Semantic Web Glasses and Semantic Resolvers

Nikita Ogievetsky
Cogitative Technologies,
Cogitech Inc.
Hewlett, NY 11557, USA
E-mail: nogievet@cogx.com

Abstract

In this paper, author proposes a solution to make URI Resolver class available in different programming languages. The Semantic Resolver derived from a corresponding .NET or Java classes resolves resources using semantics rather than physical mapping. When a Web Service consumer tries to parse information retrieved from a web service, Semantic Resolver based on XML Namespace and other indicators seamlessly translates it into a format understandable by the application.

1. Introduction

Consumer Applications in the SOA world receive information via Web Services. However, the information that an application needs might be available in a format that the application does not understand out of the box and needs to be either discarded or translated to the native for Web Service consumer format.

On today's web the information in the expanding semantic universe exists in numerous internationally adopted and proprietary formats. Information publishers rush to take their information assets out to the market and some times may take little time to diligently research the best fit among information standards. They might mistakenly adopt a model created for a slightly different purpose. Developers who are writing web services may reuse data models from their previous projects because they are familiar with it, not because it fits the purpose. Lastly, they can simply reuse data model of an underlying database. The more desirable their information assets are, the easier it is for them to get away with something proprietary or inconsistent. This situation arises also when there are several competing standards, as for example was the case with knowledge management standards (for example, RDF, Topic Maps) and some financial standards. In the latter case the industry constantly tries to unify standards, to increase interoperability between the stakeholders.

I had a very interesting experience related to Media extensions for Really Simple Syndication (RSS) standard. RSS is an XML format for syndicating digests of web resources. Simply speaking, RSS feed represents a response to a Web query. For example, RSS responses to such queries as: "What's new on your Web site?", "What is the latest news about global warming?", "What is the latest sport news?" will result in an XML document wrapping a set of information items containing digests of corresponding web resources. One can subscribe to an RSS feed from your favorite website and receive news and updates using one of the freely available RSS readers. Publishers strive to produce valid RSS because it gives them an easy way to advertise and deliver content to consumers

RSS is often enriched with properties from other standards. For example, Dublin Core [2] is often used to express metadata and Creative Commons [1] might be used to express copyright licenses and related metadata, etc. Even though there is sometimes an overlap between metadata standards, it might be tolerable because the data itself is represented in a standard way. With video blogging, and video sharing gaining momentum people became interested in adding media related extensions to RSS fids. Recently Yahoo proposed Media RSS extension to RSS standard, and Apple independently introduced iTunes RSS module to express similar set of properties. Currently publishers use one or the other, mix of iTunes and MRSS elements or just invent properties of their own and hide them under iTunes or Media RSS namespaces. The lack of a standard schema for media assets makes it harder to exchange information via Web Services.

In 2000, I suggested the notion of Semantic Web Glasses (SW Glasses): a piece of software that translates information from one format to another. As an example I used two competing at that time knowledge representation standards: XML Topic Maps (XTM) and Resource Description Framework (RDF). Presentation showed how SW Glasses enabled RDF - aware applications to work with information expressed in XTM or other knowledge representation formats, and vice versa.

Semantic Web Glasses could be classified as either spy-glasses - for applications that want to consume as much information as possible, or non-intrusive SW Glasses - for applications that want only the information that was specifically intended for them. Non-intrusive SW Glasses are valuable in the consumers market – where publishers compete to win as many consumers as possible. Spyglasses on the other hand might be used in the publishers market – where consumers are hungry and would like to devour any information that they could get. Non-intrusive SW Glasses may also be used when collecting very specific data that needs not be confused with anything else. Depending on their purpose, SW Glasses can perform different types of semantic transformations:

- Syntax translation – translates XML documents from one XML syntax to another.
- Syntactic filtering - removes invalid, corrupted or otherwise incomprehensible pieces of information.
- Semantic filtering - removes pieces of information that have no value to consuming applications.
- Conceptual filtering - removes those pieces of information that consuming applications cannot handle.
- Information enriching - expands data with relevant information.
- Information localization - translates numerical measures and formats from one system to another.

In the heart of SW Glasses work lays semantic mapping. Translation between two information representations is impossible if underlying models are not mapped. Different people or vendors might have different ideas of how this translation must be implemented, which creates a market for SW Glasses implementations. Some times there is no right way, and different approaches must be used depending on the context, as we would use dark glasses on the beach and orange goggles when skiing downhill.

Next comes a question: how do we “hold” semantic glasses in front of Web Service consumers? This presentation describes the machinery for integrating Semantic Web Glasses in a friendly and transparent for the consumer application way.

The proposed solution uses URI Resolver class available in different flavors in Java and .NET worlds. Developers often use URI Resolver to map and resolve an abstract identifier or a URN to a physical path from which a corresponding resource could be retrieved. Semantic Resolver derived from a corresponding .NET or Java classes resolves resources using semantic rather than physical mapping. When a Web Service consumer tries to parse information retrieved from a web service, Semantic Resolver based on XML Namespace and other indicators seamlessly translates it into a format understandable by the application.

References

[1] http://www.creativecommons.org

[2] http://dublincore.org/documents/dces