JSON Content Rules (JCR)

A Data Definition Language for JSON

draft-newton-json-content-rules-09

A. Newton

P. Cordell

The Problem

- Complex JSON data structures can be difficult to describe with prose.
- Lengthy prose are tedious to read, challenging if English is not your native language.
- Without formal definitions, protocol test suites are difficult to construct.
- Examples are necessary but not sufficient
  - RDAP and jCard are case studies on why examples only get you so far

We need something that…

- will help specification authors concisely and clearly describe complex JSON
- aid software developers to verify their implementations are specification conformant
- enable the construction of test suites
- and is yet legible to the "casual" reader

  (persons knowledgeable with JSON, experienced with DDLs and schema languages but perhaps not up-to-speed on JCR)

History of JCR

- Original syntax was "JSON-like"
  (one of the inspirations for CDDL)
- Tried a more wordy, verbose syntax. Nobody like it.
- Syntax is now a super-set of JSON
  - Makes it easy to start writing rules from examples
  - Other cues from ABNF, regex, etc…
  - Many discussions on syntax, and even some usability testing
- Authors are very keen on getting it to work in software first
  - Real-world testing – NicInfo now uses it to verify RDAP

An Example: Decomposing CIDR

```
[
  {
    "prefix" : "192.0.20.0",
    "length" : 24
  },
  ..
]
```

```
[
  {
    "prefix" : ipaddr,
    "length" : integer
  } *
]
```



```
$v4cidr = {
  "v4prefix" : ipv4,
  "length"   : 0..32
}

$v6cidr = {
  "v6prefix" : ipv6,
  "length"   : 0..128
}

$cidrs = [
  $v4cidr + | $v6cidr
+       ]
```

Moving Forward

- Lot's of work done
  - Discussion on JSON list (wg is now closed)
  - Also in the GitHub trackers
  - Private email
- Two open issues left
  1. Better way to OR members of objects and their values
  2. A more formal means of extending rules
- Seeking AD sponsorship via DISPATCH