

JCR for RDAP

draft-newton-rdap-jcr

Andy Newton

REGEXT WG IETF 100

RDAP Responses Are JSON

- But RDAP defines its JSON using prose
- No formalism
- This can make testing difficult

draft-newton-rdap-jcr

- Uses JSON Content Rules (JCR, draft-newton-json-content-rules) to define RDAP JSON
- JCR is a data definition language or schema language for JSON
 - Similar to XML Schema or Relax NG for XML
- More concise than prose
- Easier to read, especially for some non-English speakers

Why a JSON DDL?

- It is true... most uses of JSON in the IETF do not utilize complex structures and therefore easily and cleanly describe JSON with prose
- RDAP JSON is far from simple

“Speaking as a person who’s been skeptical of JSON schema efforts, it pains me to say this, but the information about large-scale message structure is scattered through this document in a diffuse way and it’d make me nervous as an implementer whether or not I was getting it right. I think it might be helpful to have a “large-scale message structure” section that quickly runs through the allowable top-level shapes of messages, and exactly what can be nested inside what.”

-- Tim Bray, APPS AREA reviewer of RDAP

About JCR

- A superset of JSON, therefore all JSON is JCR
 - Easy to take existing JSON and turn it into JCR
- Features to enable software testing

```
{
  "file-name" : "rfc7159.txt",
  "line-count" : 3426,
  "word-count" : 27886
}
```

```
{
  "file-name" : sting,
  "line-count" : 0..,
  "word-count" : 0..
}
```

```
{
  $fn,
  $lc,
  $wc
}

$fn = "file-name" : string
$lc = "line-count" : 0..
$wc = "word-count" : 0..
```

```
# for use case X
$fn = "file-name":"rfc7159.txt"
And
$lc = "line-count" : 3426
```

Next Steps

- Collecting feedback
 - Some very useful feedback on the mailing list so far
- Incorporate into NicInfo
 - Already does a little RDAP testing... this would be a good addition
- Next year – route object RDAP extensions in JCR
 - RIRs already using JCR to describe inter-RIR transfers