The **SPARQL-RW Framework**

The **SPARQL-RW (SPARQL ReWriting) Framework** provides a generic method for SPARQL query rewriting, with respect to a set of predefined mappings between ontology schemas.

**Key Features**
- **Modeling Mapping**
  - Specifies mapping types which can be exploited by the SPARQL query rewriting process
  - Formally described using Description Logics
  - Supports Flexible & Rich Mappings
- **Query Rewriting Method**
  - Based on a set of predefined mappings between ontologies
  - Semantics Preserving
- **Java System Prototype**

**Architecture**

**Problem Definition**

Let a source ontology $O_s$, a target ontology $O_t$, and a set of mappings $M$ between $O_s$ and $O_t$. Our framework takes as input a SPARQL query $Q_s$ expressed over $O_s$, and rewrites it to a semantically correspondent SPARQL query $Q_t$ (expressed over $O_t$) w.r.t. $M$.

**Intro**
- Describes mappings between OWL/RDF-S ontology schemas
- Defines and supports all the possible mapping types which can be exploited by the SPARQL query rewriting process.
- The supported mappings are highly dependent to the SPARQL expressiveness.

**Grammar**
- Four basic notions:
  - **Class Expression**
  - **Object Property Expression**
  - **Datatype Property Expression**
  - **Instance Expression**
- Expressions are defined using several operators: **unions** ($\cup$), **intersection** ($\cap$), **composition** ($\circ$), **inverse** ($\sim$), **domain/range restrictions**, etc.
- Supports $N:M$ cardinality mappings using **equivalence** or **subsumption** relationships.

**Semantics**

Formal semantics based on Description Logics.

**Overview**
- The SPARQL query rewriting process lies in the query's graph pattern rewriting.
- For each **Basic Graph Pattern**, rewrite each **triplet pattern** $a)$ by **Predicate** $b)$ by **Object** $c)$ by **Subject**.
- The rewriting is independent of the **query type** (i.e., **Select**, **Ask**), the SPARQL **solution sequence modifiers** (i.e., **Order By**, **Distinct**) and the SPARQL **algebra operators** (i.e., **Union**, **Optional**).
- **Triple Patterns Types**
  - **Data Triple Patterns**: Deal with data and not schema info.
  - **Schema Triple Patterns**: Deal with schema info. Contain RDF/RDFS/OWL terms.

**Motivating Example**

**Family Framework**

The **SPARQL2XML Query Framework**

Bridging the Gap between the XML and the Semantic Web Worlds.

- **XML Schema to OWL Transformation** (Xs2OWL plug-in)
- **Mapping Specification & Generation between OWL - RDF/S and XML Schemas**
- **SPARQL to XQuery Query Translation**
- **XML – RDF Data Transformations**