JSONPath standardization

Almost, but not entirely unlike XPath for JSON

Discussion slot at DISPATCH WG meeting @ IETF108

Carsten Bormann, 2020-07-27

JSON is a data representation language

- JSON is the premier format for representing tree-shaped data for interchange
- Often, there is a need to identify elements or subtrees in such a tree, without transferring the whole tree
- XML has XPath, a complex, Turing-equivalent query language
- JSON has **JSONPath**, proposed in 2007, but never standardized

JSONPath examples

XPath	JSONPath	Result
/store/book/author	$.store.book[*].author	the authors of all books in the store
//author	$..author	all authors
/store/*	$.store.*	all things in store, which are some books and a red bicycle.
/store//price	$.store..price	the price of everything in the store.
//book[3]	$..book[2]	the third book
//book[last()]	$..book[(@.length-1)] $..book[-1:]	the last book in order.
//book[position()<3]	$..book[0,1] $..book[:2]	the first two books
//book[isbn]	$..book[?(@.isbn)]	filter all books with isbn number
//book[price<10]	$..book[?(@.price<10)]	filter all books cheapier than 10
//*	$..*	all Elements in XML document. All members of JSON structure.

```
{ "store": {
    "book": [
      { "category": "reference",
        "author": "Nigel Rees",
        "title": "Sayings of the
Century",
        "price": 8.95
      },
      { "category": "fiction",
        "author": "Evelyn Waugh",
        "title": "Sword of Honour",
        "price": 12.99
      },
      { "category": "fiction",
        "author": "Herman Melville",
        "title": "Moby Dick",
        "isbn": "0-553-21311-3",
        "price": 8.99
      },
      { "category": "fiction",
        "author": "J. R. R. Tolkien",
        "title": "The Lord of the Rings",
        "isbn": "0-395-19395-8",
        "price": 22.99
      }
    ],
    "bicycle": {
      "color": "red",
      "price": 19.95
    }
  }
}
```


https://goessner.net/articles/JsonPath/

Aren’t there other ways to do this?

Sure:

- RFC 6901, JSON Pointer, very similar in idea, but different in syntax, and limited towards pointing into a single place in a known structure
- XPath extensions for JSON — all the complexity for very little functionality
- (Insert your favorite query language here)
- But the question is less “is JSONPath always the best tool for the job” — it is being used a lot, and it would benefit from a common standard

Why do we want this now?

- Well, many of us have wanted this for a while…
  (Current occasion for me: JSONPath fits some IoT discovery well)

- Now the stars seem to have aligned to make this possible:
  - The original JSONPath author is interested in getting this done
  - An amazing project has started documenting implementation deviations

- So let’s do it, like we did RFC 6901 earlier.

Why this isn’t trivial

- JSONPath was defined in 2007, and implemented many times since
- JSONPath left expressions/filters to an “underlying scripting language”
  - JSONPath implementations have used their implementation language or a synthetic language patterned after common usage
  - These are mostly close, but not identical
- Lots of details differ
  - Fortunately, there is https://cburgmer.github.io/json-path-comparison/ (225 test cases against 37 implementations, “Proposal A” in the making)

https://cburgmer.github.io/json-path-comparison/

WG needs to decide on direction

- Find lowest common denominator and standardize that
  - Won’t cover that many real-world examples, no big benefit
- Find the gaps and start filling them all by dumping in more rubble
  - High complexity of the result, bugs will be plenty
- Define a middle ground, filling gaps neatly where existing usage abounds
  - The right thing, but probably needs some more detailed guidance
  - Needs input from implementers **and** users
  - Janus approach: look both back **and** forward

Where do we want to do this?

- New Working Group (?)
- Revive JSON Working Group specifically for this
- Do this in CBOR working group because that is alive, has the same generic data model, and actually needs a tree query language as well (and probably would promise to standardize JSONPath for JSON first)
- Stuff this into a new HTTPAPI working group, because JSONPath often is used over HTTP (as is everything else)

➔ For the DISPATCH WG to decide!

Please DISPATCH !