# *OAGi Embraces Cloud Computing and Java Script Object Notation (JSON)*

*an OAGi Position Paper*

*Document #20111027V1.0*

*Open Standards that Open Markets™*

## Background

In 1996, when the first release of the Open Application Group Integration Specification (OAGIS) was published, the expression form of the OAGIS data exchange data model was a proprietary meta data format invented by OAGi. The membership invented and refined this meta data to provide more flexibility in the expression of the OAGIS data exchange data model than traditional EDI data structures supported.

Since then, many improved forms of data expression have been developed and the OAGIS technology evolved to support XML Document Type Definitions (DTD) in 1987 and then XML Schema (XSD) in 2002. In 2005, OAGi elected to evolve OAGIS to support a stricter Schema expression as defined by the UN/CEFACT Naming and Design Rules (NDR) and Core Component Technical Specification (CCTS). This change improved the grammar expressions in OAGIS, reduced ambiguity, and improved both intra and inter supply chain interoperability.

This evolution of the technology used to express the data exchange data model that is OAGIS has been important to our users of the standard in order to better support the business solutions they require both outside and inside their organizations.

## Why JSON?

With the move to Web 2.0 and Cloud Computing, a new class of solutions are being built and deployed by organizations like Microsoft, Google, Yahoo, Facebook, LinkedIn and Amazon. Virtually all of the Cloud Computing interfaces (API's) offered by these organizations are using a meta data markup language for data transfers called JavaScript Object Notation (JSON). In fact, JSON has become the defacto data transfer expression language for web applications posting data to a web browser.

Inside the enterprise, enterprises have created and deployed web apps for business solutions such as Enterprise Portals that provide important business intelligence and other services using a sophisticated technique called "Integration on the glass."

"Integration on the glass" is a term used to describe the delivery of business functionality in a real time mode by accessing back-end applications or subsets called applets, comprised of functionality from systems such as Enterprise Resource Planning, Customer Relationship

Management, Supply Chain Management, Human Resources, and Content Management. There will be more on this concept later is the Position Paper.

Beyond “Integration on the glass”, enterprises are using JSON to deploy their own web interfaces for their partners. The use of JSON is also evolving so that it used for internal integration and B2B in certain circumstances. Again, these interfaces are used to complement existing integration and data transfer practices.

## What is JSON?

JavaScript Object Notation (JSON) is a lightweight text-based open standard designed for human-readable data interchange. It is derived from the JavaScript scripting language for representing simple data structures and associative arrays, called objects. Despite its relationship to JavaScript, it is language-independent, with parsers available for most languages.

The JSON format was originally specified by Douglas Crockford, and is described in IETF RFC 4627. The JSON format is often used for serializing and transmitting structured data over a network connection. It is primarily used to transmit data between a server and web application. JSON was used at State Software, a company co-founded by Crockford, starting around 2001. The JSON.org website was launched in 2002. In December 2005, Yahoo! began offering some of its web services in JSON. Google started offering JSON feeds for its GData web protocol in December 2006.

Although JSON was based on a subset of the JavaScript scripting language (specifically, Standard ECMA-262 3rd Edition—December 1999) and is commonly used with that language, it is a language-independent data format. Code for parsing and generating JSON data is readily available for a large variety of programming languages. json.org provides a comprehensive listing of existing JSON libraries, organized by language.

*Source: Wikipedia*

## EDI, XML, and JSON as Complimentary Technologies

When XML first started to appear in technical solutions in 1997, there were some who predicted that EDI was "dead." Yet 14 years later, EDI is still going strong. It used to be a standing joke that EDI would probably go away about as fast as the mainframe did, making reference to the early days of client server technology.

A July 28, 2011 article in the Wall Street Journal recently reported that IBM mainframe sales were up 61% in the second quarter. In fact, there have been very few cases where XML replaced EDI solutions that were already in place. Many industries around the world still rely on EDI technologies as their primary form of B2B.

XML did became very successful for B2B use in industries such as Chemical, High Tech, and Automotive Retail and also for use in Enterprise Integration, where there was not a clear solution and therefore it filled in the gaps where EDI had not penetrated or was not the right solution.

This same adoption pattern is also taking place with JSON complementing XML based solutions. JSON is very useful for the next wave of web applications and solutions and the Web 2.0 world. Its success came organically because it is easy to use and is native to one of the most used web programming languages, JavaScript.

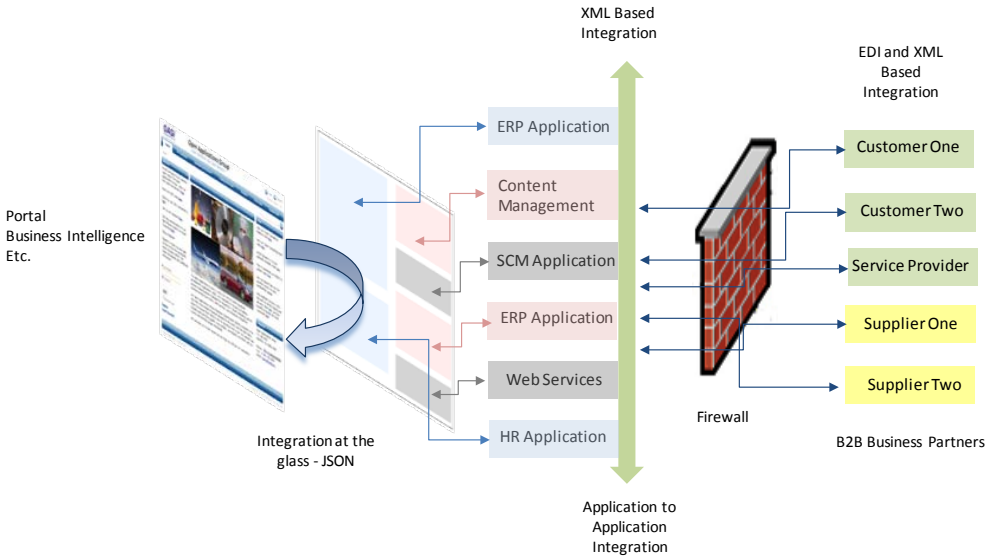


**Figure One – EDI, XML, and JSON in complementary roles**

## OAGi Embraces JSON and Cloud Computing

As part of our Cloud Computing Initiative, where OAGi members are researching technologies to enable our users of OAGIS to ensure support of Cloud Computing, OAGi is embracing the use of JSON for OAGIS instances on the wire and will begin shipping example instances of OAGIS BODs expressed as JSON.

OAGi will defer the delivery of JSON Schema as it is not complete and still under debate in the wider technical community. OAGi is leaving this issue open as a possible future OAGIS deliverable should this situation changes.

XSD will remain the normative form for the OAGIS Standard and the OAGIS data model will still be delivered in the XML Schema Form. In addition, OAGi will continue to ship XML instances with the OAGIS standard.

## Learn More

The OAGIS® Standard is most widely used and richest business language for data exchange in the world. It is used by thousands of organizations for Cloud Applications, Business to Business (B2B), Web Services, Service Oriented Architecture (SOA), Master Data Management (MDM), and Application to Application (A2A) interoperability. It is free to all to download at www.oagi.org.

Membership in the Open Applications Group organization provides many benefits to solution providers and End-Users alike through teaching, sharing of best practices, leading the way in building content and technical solutions and a community for all who need to solve problems or sell product in the Business Software Interoperability Marketplace.

For more information, please visit us on the web at www.oagi.org, or contact us directly through email at membership@oagi.org.