WISHI

2019-02-21 call

Draft agenda:

— Exposing more structured media types in hypermedia exchanges
— Parsing and translating binary data [skipped]
— Planning for IETF 104

Exposing more structured media types in hypermedia exchanges

Slides from CoRE interim yesterday follow

More general problem: media types are rather static (often need to know before coding), but IoT devices are dynamic

Resource tree vs. object tree: Some of the structure may be visible on both levels (YANG!)

Other mechanisms based on media-types (Content-Formats) may have to adapt. Yesterday in CoRE: Content Negotiation

Content Negotiation

Basic REST concept:

— Server can provide multiple variants of a resource
— Client can ask for specific variant

HTTP:

— Accept-Language: user's language preferences
— Accept: media types (content types) supported
— Accept-Charset: (overtaken by events)
— Accept-Encoding: indicate compression schemes

Proactive vs Reactive Content Negotiation

"proactive": server selects the representation based upon the user agent's stated preferences

— really about client's (or user's) capabilities

— hard to describe (and to act on by server)

"reactive": server provides a list of representations for the user agent to choose from

— hypermedia style

HTTP

Browser Web: Proactive Content Negotiation hindered by large amount of choices

Web APIs: Accept header often used to select specific form (e.g., serialization)

CoAP

Tried to simplify:

— Only one choice can be expressed in `Accept` Option

— Selected choice is a **mandatory** request (4.06 otherwise)

— Using Content-Format number as opposed to content-type spec (media-type + parameters) + content-coding

Multipart-core

Media types can nest
(also true for senml-ct)

Now how to do content negotiation?

— outer level (governed by Accept)

— inner level (???)

Bigger picture

— Composition (embedding content-formats in other content-formats)

— Inheritance
(e.g., both `application/cbor` and `application/foo+cbor`)

— Choice

— Serialization (and other client capabilities)

End of CoRE slides

Collection of Resources ➔
Tree of Resources, each of which has Tree of Items

(Actually, a Graph, but we like dominant trees)

Content-Type describes internals of a representation

Tools completely change between structure-of-service and structure-of-representation: But there is no hard transition

What are we seeing

Thing description tries to do structure of representation using jsc-like description

OCF resources have interaction description (RAML) and structure (jsc-like)

MJK: Protocol bindings separate structure-of-service (affordances: event/action/properties) from structure-of-data (data shape)

Data/Meta-data separation

From structural to semantic interoperability

So far, most of this is structural interoperability
(We describe structure of some data types, but rarely what they mean)

Structural Interoperability is a prerequisite, but:
What would be needed for semantic interoperability?

Interaction model is concerned more with structure-of-service

Questions

Which of this is just transitional?
(We already no longer talk about number of bits etc.)
MJK: data constraints do remain, min/max/step, ranges…
Work on the equivalence models, separate actual semantic points from representation issues

Where do have to optimize; where is optimization detrimental to interoperability

Behavioral models (cf. mobile code in HTML)

Planning for IETF 104

- Friday meeting:
  - CoRAL discussion (Fri morning)
  - Requirements for Schema Languages, Data model component re-usability
  - Edge computing: finding the "next" place to talk to and to hold information; consumer vs. industrial
  - W3C Web of Things: reviews and discussion (maybe add a side meeting during IETF)
  - Planning further work on IoT security; security vs. semantics
- Hackathon Sat/Sun:
  - CoRAL implementation, RD
  - TD, turning on CBOR
- Summary meeting:
  - Disseminate