CoRE: Problem Details

May 25th (Wednesday), 14:00-15:30 UTC

draft-ietf-core-problem-details
Disposition of WGLC comments
branch wglc-processing

PR #18
~ 18 commits
directly taken from discussion in the mailing list

legend:
CPDdi = **Concise** Problem Details data item
SPDe = **Standard** Problem Details entry (+ SPDk)
CPDe = **Custom** Problem Details entry (+ CPDk)

CoRE interim 2022-05-25 • Carsten Bormann cabo@tzi.org
editorial (11/18)

— Reference tweaks, terminology 3cffe3c3a
— consistently use CPDdi 0cf8599; + misc 2d26665
— Bring abstract closer to title 939184b
— 4a2e07e fbb939f ad9f3e4 1c130e3
— 84beee4 c5aa620 (Thanks, Ari 16dbf55)
technical nits

— oltext (text / tag38) in registry 9c08121
Tag 38

— add explicit & (auto: null) 2836667
— explain that STRING-META is ongoing standardization 3b4ba14
— point (informatively) to W3C group draft note Strings on the Web: Language and Direction Metadata https://www.w3.org/TR/string-meta/
term: problem shape

- **Problem shape**: 8acbd7f
  The shape of the data in the problem details
  - replaces "problem type" from RFC 7807
  - less pointy
  - often there is a dominating CPDe
  - often the "problem type" is the cross-product of CPDes
- If needed, problem type can be imported from RFC 7807
  
  \{ 7807: ... \}
add and explain base-uri

— problem details can contain relative URI references
  — These are relative to a base URI (RFC 3986 § 5.1)
  — usually comes implicit from request
  — not so for stored/relayed CPDdi
— ➔ add base-uri SPDe (-5)
  — 24091df
explain response-code

— Why both
  — the response-code SPDc and
  — the CoAP response-code
— explain how they relate:
  — (approximately like in base-uri)
  — b50c416
explain ignore unknown

- "ignore unknown": Common JSON pattern
- extensibility allows adding members to a map
  - forward compatibility: ignore unknown
- note: if that is not the Right Choice:
  - a new CPDe can be defined instead
  - prevent false interoperability
- d8b2e6c
ignore unknown vs. domain extensions?

— what if an existing map key needs different data?
  — "domain extension"
— is this an instance of "ignore unknown"
  — often yes (e.g., new enum)
  — sometimes just ignore domain restriction (20 for 0..10)
  — much more murky
— do we need to spend text on this?
editorial/technical fix to 29112 example?

Example:

? &(invalidparms: 1) =>
[ * [param: ~uri, ? reason: text ]]

More true to TS29112 JSON form:

? &(invalidparms: 1) =>
{ * {param: ~uri, ? reason: text }}