

SUPPORT FOR JSON IN ATML-RELATED STANDARDS



AGENDA

- Background
- Relevance?
- Known Incompatibilities
- Potential Path Forward

BACKGROUND

- JSON stands for **J**avaScript **O**bject **N**otation
 - Lightweight format for storing and transporting data
 - Often used when data is sent from a server to a web page
 - "self-describing" and easy to understand
 - A data interchange format.

This example defines an employees object: an array of 3 employee records (objects):

```
{
  "employees":[
    {"firstName":"John", "lastName":"Doe"},
    {"firstName":"Anna", "lastName":"Smith"},
    {"firstName":"Peter", "lastName":"Jones"}
  ]
}
```

RELEVANCE?

- JSON is the preferred data format for communication between services and services and clients
- Converting from JSON to XML causes an additional work when trying to use ATML
 - Potential failure point
- Users have more libraries to process JSON data
- Customers have expressed concern:
 - XML is obsolete hence ATML must be old/obsolete
 - XML is very bulky to process and transfer
 - [JSON:The Fat-Free Alternative to XML](#)

KNOWN INCOMPATIBILITIES

- Known issues
 - Schema-related
 - Choice
 - Extensions
 - <[CDATA[]]>
 - namespace
 - XPATH (JSON-path?), XQuery
 - Comments
 - Others?
- Schema definition

JSON SCHEMA

- [GitHub - json-schema-org/json-schema-spec:The JSON Schema I-D sources](#)
- [JSON Schema | The home of JSON Schema \(json-schema.org\)](#)
- Describes your existing data format(s).
- Provides clear human- and machine- readable documentation.
- Validates data which is useful for:
 - Automated testing.
 - Ensuring quality of client submitted data.
- Active GitHub repo
- Will be published as an [IETF RFC](#), which has already been adopted by the HTTP APIs working group.

SUPPORT IN ALTOVA XMLSPY

- [Altova XMLSpy 2023 Enterprise Edition](#) provides support JSON Schema
- Functionality:
 - Create JSON schemas graphically in JSON Schema View.
 - Validation with the JSONValidator of XMLSpy:
 - Assign a JSON schema to a JSON instance document, and validate the instance document from within XMLSpy
 - Setting JSON validation options.
 - Generating JSON Schema from a JSON Instance
 - Converting between JSON and XML
 - Automatically convert XML schema to JSON schema (with some cleanup)
- Shared by Ion Neag

POTENTIAL PATH FORWARD

- Investigate incompatibilities listed and others?
- Prototype using XMLSpy functionality with existing documents/schemas
- Add a recommended practice standard to convert ATML-related data to JSON and back
 - Define how to convert between XML and JSON formats
 - [Using JSON in the Google Data Protocol | Google Data APIs | Google Developers](#)
 - Will support use of JSON for snippets of ATML Data Model
 - ~~If JSON schema is approved, define a JSON schema with standards~~

ADDITIONAL SLIDES



SHOULD WE SKIP JSON AND PROCESS TO ION???

- [Amazon Ion \(amazon-ion.github.io\)](https://amazon-ion.github.io)
- **Amazon Ion** is a richly-typed, self-describing, hierarchical data serialization format offering interchangeable binary and text representations. The text format (a superset of JSON) is easy to read and author, supporting rapid prototyping. The binary representation is efficient to store, transmit, and skip-scan parse. The rich type system provides unambiguous semantics for long-term preservation of data which can survive multiple generations of software evolution.
- Shared by Ion Neag