

ITS for JSON

FEISGILTT – Berlin – June 2015

Yves Savourel (ENLASO)



ITS – Internationalization Tag Set

- Set of internationalization and localization-related data categories
- Defined in an abstract way
- One representation for XML
- One representation for HTML 5
- Mappings for RDF, XLIFF 2, etc.
- Nothing prevent you to use ITS with other representations

JSON – JavaScript Object Notation

- Simple, compact, very easily mapped to JavaScript and other programming languages
- Implemented in virtually all programming languages
- More and more applications, especially Web applications, use JSON for data transport
- One of the main formats to carry data in the **IoT** world

Example

```
{  
  "data1": "Some date not to translate",  
  "msgs": [  
    {  
      "msg2": "Time zone: "  
    },  
    {  
      "msg3": "Weather condition below: "  
    }  
  ],  
  "data4": "Additional data to not translate"  
}
```

A first ITS feature useful in JSON: Translate

- Translate data category:
Indicates what is translatable and what is not
- With the ITS rules (external or embedded in the JSON string itself) one can process the full set of objects for localization without disturbing the non-localizable data.

Basic mechanism

- ITS applies to the name/value pairs for which the value is a string (the “string nodes”)
- The “itsRules” object is reserved for ITS
- The rules can be embedded in the data, or external (as a separate JSON object)
- By default:
 - Selectors are in JSONPath syntax
 - All string nodes are translatable
 - The content of the itsRules object is never translatable

JSONPath for selectors

XPath

//*

/data/item[2]

//item/msg

//item[type<1]

JSONPath

\$..*

\$.data..item[2]

\$..item[*].msg

\$..item[?(@.type<1)]

Demonstration

- Simple JSON+ITS reader/writer
- Load extracted data into the Acorn's XLIFF Object Model
- From there it can be translated, saved as XLIFF, merged back, etc.
(it's not JSON anymore, but extracted/abstracted translatable data)

Demonstration in the LocWorld session AL3 Thursday, June 4, 2015, 3:00pm – 4:30pm. JSON+ITS used for the Internet Of things.

Next steps

- Formalize the JSON+ITS library in Okapi (Acorn's code is just experimental)
- Write an Okapi filter that uses the library, add it to the other filters.
- Possibly, implement additional data categories
- Possibly, implement additional selector syntaxes

Useful links

- JSON specification:
<http://www.json.org>
- ITS 2.0 specification:
<http://www.w3.org/TR/its20>
- JSON + ITS wiki page:
<https://www.w3.org/International/its/wiki/JSON%2BITS>
- Okapi Acorn project (includes JSON+ITS experimental library):
<https://bitbucket.org/okapiframework/acorn>