Getting Started With LINQ to RDF

Semantic web applications in .NET

This short document shows you the steps needed to get LinqToRdf up and running. It covers the tools and systems prerequisites, the techniques and the expected behavior. All examples are in C#, although LinqToRdf will work on all .NET language.

Andrew Matthews

6/19/2007

Getting Started With LINQ to RDF

Semantic web applications in .NET

What are the components of a semantic web application?

Within the context of this document, Semantic Web Application means "*application that uses RDF and related technologies*". That is – an application that represents information as a graph structure using RDF. It's beyond the scope of this document to explain the whole pyramid of standards and technologies needed to support the semantic web. Instead I'll give enough context for you to know what steps are required to get your Semantic Web Application working.

Figure 1 Major technologies employed with LinqToRdf

LinqToRdf uses the SemWeb class library by Joshua Tauberer, which provides a platform for working with OWL and SPARQL. It uses the .NET 3.5 namespace System.Query which will be released as part of Visual Studio .NET 2008.

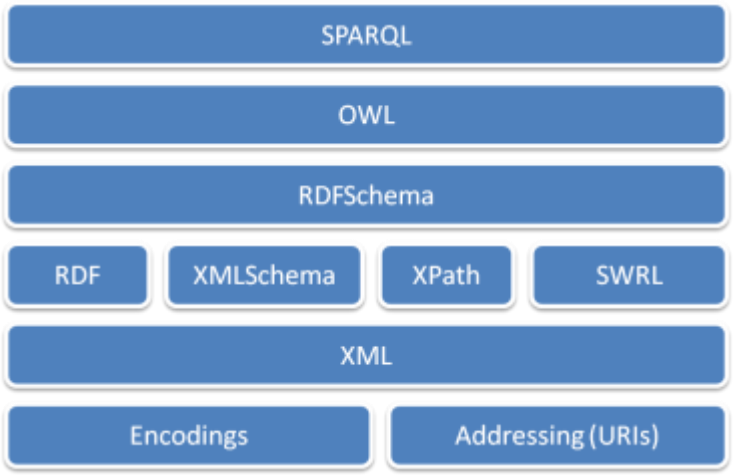

Figure 2 The semantic web technology stack

Figure 2 above shows the hierarchy of technologies that are involved in the semantic web.

What do you need to do semantic web programming with LinqToRdf?

LinqToRdf requires an IDE that supports the latest version of LINQ, which at the time of writing was Visual Studio .NET Orcas Beta 1. The LINQ framework is still subject to change, and as it stabilizes, the LinqToRdf library will be updated to reflect the new changes.

The

Where to get LinqToRdf

Installation procedure

Creating an Ontology

Hosting your ontology

Linking to the ontology from .NET

Querying the ontology using SPARQL

Reference Material