Content-Type: application/xml; charset="utf-8"
Content-Length: xxxx
```

```
<?xml version="1.0" encoding="utf-8" ?>
<D:mkcol-response xmlns:D="DAV:"
                  xmlns:C="urn:ietf:params:xml:ns:caldav">
  <D:propstat>
    <D:prop>
      <D:resourcetype/>
    </D:prop>
    <D:status>HTTP/1.1 200 OK</D:status>
  </D:propstat>
  <D:mkworkspace-response xmlns:D="DAV:"
                        xmlns:E="http://source.example.com/ns/">
    <E:revision>r12345</E:revision>
  </D:mkworkspace-response>
</D:mkcol-response>
```

4.3. MKACTIVITY replacement

Versioning Extensions to WebDAV [[RFC3253](#)] defines the MKACTIVITY method to create a resource to represent a versioning activity.

The extended MKCOL method can be used instead by specifying the DAV:activity XML element in the DAV:resourcetype property set during the extended MKCOL request.

In addition, any DAV:mkactivity XML element that would have been sent in the MKACTIVITY request can also be included in the DAV:mkcol XML element.

Also, the DAV:mkcol-response XML element in the response to the extended MKCOL request can also include the DAV:mkactivity-response element, if that would have been sent in the response to the equivalent MKACTIVITY request.

4.3.1. Example: Replacing MKACTIVITY with MKCOL

The first example below shows an MKACTIVITY request containing a DAV:mkactivity XML element in the request body, and returning a DAV:mkactivity-response XML element in the response body. The second example shows the equivalent extended MKCOL request with the same request and response XML elements.

>> MKACTIVITY Request <<

MKACTIVITY /repository/project/caldav/release-1 HTTP/1.1
Host: source.example.com
Content-Type: application/xml; charset="utf-8"
Content-Length: xxxx

```
<?xml version="1.0" encoding="utf-8" ?>
<D:mkactivity xmlns:D="DAV:"
               xmlns:E="http://source.example.com/ns/">
  <E:tag>Release-1.0</E:tag>
  <E:message>Creating tag for v1.0 release.</E:message>
</D:mkactivity>
```

>> MKACTIVITY Response <<

HTTP/1.1 201 Created
Cache-Control: no-cache
Date: Sat, 11 Nov 2006 09:32:12 GMT
Content-Type: application/xml; charset="utf-8"
Content-Length: xxxx

```
<?xml version="1.0" encoding="utf-8" ?>
<D:mkactivity-response xmlns:D="DAV:"
                       xmlns:E="http://source.example.com/ns/">
  <E:tag>Release-1.0</E:tag>
</D:mkactivity-response>
```


>> MKCOL Request <<

MKCOL /repository/project/caldav/release-1 HTTP/1.1

Host: source.example.com

Content-Type: application/xml; charset="utf-8"

Content-Length: xxxx

<?xml version="1.0" encoding="utf-8" ?>

<D:mkcol xmlns:D="DAV:"

xmlns:C="urn:ietf:params:xml:ns:caldav">

<D:set>

<D:prop>

<D:resourcetype>

<D:activity/>

</D:resourcetype>

</D:prop>

</D:set>

<D:mkactivity xmlns:D="DAV:"

xmlns:E="http://source.example.com/ns/">

<E:tag>Release-1.0</E:tag>

<E:message>Creating tag for v1.0 release.</E:message>

</D:mkactivity>

</D:mkcol>

>> MKCOL Response <<

HTTP/1.1 201 Created

Cache-Control: no-cache

Date: Sat, 11 Nov 2006 09:32:12 GMT

Content-Type: application/xml; charset="utf-8"

Content-Length: xxxx

<?xml version="1.0" encoding="utf-8" ?>

<D:mkcol-response xmlns:D="DAV:"

xmlns:C="urn:ietf:params:xml:ns:caldav">

<D:propstat>

<D:prop>

<D:resourcetype/>

</D:prop>

<D:status>HTTP/1.1 200 OK</D:status>

</D:propstat>

<D:mkactivity-response xmlns:D="DAV:"

xmlns:E="http://source.example.com/ns/">

<E:tag>Release-1.0</E:tag>

</D:mkactivity-response>

</D:mkcol-response>

5. XML Element Definitions

5.1. mkcol XML Element

Name: mkcol

Namespace: DAV:

Purpose: Used in a request to specify properties to be set in an extended MKCOL request, as well as any additional information needed when creating the resource.

Description: This XML element is a container for the information required to modify the properties on a collection resource as it is created in an extended MKCOL request.

Definition:

```
<!ELEMENT mkcol (set+, ANY)>
```

5.2. mkcol-response XML Element

Name: mkcol-response

Namespace: DAV:

Purpose: Used in a response to indicate the status of properties that were set or failed to be set during an extended MKCOL request.

Description: This XML element is a container for the information returned about a resource that has been created in an extended MKCOL request.

Definition:

```
<!ELEMENT mkcol-response (propstat+, ANY)>
```

6. Security Considerations

This extension does not introduce any new security concerns beyond those already described in HTTP and WebDAV.

7. IANA Considerations

This document does not require any actions on the part of IANA.

8. Acknowledgments

Several people suggested this approach, including Julian Reschke and Bernard Desruisseaux.

9. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", [RFC 2616](#), June 1999.
- [RFC3253] Clemm, G., Amsden, J., Ellison, T., Kaler, C., and J. Whitehead, "Versioning Extensions to WebDAV (Web Distributed Authoring and Versioning)", [RFC 3253](#), March 2002.
- [RFC4791] Daboo, C., Desruisseaux, B., and L. Dusseault, "Calendaring Extensions to WebDAV (CalDAV)", [RFC 4791](#), March 2007.
- [RFC4918] Dusseault, L., "HTTP Extensions for Web Distributed Authoring and Versioning (WebDAV)", [RFC 4918](#), June 2007.

Author's Address

Cyrus Daboo
Apple Computer, Inc.
1 Infinite Loop
Cupertino, CA 95014
USA

EMail: cyrus@daboo.name
URI: <http://www.apple.com/>

Full Copyright Statement

Copyright (C) The IETF Trust (2007).

This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Acknowledgement

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).

