Knowledge Control System - The Backbone Of Ontology Middleware

Atanas Kiryakov[1], Damyan Ognyanov[1], Kiril Simov[2,1], Borislav Popov[1], Stanislav Jordanov[1]

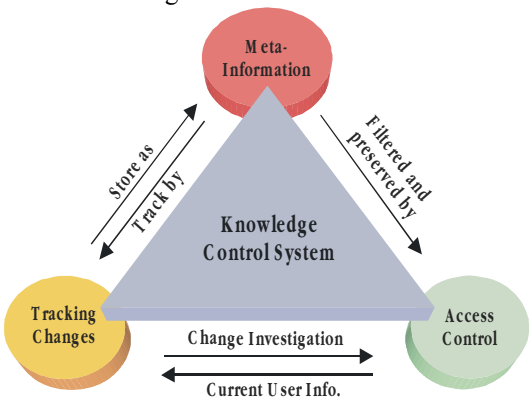
[1] OntoText Lab, Sirma AI EOOD, 38A Chr. Botev blvd, 1000 Sofia, Bulgaria
{naso, damyan, borislav, stenly}@sirma.bg
[2] Linguistic Modelling Lab, CICT, Bulgarian Academy of Sciences, 25 Acad. G. Bontchev str, 1113 Sofia, Bulgaria
kivs@bgcict.acad.bg

Introduction

Our group is currently working on a Knowledge Control System (KCS) which is considered a backbone for robust ontology middleware. The KCS is a part of the On-To-Knowledge Project and uses Sesame as a repository access layer. The following features have been considered:

- Versioning (tracking changes) of knowledge bases;
- Access control (security) system;
- Meta-information for knowledge bases.

These three aspects are interrelated as depicted on the following scheme.



Tracking Changes, Versioning

We aim to provide versioning of RDF(S) on a structural level in the spirit of the software source control systems. The main principles are outlined below:

VPR1: *The RDF statement is the smallest directly manageable piece of knowledge.*

VPR2: *An RDF statement cannot be changed – it can only be added and removed.*

VPR3: *The two basic types of updates are addition and removal of a statement*

VPR4: *Each update turns the repository into a new state*

Versioning of RDF(S) repositories, batch updates, versioning of imported and inferred statements, versioning of meta-information, versioning of knowledge represented in files are the *requirements* for the KCS.

Security And Access Control Over RDF(S) Repositories

The access control system proposed has to be universal and flexible as well as simple and intuitive, applicable for both RDF(S) and DAML+OIL. It should provide native support for restrictions which are typical for knowledge management and engineering tasks. Such are restrictions over the ontology (or schema) part of the repository, over instances of certain classes, over relation (or properties) of certain types. The following **restriction types** to be supported, according to the data they describe and the way they are defined: **Repository, Schema, Classes, Instances, Properties, Pattern, Query.**

The following **Rights** will be supported: **Read, Add, Remove, Admin, Clear,** and **History.** ***Roles*** *will be supported. Each role is defined as a set of* ***Security Rules*** *and other roles.*

There will be *three levels of compelxity* supported: **Standard** – the basic restriction types (without security classes and query restrictions); **Extended** - involving Security classes; and **Unrestricted** involving restrictions of type **Query**. Those restrictions require a separate query evaluation each time they have to be checked.

All the data necessary for the knowledge control system (KCS) can be formally represented in RDF according to the schema
http://www.ontotext.com/otk/2002/03/kcs.rdfs

Meta-information – versioning and access control

Both versioning of and access control over meta-information are provided.

For more detailed description of the KCS and ontology middleware please refer to [1].

References

1.Atanas Kiryakov, Kiril Iv. Simov, Damyan Ognyanov.

Ontology Middleware: Analysis and Design. Deliverable 38, On-To-Knowledge project, March 2002. http://www.ontotext.com/publications