

HTTP, AtomPub, REST & CMIS

27-jan-2009 / CMIS F2F - Redmond

David Nuescheler
with input from Julian Reschke

Agenda

- **REST & HTTP intro**
- **AtomPub vs. CMIS**
- **A good “Web”-citizen: Specific issues**
- **General Questions for the TC**

REST & HTTP Introduction related to CMIS

REST & HTTP intro

REST is an
architectural style.

*not a protocol.



REST & HTTP intro

A “good HTTP citizen”
follows REST constraints

*not necessarily all constraints.



Good HTTP citizens (incomplete)

(1) Stateless



Good HTTP citizens (incomplete)

- (2) Data vs. Control Orientation
- (3) Bookmarkability / Linkability
- (4) Proper use of HTTP Methods
- (5) Distributed Cacheability
- (6) Self Descriptive Messages
- (7) Decoupling



AtomPub vs. CMIS

AtomPub vs. CMIS

AtomPub is by design
functionally very limited.

*currently



AtomPub vs. CMIS

AtomPub does not support
Folders.

*and a lot of other things.



AtomPub vs. CMIS

Be a “good AtomPub citizen”.

*well designed extension.



A good “Web”-citizen: Specific CMIS issues.

Reference to REST

Rename REST-Bindings to "AtomPub Bindings" or "AtomPub Extensions"



CMIS- headers

Reconsider HTTP Extensions

Recommendation: Drop headers in favour of query parameters



Terminology

various terminology issues

APP -> AtomPub

Atom call -> [HTTP|AtomPub] request

HTTP Header Tag -> HTTP Header

URI arguments -> query parameters



JSON MIME type

bad MIME type for JSON specified

“The MIME media type for JSON text is application/json.”

<http://tools.ietf.org/html/rfc4627>



broken XML in examples

Fix XML

There are several examples where the XML is broken in that ampersands appear unescaped in attribute values, such as in:

query:

```
<link rel="next" href="http://example.org/cmis/main?unfiled&n=100" />
```



cacheability extended request headers

cacheability in presence of extension request headers

Should specify Vary header.



Permission Denied

mapping for
PermissionDeniedException

Should be 403 not 401



syntax for CMIS link types

'cmis-' before the link name

Short names need to be registered with IANA. The alternative is to use absolute URIs for the link names (actually IRIs)

<http://greenbytes.de/tech/webdav/rfc4287.html#rfc.section.4.2.7.2>



Stability of IDs in atom feeds

URL does not satisfy atom id requirements

<http://greenbytes.de/tech/webdav/rfc4287.html#rfc.section.4.2.6>

"The "atom:id" element conveys a permanent, universally unique identifier for an entry or feed."



use of PUT for partial updates

Partial update should not use
PUT

This really should use PATCH (<http://tools.ietf.org/html/rfc2068#section-19.6.1.1>), or alternatively POST



Location header in examples

Wrong location headers

In general, for PUT->2xx, the only legal Location header value would be one that is identical to the RequestURI.



confusion about mime types

Mime-types?

inconsistent use of mimetypes

“Application/cmismquery+xml”

“Application/cmismallowableactions+xml”

"application/cmism+xml;type=searchrequest"

mime-types need a proper definition suitable for IANA registration



creating new resources

AtomPub vs. CMIS

RFC 5023: new non-atom-entry resources are created by POSTing to a media collection

In CMIS, createDocument suggests to create the media entry first, and to either inline the content (base64!), or to set the content in a second request on the media entry.



PUT semantics conflict

"Because repositories **MAY** automatically create new Document Versions on a user's behalf, the Location returned may not match the URL of the PUT request."

Don't use PUT.



content negotiation priorities

"When the client sends an Accept header that applies to more than one format, the CMIS Atom format will be chosen."

Even if quality values give preference to a different format, and that different format is supported by the server?



content negotiation priorities

"When the client sends an Accept header that applies to more than one format, the CMIS Atom format will be chosen."

Even if quality values give preference to a different format, and that different format is supported by the server?



General Questions for the TC

Why have a “REST” binding?

What's the motivation behind the second binding?

Motivation? Guesses....

1. easy HTTP access using browsers?
2. standard HTTP interface for files and folders?
3. easy Atom access using feed tools / Atom Readers?
4. buzzword compliance?
5. ??

1) Browsers?

**Recommendation:
XHTML + microformats**

2) Files and Folders?

**Recommendation:
WebDAV**

3) Atom Feed tools?

Is this really a match for the CMIS Domain Model?
If yes, we should really make it Atom.

4) Buzzword compliance?

Recommendation:

Define your own mime-types.

Use standard link relations and hypertext-enabled data formats.

All important resources must have a URI.

Why two mandatory bindings?

**Is this really a good thing?
Are the AtomPub bindings
worth the trouble?**