Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 19, 2009

M. Nottingham
E. Hammer-Lahav

October 16, 2008

draft-nottingham-site-meta-00

Status of this Memo

Abstract

This memo describes a method for locating site-wide metadata for Web sites.

Table of Contents

1. Introduction

It is increasingly common for Web-based protocols to require the discovery of policy or metadata about a site before communicating with it. For example, the Robots Exclusion Protocol specifies a way for automated processes to obtain permission to access resources; likewise, the Platform for Privacy Preferences [W3C.REC-P3P-20020416] tells user-agents how to discover privacy policy beforehand.

While there are several ways to access per-resource metadata (e.g., HTTP headers, WebDAV's PROPFIND [RFC4918]), the overhead associated with them often precludes their use in these scenarios.

When this happens, it is common to designate a "well-known location" for site metadata, so that it can be easily located. However, this approach has the drawback of risking collisions, both with other such designated "well-known locations" and with pre-existing resources.

To address this, this memo proposes a single (and hopefully last) "well-known location", /site-meta, which acts as a directory to the interesting metadata about that site. Future mechanisms that require site-wide metadata can easily include an entry in the site-meta directory, thereby making their metadata cheaply available (indeed, because the site directory can be cached, the more mechanisms that use it, the more efficient it becomes) without impinging on sites' URI space.

The directory format allows different types of site metadata to be referenced by URI or included inline.

Please discuss this draft on the www-talk@w3.org [1] mailing list.

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 RFC 2119 [RFC2119].

3. the site-meta File Format

The site-meta file format is an extremely simple XML-based language [W3C.REC-xml] that allows an authority (in the URI sense) to indicate where metadata about its resources is located.

The root element is the "metadata" element, which may contain any number of "meta" elements.

```
  <metadata>
    <meta href="/robots.txt" rel="robots"/>
    <meta rel="privacy" type="application/p3p.xml" href="/w3c/p3p.xml"/>
    <meta type="application/example+xml" rel="http://example.com/rel"
          href="http://other.example.net/example">
      <example-root xmlns="http://www.example.com">
        <!-- some metadata here -->
      </example-root>
    </meta>
    <meta type="text/example">
  foo = bar
  baz = bat
  </meta>
  </metadata>
```

Unrecognised elements and attributes SHOULD be silently ignored when parsing the format, unless specified otherwise. Likewise, unless otherwise specified ordering of sibling elements SHOULD be ignored.

3.1. Site Metadata Entries

Each "meta" element represents a kind of site metadata that is available. It MUST have a "rel" attribute containing a link relation [ref TBD]. It SHOULD have a "type" attribute whose content MUST be an internet media type [RFC4288], hinting its format.

The actual metadata content may be inlined as the content of the "meta" element, and/or referred to using the "href" attribute. The metadata MUST be made available by at least one of these methods.

If the "href" attribute is present, its content MUST be a URI-Reference [RFC3986] that locates the metadata. Relative URIs MUST be evaluated with the site root URI as the base URI.

If the metadata content is included inline, it MUST appear as a child of the "meta" element. If the metadata format is XML-based, the root element of the metadata will thus be the first (and only) child element of the "meta" element. If the metadata format is textual, it will be the text content of the "meta" element (appropriately escaped, with CDATA section(s) and/or entities).

the "meta" element MUST NOT contain any children other than inlined metadata content.

4. Discovering site-meta Files

The site-wide metadata for a given authority can be discovered by

dereferencing the path /site-meta. For example, in HTTP the following request would obtain site metadata for the authority "www.example.com";

```
GET /site-meta HTTP/1.1
Host: www.example.com
```

If the resource is not available or existent (in HTTP, the 404 or 410 status code), the client SHOULD infer that site metadata is not available via this mechanism. If a representation is successfully obtained, but is not in the format described above, clients SHOULD infer that the site is using this URI for other purposes, and not process it as a site-meta file.

To aid in this process, sites using this mechanism SHOULD correctly label site-meta responses with the "application/site-meta+xml" internet media type.

5. Security Considerations

6. IANA Considerations

6.1. application/site-meta+xml media type registration

The site-meta format, when serialized as XML 1.0, can be identified with the following media type:

MIME media type name: application
MIME subtype name: site-meta+xml
Mandatory parameters: None.
Optional parameters:
  "charset": This parameter has identical semantics to the charset parameter of the "application/xml" media type as specified in RFC 3023 [RFC3023]. [RFC3023].
Encoding considerations: Identical to those of "application/xml" as described in RFC 3023 [RFC3023], section 3.2.
Security considerations: As defined in this specification. [[update upon publication]]
  In addition, as this media type uses the "+xml" convention, it shares the same security considerations as described in RFC 3023 [RFC3023], section 10.
Interoperability considerations: There are no known interoperability issues.

Published specification: This specification. [[update upon publication]]

Applications which use this media type: No known applications currently use this media type.

Additional information:

Magic number(s): As specified for "application/xml" in RFC 3023 [RFC3023], section 3.2.

File extension: None

Fragment identifiers: As specified for "application/xml" in RFC 3023 [RFC3023], section 5.

Base URI: As specified in RFC 3023 [RFC3023], section 6.

Macintosh File Type code: TEXT

Person and email address to contact for further information: Mark Nottingham <mnot@pobox.com>

Intended usage: COMMON

Author/Change controller: This specification's author(s). [[update upon publication]]

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3023] Murata, M., St. Laurent, S., and D. Kohn, "XML Media Types", RFC 3023, January 2001.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005.

[RFC4288] Freed, N. and J. Klensin, "Media Type Specifications and Registration Procedures", BCP 13, RFC 4288, December 2005.

[W3C.REC-xml] Bray, T., Paoli, J., Sperberg-McQueen, C., and E. Maler, "Extensible Markup Language (XML) 1.0 (2nd ed)", W3C REC-xml, October 2000, <http://www.w3.org/TR/REC-xml>.

7.2. Informative References

[RFC4918] Dusseault, L., "HTTP Extensions for Web Distributed Authoring and Versioning (WebDAV)", RFC 4918, June 2007.

[W3C.REC-P3P-20020416]
Marchiori, M., "The Platform for Privacy Preferences 1.0 (P3P1.0) Specification", W3C REC REC-P3P-20020416, April 2002.

URIs

[1] <http://lists.w3.org/Archives/Public/www-talk/>

Appendix A. Acknowledgements

The authors take all responsibility for errors and omissions.

Appendix B. Frequently Asked Questions

B.1. Is this mechanism appropriate for all kinds of metadata?

No. The primary use cases are described in the introduction; when it's necessary to discover metadata or policy before a resource is accessed, and/or it's necessary to describe metadata for a whole site (or large portions of it), site-meta is appropriate. In other cases (e.g., fine-grained metadata that doesn't need to be known ahead of time), other mechanisms are more appropriate.

B.2. Why not use OPTIONS * with content negotiation to discover different types of metadata directly?

Two reasons; a) OPTIONS is not cacheable -- a severe problem for scaling -- and b) it is not well-supported in browsers, and difficult to configure in servers.

B.3. Why not use a META tag or microformat in the root resource?

This places constraints on the format of a site's root resource to be HTML or similar. While extremely common, it isn't universal (e.g., mobile sites, machine-to-machine communication, etc.). Also, some root resources are very large, which would place additional overhead on clients and intervening networks.

B.4. Why not use response headers on the root resource, and have clients use HEAD?

This is attractive, in that you could either put metadata directly in response headers, or you could refer to a resource in a similar manner to site-meta. However, it requires an extra round-trip for metadata discovery, which is unacceptable in some scenarios.

B.5. Why scope metadata to be site-wide?

The alternative is to allow scoping to be dynamic and determined locally, but this has its own issues, which usually come down to a) an unreasonable number of requests to determine authoritative metadata, b) increased complexity, with a higher likelihood of implementation and interoperability (or even security) problems. Besides, many mechanisms on the Web already presume a site scope (e.g., robots.txt, P3P, cookies, javascript security), and the effort and cost required to mint a new URI authority is small and shrinking.

B.6. Why /site-meta?

It's short, descriptive and according to search indices, not widely used.

B.7. Aren't you concerned about pre-empting an authority's URI namespace?

Yes, but it's unfortunately a necessary (and already present) evil; this proposal tries to minimise future abuses.

B.8. Why use link relations instead of media types to identify kinds of metadata?

A link relation declares the intent and use of the link (or inline content, when present); a media type defines the format and processing model for those bits.

B.9. What impact does this have on existing mechanisms, such as P3P and robots.txt?

None, until they choose to use this mechanism.

B.10. Why not (insert existing similar mechanism here)?

We are aware that there are several existing proposals with similar functionality. In our estimation, none have gained sufficient traction. This may be because they were perceived to be too complex, or tied too closely to one use case.

Authors' Addresses

Mark Nottingham

Email: mnot@pobox.com
URI: http://www.mnot.net/

Eran Hammer-Lahav

Email: eran@hueniverse.com
URI: http://www.hueniverse.com/

Full Copyright Statement

Intellectual Property