Hierarchy Extensions for Atom
draft-divilly-atom-hierarchy-00

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and 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 November 21, 2009.

Copyright Notice

Copyright (c) 2009 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents in effect on the date of publication of this document (http://trustee.ietf.org/license-info). Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

Abstract

This specification defines mechanisms for hierarchical navigation among Atom feeds and entries.
Editorial Note

To provide feedback on this Internet-Draft, join the atom-syntax mailing list (http://www.imc.org/atom-syntax/) [1].

Table of Contents

1. Introduction ............................................. 3  
   1.1. Namespace ........................................... 3  
   1.2. Notational Conventions ............................... 3  
   1.3. Terminology ......................................... 3  
2. Hierarchy Model .......................................... 3  
   2.1. Entry Classification ................................. 4  
   2.2. Entry parent representation .......................... 4  
3. Inline Representation of Hierarchical Resources ............ 5  
   3.1. Representation of Linked Resources ................... 5  
       3.1.1. The "ah:count" Extension Attribute .............. 5  
   3.2. Child and descendant feeds ............................ 5  
       3.2.1. The "down" Link ................................ 5  
       3.2.2. The "down-tree" Link ............................. 5  
       3.2.3. Examples ....................................... 6  
   3.3. Parent Entries and Parent and Ascendant Feeds ........ 6  
       3.3.1. The "up" Link .................................. 6  
       3.3.2. The "up-tree" Link ............................... 7  
       3.3.3. Examples ....................................... 7  
4. Security Considerations ................................... 8  
5. IANA Considerations ...................................... 8  
6. Normative References ..................................... 9  
Appendix A. Acknowledgements ................................ 9  
Appendix B. Revision History ................................ 9  
Authors’ Addresses .......................................... 9
1. Introduction

Many applications, besides blogs, provide their data in the form of syndicated Web feeds using formats such as Atom [RFC4287]. Some such applications organize Atom Entries in a hierarchical fashion similar to a file system.

This specification describes a means of communicating about Atom Entries that are hierarchically related to each other since resource identifiers are opaque to clients and cannot be directly manipulated for the purposes of representation exchange, i.e., navigation.

This specification proposes new XML markup to extend the Atom Syndication Format and new link relations to obtain representations of hierarchically related Atom resources.

1.1. Namespace

The XML Namespaces URI for the XML data format described in this specification is:

\[ http://purl.org/atom/hierarchy/ \]

This specification uses the prefix "ah:" for the namespace name. The prefix "atom:" is used for "http://www.w3.org/2005/Atom", the namespace name of the Atom Syndication Format [RFC4287]. These namespace prefixes are not semantically significant.

1.2. Notational Conventions

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].

1.3. Terminology

This specification uses Atom link relations to identify different types of links; see the Atom specification [RFC4287] for information about their syntax, and the IANA link relation registry for more information about specific values.

2. Hierarchy Model

A hierarchy exists when a resource indicates the likelihood of a parent and/or a child resource. The terms parent and child are indicative of the need for the former to exist before the latter can be created.
2.1. Entry Classification

The Atom Syndication Format [RFC4287] defines the Atom Entry construct. The extensions in this specification define two specialized kinds of Entry construct -- parent Entry and child Entry.

A parent Entry is a container for child Entries. A parent Entry could itself be a child of another parent Entry.

Every Entry construct is represented as an Atom Entry Document [RFC4287] referred to in this specification as an "entry" and its plural. A logical Feed comprising entirely of child entries of a given Entry is called its child feed and one comprising entirely of its parent entries is called its parent feed. Both parent feed and child feed are seen from the perspective of a given Entry resource. The entries in the parent feed and child feed of an Entry SHOULD be disjoint, i.e., not share any entries.

A parent entry contains a "down" atom:link for its child feed. A parent entry may also contain a "down-tree" atom:link for a child feed of a subset of the descendants of that parent Entry.

atom:entry
  |   atom:link@rel="down" (1..1)
  |   atom:link@rel="down-tree" (0..1)

A child entry contains an "up" atom:link for its parent feed or entry if the child only allows a single parent. A child entry may also contain an "up-tree" atom:link for a parent feed of a subset of the ascendants of that child Entry.

atom:feed
  |   atom:link@rel="up" (1..1)
  |   atom:link@rel="up-tree" (0..1)

2.2. Entry parent representation

Applications MAY allow more than one parent Entry to contain a given child Entry. This is similar to hard links in filesystems. On the other hand, certain applications allow only a single parent Entry.

A child Entry MUST use a logical Feed to represent multiple parent Entries. This implies use of a feed for the URI identified as the child Entry’s "up" link. A child Entry MAY use an entry for the URI identified as the child Entry’s "up" link if it does not allow multiple parents for that child Entry. Clients SHOULD be prepared to
inspect the representation received for the "up" link of a child Entry before assuming either cardinality models.

3. Inline Representation of Hierarchical Resources

A parent or child feed or a parent entry MAY be inlined in an entry. Clients SHOULD NOT assume that the inline representations are identical to the one available from the linked URI. Clients SHOULD use URI identified by the relevant link relation to obtain its complete representation.

3.1. Representation of Linked Resources

An entry references parent and child Entries via various links. The representation of a hierarchically linked resource can be provided in the following ways:

1. Out-of-line reference: The Atom Processor can retrieve the representation by following the URI specified in the link element.
2. Inline content with out-of-line reference: The Atom Processor can use the content specified inside the link element as an approximation of the server representation as detailed below. The embedded representation is only a hint and MAY differ from the representation obtained from the URI referenced in the link.

3.1.1. The "ah:count" Extension Attribute

On the atom:link element, the value of the "ah:count" attribute MAY be a non-negative integral value identifying an approximate count of the number of entries in the inlined feed document. If the inlined representation or the type identified in the atom:link is the Atom Entry content type, i.e., application/atom+xml;type=entry, then this attribute MUST NOT be used.

3.2. Child and descendant feeds

3.2.1. The "down" Link

A parent entry MUST contain an atom:link element with link relation of "down" to indicate the child Feed URI. The type attribute of this link element (if present), MUST be the Atom Feed content type, i.e., application/atom+xml;type=feed.

3.2.2. The "down-tree" Link

A parent entry MAY contain an atom:link element with link relation of "down-tree" to indicate the URI of a feed of descendant Entry
resources. The type attribute of this link element (if present),
MUST be the Atom Feed content type, i.e., application/
atom+xml;type=feed. This specification does not prescribe any means
of limiting the depth to which descendants are available.

3.2.3. Examples

Example: Entry with out-of-line reference to child feed

<atom:entry>
  <atom:title type="text">My Portfolio</atom:title>
  <atom:link rel="down" type="application/atom+xml;type=feed"
    href="/finance/feeds/default/portfolios/1/positions" ah:count="0"/>
  ...
</atom:entry>

Example: Parent entry representation with inline child feed

<atom:entry>
  <atom:link rel="down"
    href="/finance/feeds/default/portfolios/1/positions">
    <atom:feed>
      <atom:link rel="self"
        href="/finance/feeds/default/portfolios/1/positions"/>
      ...
    </atom:feed>
  </atom:link>
  ...
</atom:entry>

Example: Entry with out-of-line reference to descendant feed

<atom:entry>
  <atom:link rel="down-tree"
    href="/finance/feeds/default/portfolios/"/>
  ...
</atom:entry>

3.3. Parent Entries and Parent and Ascendant Feeds

3.3.1. The "up" Link

Child Entries identify the URIs of their parent Entry or multiple
parent Entry resources in their own metadata. A child entry MUST
contain an atom:link element with link relation of "up" to indicate
the parent Entry URI. If the type attribute of this link is present,
it MUST be an Atom content type.
A child feed MUST identify the URI of the parent Entry represented in the feed using the "up" link. This allows navigation back and forth between the parent Entry and the child feed. This is in addition to the "up" link present in individual child entries.

3.3.2. The "up-tree" Link

A child entry MAY contain an atom:link element with link relation of "up-tree" to indicate the URI of a feed of ascendant Entry resources. The type attribute of this link element (if present), MUST be the Atom Feed content type, i.e., application/atom+xml;type=feed. This specification does not prescribe any means of limiting the height to which ascendants are available.

3.3.3. Examples

Example: Child Entry with inline multiple parent Entries

<atom:entry>
  <atom:link rel="up" href="/finance/feeds/default/positions/NASDAQ:ORCL/up">
    <atom:feed>
      <atom:entry>
        <atom:link rel="self" href="/finance/feeds/default/portfolios/1"/>
        ...
      </atom:entry>
      <atom:entry>
        <atom:link rel="self" href="/finance/feeds/default/portfolios/2"/>
        ...
      </atom:entry>
    </atom:feed>
  </atom:link>
  ...
</atom:entry>

Example: Child feed with out-of-line reference to parent Entry

<atom:feed>
  <atom:title type="text">Positions</atom:title>
  <atom:link rel="up" href="/finance/feeds/default/portfolios/1" ah:count="2"/>
  ...
</atom:feed>
Example: Entry with out-of-line reference to ascendant feed

<atom:entry>
  <atom:link rel="up-tree"
    href="/finance/feeds/default/ portfolios/1/positions"/>
  ...
</atom:entry>

4. Security Considerations

Hierarchy Extensions for Atom is subject to the security considerations found in Section 8 of [RFC4287].

The down-tree relation can overwhelm a server if reasonable limits are not placed on the depth to which hierarchy can be navigated. For this reason, applications are advised to either restrict this relation’s usage to out-of-line content or apply reasonable limits on inline representation.

5. IANA Considerations

This specification defines the following new relations that have been added to the Link Relations registry:

- Attribute Value: down
  - Description: A URI that refers to a feed of child entries in a hierarchy of Atom Entry resources.
  - Expected display characteristics: none
  - Security considerations: See this draft

- Attribute Value: down-tree
  - Description: A URI that refers to the feed of descendant entries in a hierarchy of Atom Entry resources.
  - Expected display characteristics: none
  - Security considerations: See this draft

- Attribute Value: up
  - Description: A URI that refers to one or more parent entry resources in a hierarchy of Atom Entry resources.
  - Expected display characteristics: none
  - Security considerations: See this draft

- Attribute Value: up-tree
  - Description: A URI that refers to the feed of ascendant entries in a hierarchy of Atom Entry resources.
  - Expected display characteristics: none
  - Security considerations: See this draft
6. Normative References


Appendix A. Acknowledgements

Bill de hOra and Ashish Motivala reviewed early drafts of this I-D.

Appendix B. Revision History

00 - Initial Revision.

01 - Based on feedback from Peter Keane, Julian Reschke, and members of the CMIS TC made the following changes:
   Renamed the link relation "detail" to "down" and "master" to "up"
   Removed Section 3, 4, 6, and 7
   Changed namespace prefix from h to ah
   Added new link relations "up-tree" and "down-tree"

Authors’ Addresses

Colm Divilly
Oracle Corp.

Email: colm.divilly@oracle.com
URI: http://cdivilly.wordpress.com
Nikunj R. Mehta
Oracle Corp.

Email: nikunj.mehta@oracle.com
URI:   http://o-micron.blogspot.com/