



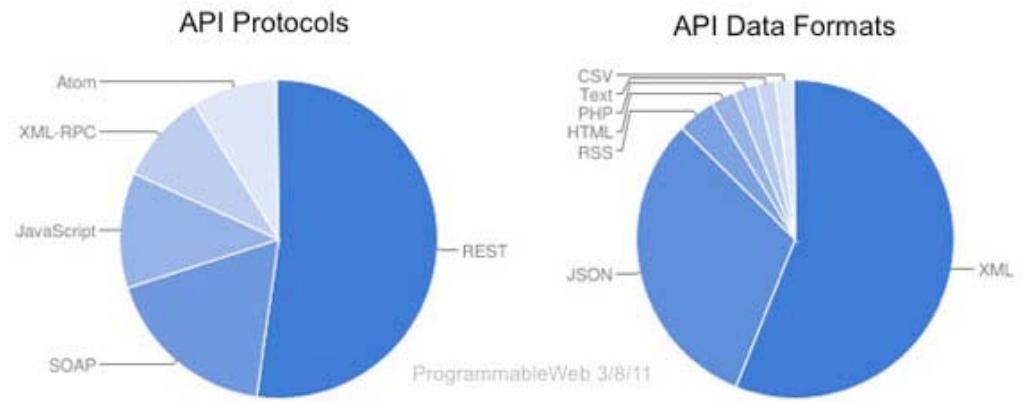
JSON in OGC standards - opportunities and challenges



Clemens Portele
OGC Interoperability Day, March 2015, Barcelona

Why is JSON relevant?

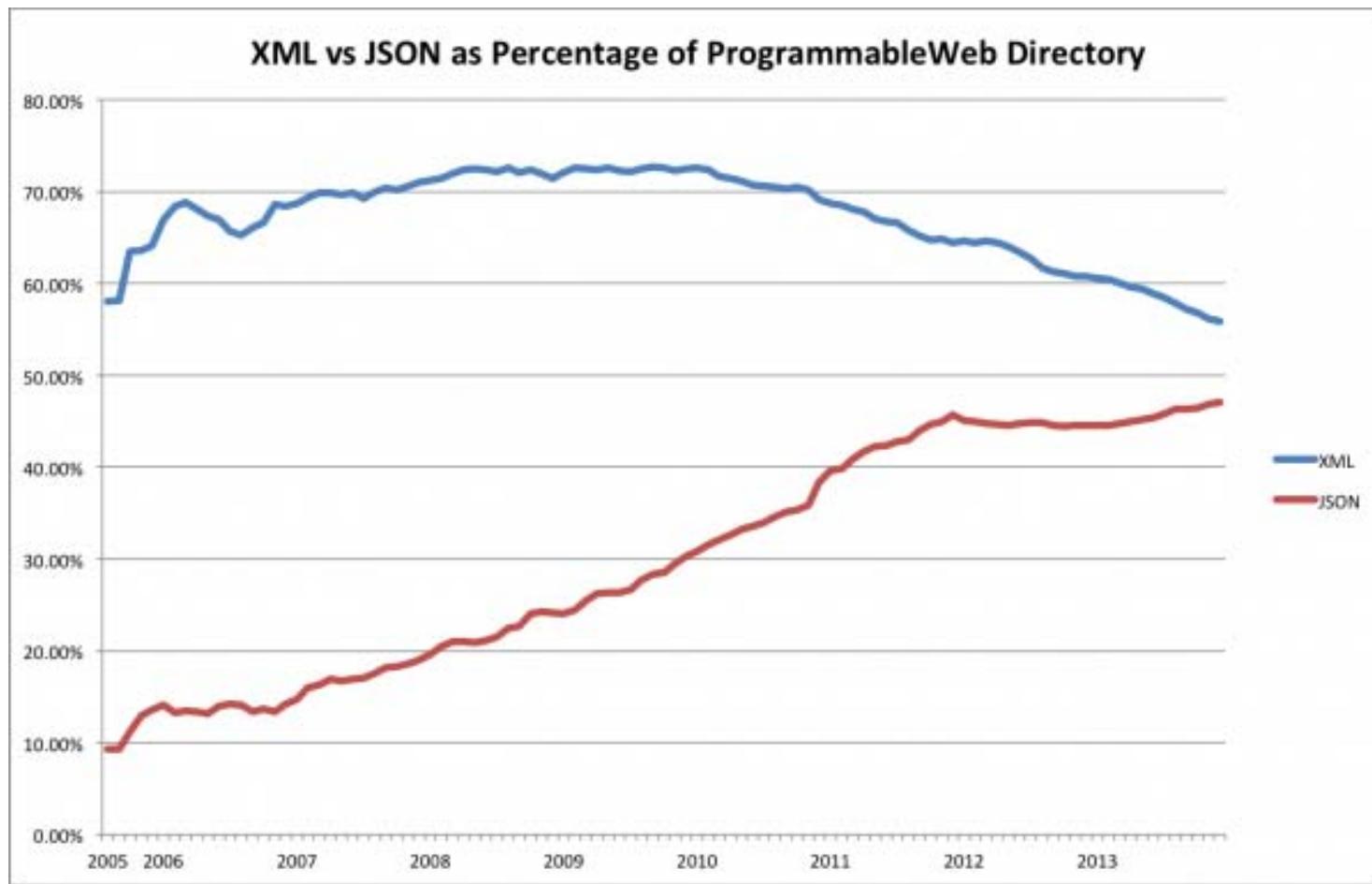
“When it comes to the data APIs serve up, **XML is still the most used format**, but JSON is hot on its heels and growing much faster. Even though there are still more XML APIs, the more recent the API, the more likely it’s serving JSON. In many cases ... **companies are rapidly moving from XML to JSON.**”



Scott Gilbertson, 2011

<http://www.webmonkey.com/2011/03/thousand-of-apis-paint-a-bright-future-for-the-web/>

Why is JSON relevant?



ProgrammableWeb, 2013

<http://www.programmableweb.com/news/jsons-eight-year-convergence-xml/2013/12/26>

Why is JSON relevant?

“Early in the rise of JSON, it seemed the format was popular simply because it was **easier to parse in a browser**. More recently, the focus has been on **XML's greater complexity and larger payload**. ‘JSON more efficient over carrier mobile networks and less memory intensive to process on mobile and embedded devices,’ tweeted Simon Prickett.”

Adam DuVander, 2012

<http://www.programmableweb.com/news/leading-apis-say-bye-xml-new-versions/2012/12/17>

Why is JSON relevant?

“So these days, if you want to **interchange tuples or tables of tuples or numbers and strings**, you have JSON. If you want to do **nontrivial publishing automation**, use XML. If you want to interchange smart bitmaps of page images, there’s PDF. I personally think we’re probably done with inventing low-level textual interchange formats.”

Tim Bray, 2013

<http://www.tbray.org/ongoing/When/201x/2013/02/10/XML-at-15>

Why is JSON relevant?

“Choose wisely. There are good reasons for using JSON, and there are still good reasons for using XML. ... XML’s strength is **extensibility** and the **avoidance of namespace clashes**. ... JSON is purposefully limited and therefore **much lighter** than XML. I suspect that, most of the time, data can be modelled with hashes and lists comprising simple data types, **making JSON the preferred route.**”

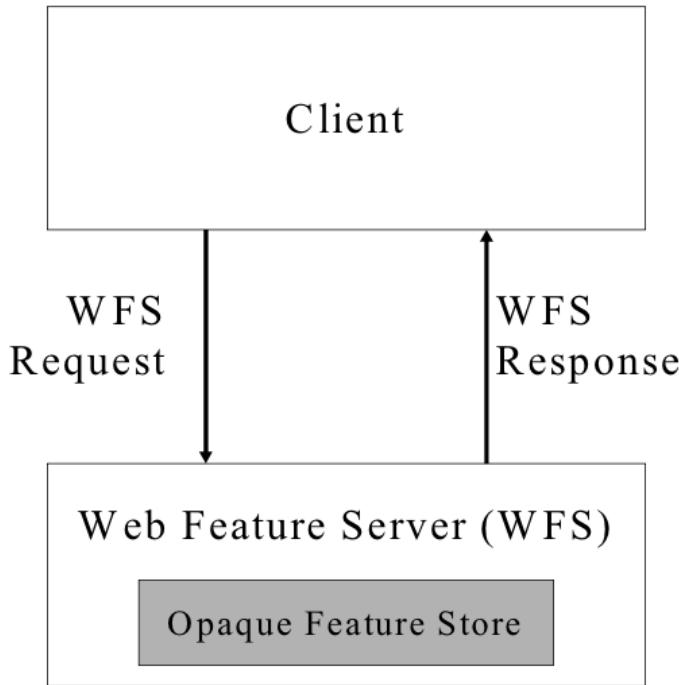
“The biggest reason that JSON is now being used over XML is that JSON is inherently **more efficient**.”

Sean Lindo, 2013

<http://www.programmableweb.com/news/xml-vs.-json-primer/how-to/2013/11/07>

Is there anything that needs to be done to support JSON in OGC standards?

e.g.
GetFeature

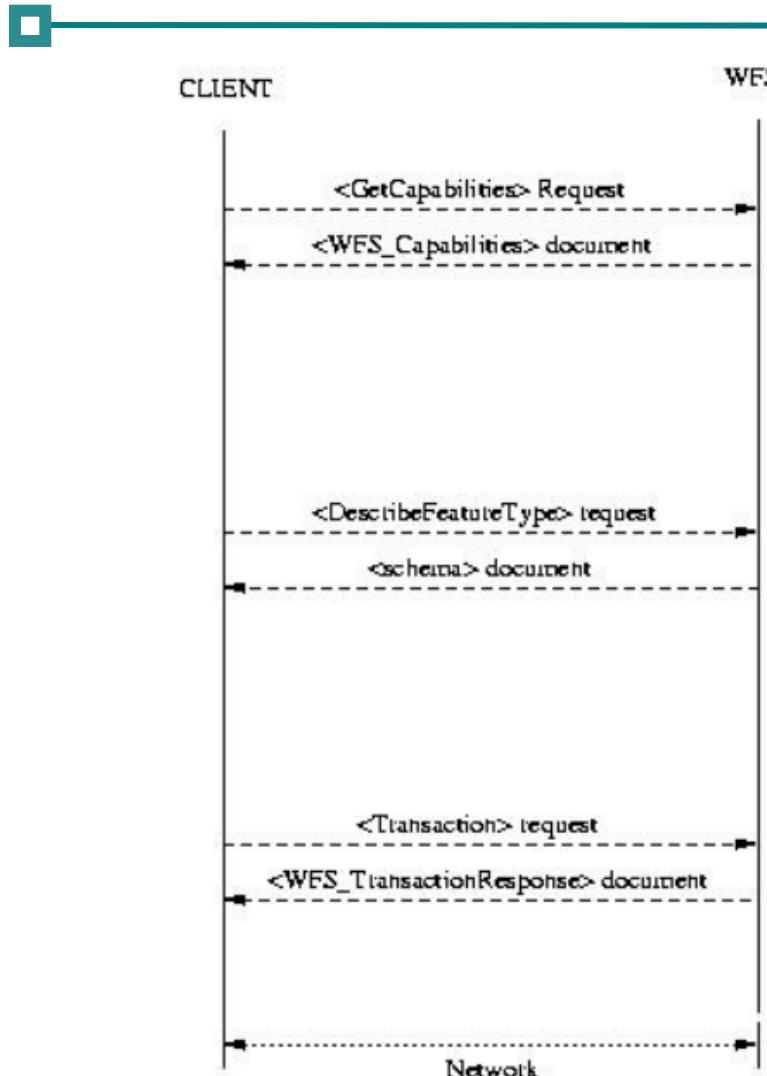


Returns GML, but using the **outputFormat** parameter it can also return JSON, e.g. GeoJSON

Is there anything that needs to be done to support JSON in OGC standards? Yes!

- Most OGC implementation standards are firmly based on
 - XML
 - RPC (Remote-Procedure-Call) patterns
- Over the years OGC and others have built a rich library of XML implementations for many concepts related to geographic information

OGC Web Services typically based on sending and receiving XML messages/documents



How to address?

Define JSON representations for messages/documents?

Concerns:

- A lot of work!
- A one-to-one translation of documents may not be appropriate
- Web developers expect RESTful APIs

Capabilities documents

- The entry point to **all** OGC web services
- Rich description of what this OGC web service offers
- Needs to be understood by a client to correctly construct the next messages in the interaction with the service

Concerns:

- Monolithic, very heavy
- Hard to use for non-OGC-experts as it is not obvious how to construct the next messages – not very web-friendly

Schema documents

- XML Schema (or Schematron, Relax-NG)
- Schema validation is essential for XML as used by OGC
- JSON usually not described by schemas
- Work on JSON Schema exists, but not much used and standardization of the draft is not progressing
- More useful for describing the JSON objects than for validation

XML Path expressions

- In many places in OGC standards use XPath to reference specific elements in an XML document
- Example: Queries
- XPath will likely be ambiguous in a JSON context
- Need to translate path expressions to something that is understood by JSON developers
- No equivalent to XPath in JSON

Linking

- Linking between resources a key concept in many OGC standards
- Usually using XLink
- Until recently no built-in support for links in JSON
- New standard: JSON-LD
- Promising approach that preserves the simplicity of JSON

Exceptions

- Exception messages are sent using XML
- Consistent JSON encoding of exceptions needed

Support for Resources

Resource Type	XML	JSON
Features	GML and GML application schemas	GeoJSON or GeoServices JSON, but restricted to simpler structures Only draft rules for application schemas in JSON exist
Geometry	GML	GeoJSON or GeoServices JSON, but limited to Simple Features geometries
Styling	SLD/SE	GeoServices JSON, but restricted to simpler requirements
Metadata	ISO 19139 XML	?
...	Many XML grammars for geographic resources	Partly GeoServices JSON, but typically restricted to simpler requirements

Automated mapping between XML and JSON and vice versa?

Challenges:

- Different concepts
 - Linking: JSON-LD vs XLink
 - Schema: JSON Schema (?) vs XML Schema / Relax NG
 - Path expressions: ? vs XPath
 - Namespaces not a concept in JSON, but central in XML used by OGC; maybe JSON-LD contexts can be leveraged
- Pre-existing JSON encodings like GeoJSON
- Which conversion rule between XML and JSON to pick: there are several
- JSON derived from XML often feels wrong

The best way to support JSON in OGC?

- Properly JSON-enable existing OGC web service standards or to develop new RESTful APIs?
- Both options are a significant task
- Investigated in the OGC Testbed 11 by Joan Maso

The best way to support JSON in OGC?

My take:

- Separate standards would probably be better
- Service interfaces should be RESTful, closely linked to the OGC strategy on this topic
- Adopt existing practices that work over developing things from scratch
- Consistency with abstract specifications of OGC/ISO required
- Informed by XML-based standards, but avoid legacy and XML artefacts
- Start simple and extend what works

Questions?



Clemens Portele

Managing Director

- Trierer Strasse 70-72, 53115 Bonn, Germany
- +49 228 91410 73
- portele@interactive-instruments.de