URN Extended Features
The URI and Forking Questions

John C Klensin

IETF 90 - 20140725

Observation About this Slide Deck

- Delayed because of changes due to informal discussions this week (and last)
- Just an issues summary
- Absolutely nothing new – all of this has been on the mailing list

RFC 2141 URNs

- “urn” + NID + NSS
- Queries, fragments, and other specialized “tail” syntax reserved for future extension
- Conforming implementations should reject those syntax forms
- Written before 3986 and so not really subject to its semantic restrictions

RFC 3986 and the URN Drama

- Effectively imposed retroactive requirements on URNs
- URNBIS charter requires dealing with the 3986 issues
- 3986 Semantics
  - 3986 Could have said ‘if a ◊ (or “?” or #”) appears, it is used to indicate a “foo”, with the interpretation of “foo” being scheme-dependent’
  - Did not. Specified what “?” and “#” meant and how and where they were to be interpreted.

URNs include several types of “names”

- May have different requirements along with unifying themes
- One set of name-categories:
  - Pure indicators, e.g., XMPP: no object, never resolved
  - Embedding of identifiers standardized and established elsewhere, e.g., ISBN, ISSN, DOI
  - Special types of reference, e.g., RFC URNs
  - URNs with DDDS resolution
  - (probably several more, and other dimensions)
- Different communities, different needs.

Could make other categories based on

- Degree of permanence/persistence required & for how long
- What those terms mean
- Whether they apply to
  - The identifier itself
  - An object (where there is one)
  - Some types of metadata

… or a mixture of the above

Those name-category communities believe they have needs

- We are thoroughly unlikely to persuade them otherwise
- We have no ability to tell them
  - *Don't* use URNs
  - Don't use URNs in that particular way

    … or expect them to listen if we do.
- They can create their own ("forked") URN standard for their communities *any time they like.*
  - *If we want to retain control of the overall design/ architecture,, we need to treat each community as having importance*

Extending URNs, Retaining the 3986 Linkage -- Requires Work

- Verify that *all* requirements for queries and fragments are consistent with 3986 syntax and semantics, including processing (and by whom)
  - Or change those requirements (affects other protocolls
- Verify that 3986-conforming queries and fragments are sufficient for all needs of present and future communities.
  - Means that all of them must be processed in the same way
  - Requires predicting the future with high confidence or defining a migration plan if we are wrong.

Procrustes is not our friend

- Myth about a guy with a bed and how it fits
- If we
  - Create a set of constraints that make sense only in the context of
    - Needs of other communities
    - Our compulsion to stay with a particular generic standard
    - And the result looks silly enough
- Those other communities will just make their own URN standards

Why “URNs are not URIs”

- Nasty, ugly, procedural trick
  - Avoids dealing with generic URI semantics that many people don’t realize are there
  - If needed, allows a discussion of other syntax than that specified for generic URIs
- Does not modify 3986 itself
  - Avoids risk of unintentional changes to other protocols
  - Avoids delaying URN work while a time-consuming revision is done
  - 3986 has other problems
- Maybe should be called “Let URNs be URNs”

Separation Damage

- If “URI applications” are
  - Interpreting URIs for which they don’t understand the schemes
  - Doing careful parsing and syntax checks or analysis that use 3986 syntax in-depth
- Then some things will stop working with URNs
  - That would be bad
  - But there is little evidence that is happening in problematic ways.
  - If not, a much lighter-weight URN syntax would still conform in practice (and be a good idea)

3986 Has its own problems

- No one wants to think about/ discuss this but…
- If WHATWG / W3C succeed in killing 3986, presumably replacing it with something…
  - Web-centric
  - Assuming that persistent names without location properties are a myth (or just dumb)
- What happens to URNs and future revisions of whatever we do now?

3986-imposed query restrictions

- Case-sensitive query and fragment compares unless changed for *all* URNs.
- Query is always part of equality comparison
  - Fragments apparently never
- "?" inside query is data, not a canonically-separate query so
  - Urn:foo:bar?a=b?c=d   and
  - Urn:foo:bar?c=d?a=b
  - Never compare equal

3986-imposed fragment restrictions

- Fragment semantics depend on retrieval of primary resource & its media type
  - No object, no retrieval, or no media type = no fragments
  - Cannot be redefined by scheme
  - Processed solely by UA
  - Cannot access metadata not bound to retrieved object
  - Not part of equality check on primary reference URI

3986 Baggage
(Do not restrict URNs, but unneeded)

- Very long, complex spec – few actually read it
- Extensive sets of rules irrelevant to URNs (e.g., relative URLs, dot-removal)
- Non-canonical formats (no necessary URN equivalent except, maybe, for escapes)
- Complex equivalence algorithm w/ false negatives
  - Multiple processing levels & degrees
  - Assumptions about object comparison

Where is the resolver?

- If a URN my be resolved (not a pure indicator)
  - One has to find a resolver
- Not all URNs use the same one (very old principle)
- Resolver can be specified
  - As part of the NID definition (now)
    - Might indirect through an IANA registry or specified domain
  - As part of the NSS (probably now – a stretch)
  - As an additional parameter (if we allow it)
  - If we want the third, not with 3986.

If we are sensible separation from 3986

- Real issue different from perceptions
- Example:
  - If we need “?a=c?d=e” and “?d=e?a=c” equivalency
  - Have to do something about 3986 requirement
  - But probably nothing will notice, especially not generic, 3986-conforming, parsers.
- Same comment about, e.g., slipping in another delimiter that terminates the query or interpreting “?” differently

If we need such a small deviation Why separate from 3986

- Purely about expediency – fairly quick
- Alternatives include
  - Amending (“updating”) 3986 for the specific things needed by URNs, and for URNs only

    (probably n 2141bis)
  - Rewriting 3986 to make it less restrictive

    (probably best solution if we had unlimited time and energy)

If we can agree and move forward

- Have to decide whether we understand all future uses well enough to know “?” and “#” will be enough
  - If not, have to devise an extensibility mechanism for new cases
- If dependent on IANA Registries or a Domain, implications for
  - “persistence”
  - performance

What if we cannot agree?

- Almost certainly end up with a forked standard
  - Those communities are not going to go away
  - Some of them have been around a lot longer than the IETF and will likely outlive it.
- No reason to believe that a single NID registry will be preserved either, although some would fight for it
- The IESG could apply “not enough energy to do work” to the URN topic