From Web 1.0 → 3.0: Is RDF access to RDB enough?

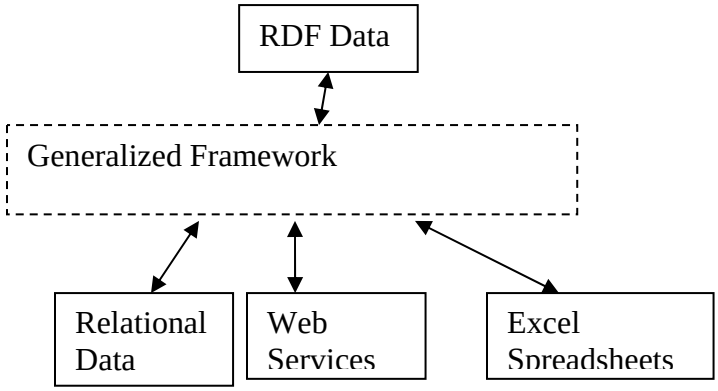
Vipul Kashyap, Senior Medical Informatician, Partners Healthcare System, vkashyap1@partners.org
Martin Flanagan, CTO, InSilico Discovery, mflanagan@insilicodiscovery.com

Abstract:

There have been a host of initiatives led by the W3C to create standards and infrastructure that enable the transition from the current web to the semantic web. In this paper, we present a position that argues for a generalized approach for RDF access to; (a) Relational Databases; (b) Web Services; and (c) Tabular data sources such as Excel Spreadsheets. A Life Sciences use case is presented and the use of a common framework is demonstrated through illustrations from a working system. We believe that the effort in evolving a common framework is incremental, and extends the reach and value of Semantic Web technologies in the Healthcare and Life Sciences.

1. Introduction

The wide variety of life science data using different types of data storage and access schemes are a barrier to understanding of biological knowledge and its application to clinical research and practice. For example how does software discover the biological reasons explaining the statistical correlations observed between multiple genes in different types of studies? A large amount of biological data is currently available through web-based data repositories and web services [1,2]. The value of this information would increase dramatically if they were accessible to systems which might not directly support the native data storage format or access mechanism, but which do support RDF [3]. There is a need for a generalized framework for providing RDF access to various types of data storage and access schemes.



We believe that the effort for evolving a generalized framework is incremental and can be integrated with the efforts to create RDF access to Relational Databases. We now discuss a use case from the healthcare and life sciences and demonstrate through illustrations of a working system, how this common framework is achieved today. A live demonstration of the system will be presented at the workshop, if selected for the workshop.

2. Use Scenario: Biological Explanations for Statistical Correlations

This use case scenario is based on a typical Life Science Researcher, Professor Genomus who performs various types of gene correlation studies and experiments. One of the key motivations of Professor Genomus is to determine the possible biological explanations of the correlations observed between two biological artifacts, for e.g., a Gene and a SNP Variant; or understand the clinical implications, for e.g., whether a Gene or a Variant is implicated in some disease. Some typical questions that a professor might answer along with different types of data sources that might contain a potential answer are:

a) What is the location of a given Gene, e.g., CPNE1 on the Human Genome?
   Data Repository: NCBI Entrez, **Access Mechanism:** Web Services
b) For what gene(s) is a given SNP, e.g.., rs6060535 in the upstream regulatory region?
   Data Repository: RDBMS containing dbSNP and regulatory region data,
   Access Mechanism: JDBC/SQL
c) What genes have been found to be "coexpressed" with CPNE1 and in what study?
   Data Repository: Excel Spreadsheet containing the co-expression patterns of various genes in various studies.
   Access Mechanism: .NET API, MS Office API

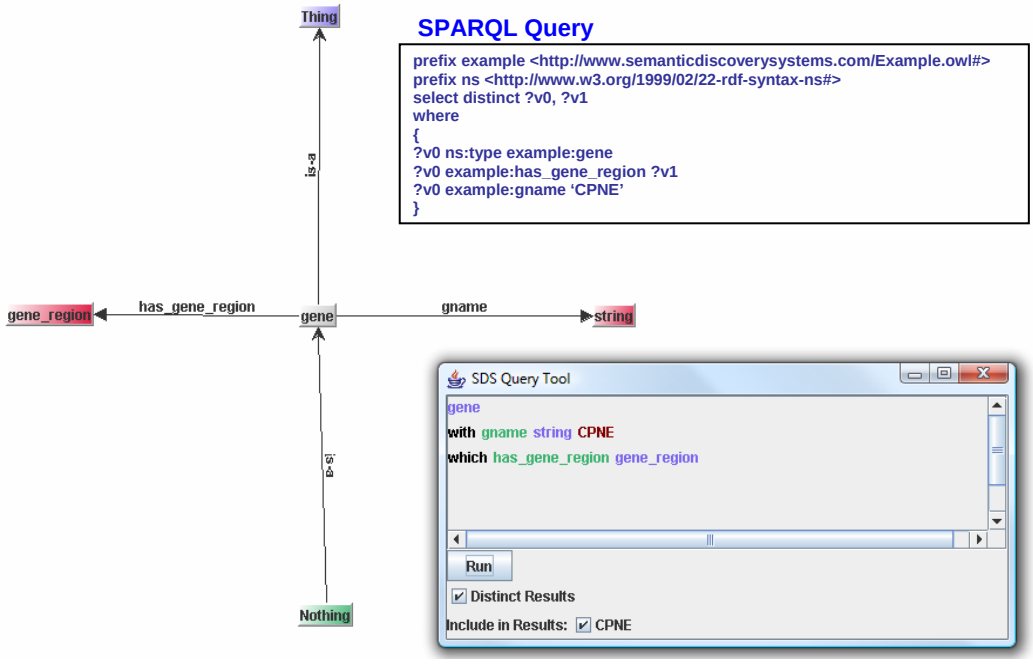
We now demonstrate a solution approach and mapping framework. Given the theme of the workshop, we will focus on a simple example that retrieves information about gene regions and use it to illustrate the mapping framework.

3. Solution Approach

The various steps in the solution approach are presented below.

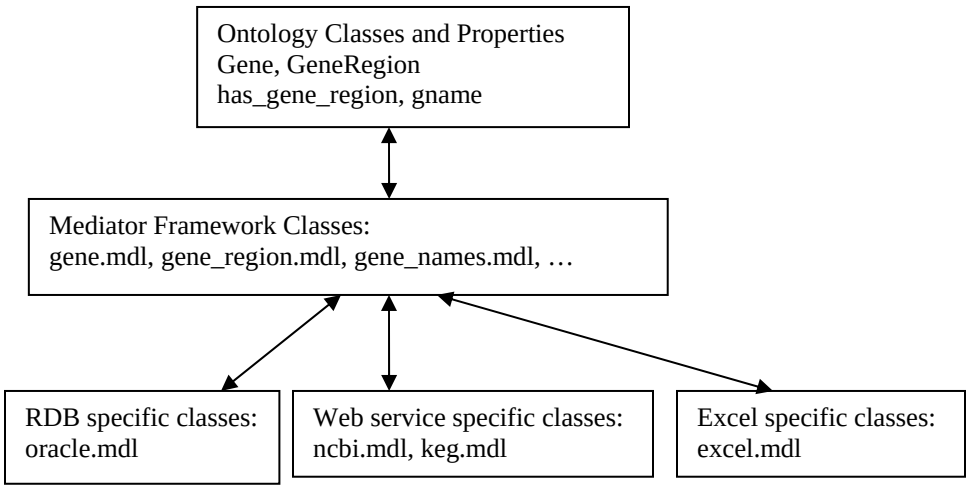
3.1 Ontology-based Query Specification

Professor Genomus invokes the Semantic Discovery System (SDS) [4] and as illustrated below, specifies a query based on the ontology of his interest.



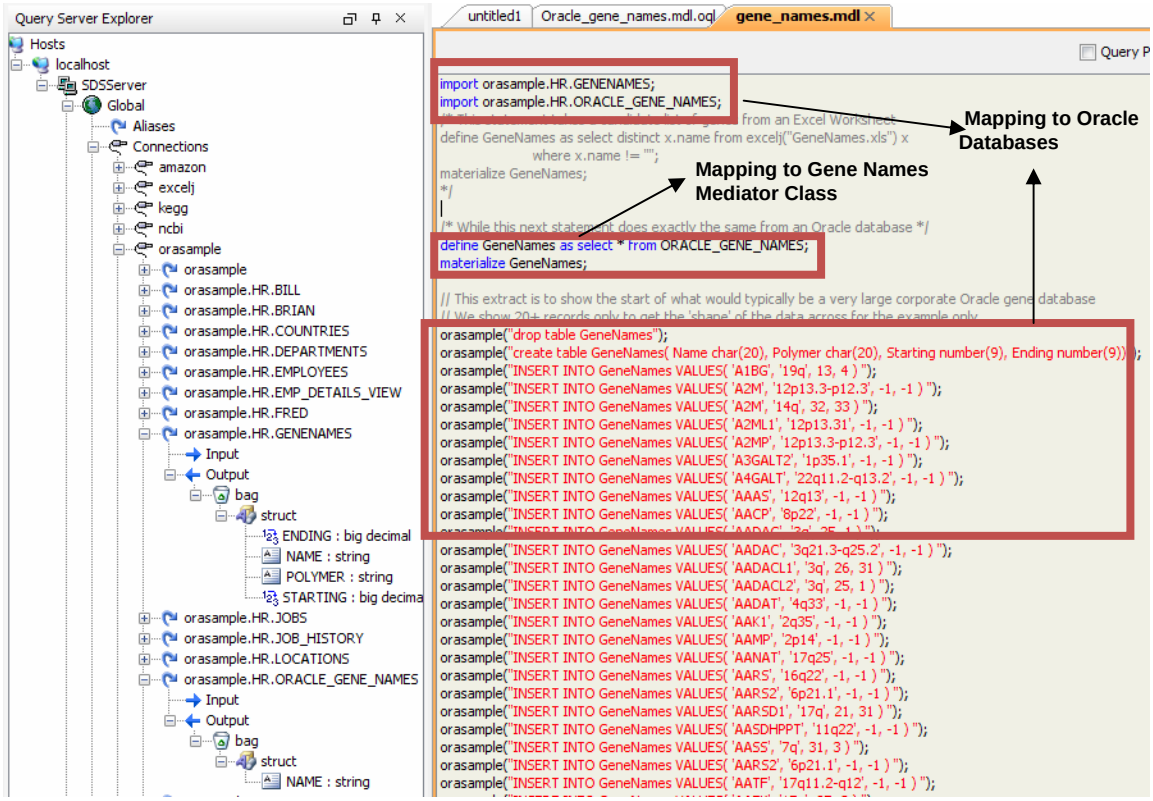
3.2 Mapping Framework

Professor Genomus then gets in touch with Mr. Information Architect and asks him to include the relevant data sources into the system. Mr. Information Architect taps into the generalized framework [5.6] supported by the SDS as illustrated below. A detailed step through of the mapping framework is available at [7].



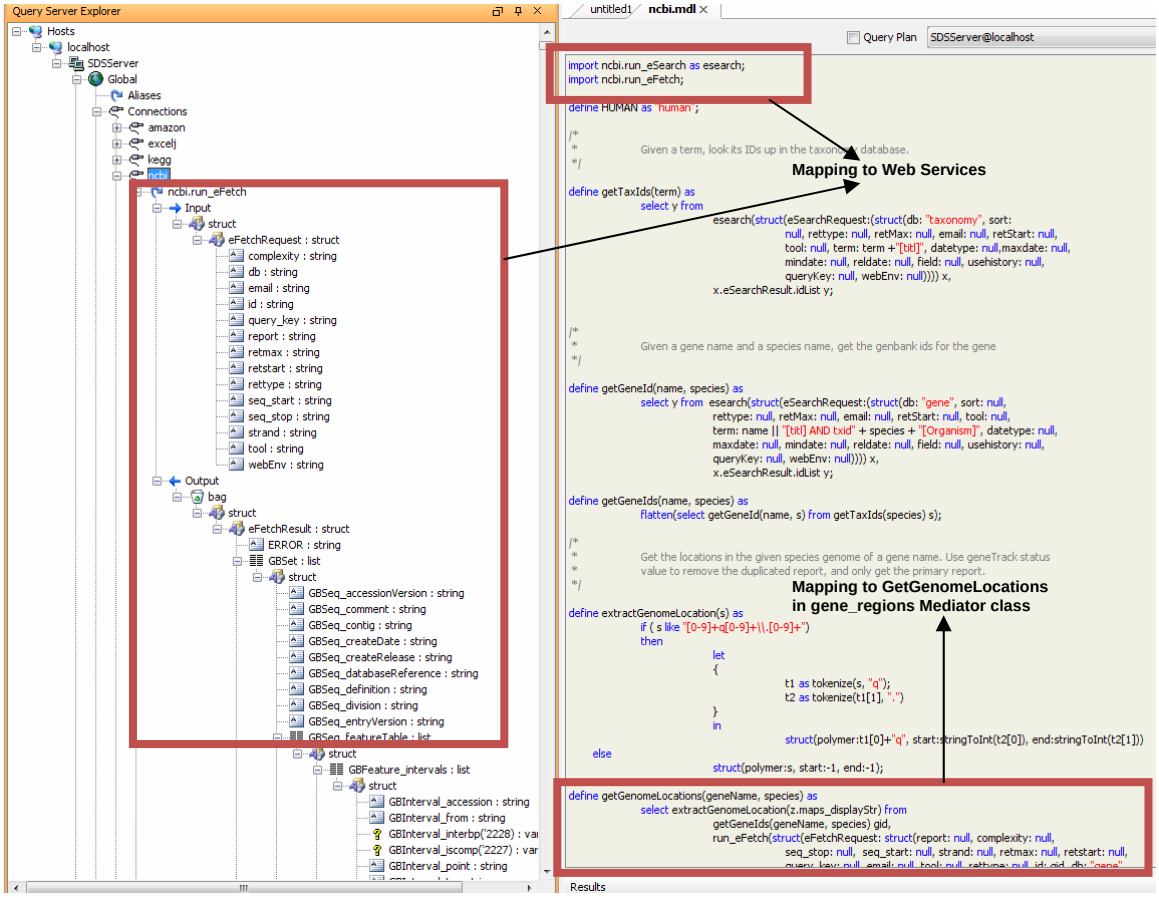
3.2.1 Mapping to Relational Databases

Mr. Information Architect maps the mediator class to a relational database (Oracle).



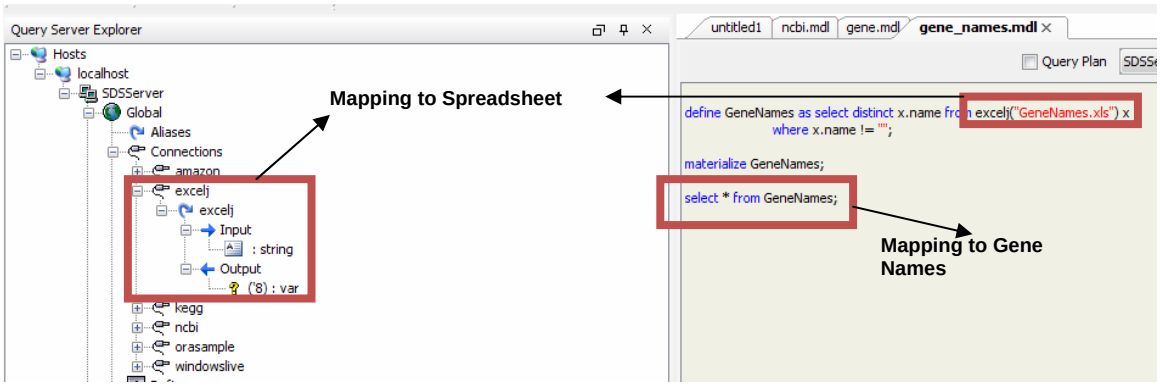
3.2.2 Mapping to Web Services

Mr. Information Architect maps the mediator class to a web service.



3.2.3 Mapping to Excel Spreadsheets

Mr. Information Architect then maps the mediator class to an excel spreadsheet.



3.3 SPARQL Translation to Access Mechanisms and Execution

Once the mappings from the mediator classes (and hence to ontological elements) to the underlying data sources have been established, the SPARQL query is automatically translated into the local query languages and access mechanism supported by the data sources. For instance in the example discussed above:

- The Oracle database executes a SQL query of the form: `SELECT * FROM GENE_NAMES`
- The NCBI web service, `run_efetch` is invoked
- The MS Office API is invoked to retrieve the spreadsheet

After the data is retrieved from the underlying data sources, the generalized mediator framework is responsible for merging the retrieved data. It also optimizes the query processing to return results in a reasonable amount of time.

4. Conclusions

In this paper, we have proposed that the mapping between RDF data to RDB storage/access be viewed as a special case of a generalized mapping mechanism which seeks to map ontological concepts to data retrieval and access via RDBs, Web Services and Structured Tabular data as exemplified by Spreadsheets. A system that supports this mapping framework is also presented. The central vision is one of enabling point and click mapping using a GUI to map ontological classes and properties to different types of data sources. Some of these interfaces enable functionality for a subject matter expert such as Professor Genomus, whereas other interfaces would enable functionality for an architect such as Mr. Information Architect. The key to building out such user interfaces and web-enabling them is the design of a generic mapping framework to handle different types of data resources on the web. We believe RDBs, web services and structured tabular data sources such as Excel spreadsheets are easily incorporated into a generalizable framework as demonstrated by the system discussed in this paper. The longer term goal is to incorporate frameworks for semi-structured and unstructured content such as GRDDL [8] and GATE [9], a framework for NLP functionality.

References

[1] http://www.ncbi.nlm.nih.gov/entrez/query/static/esoap_help.html
[2] http://www.genome.jp/kegg/soap/
[3] http://www.w3.org/RDF/
[4] http://www.insilcodiscovery.com
[5] A Guide and Reference Manual to the SDS SQL and MDL languages. http://www.insilicodiscovery.com/v2/index.php?option=com_kb&Itemid=149&page=articles&articleid=44
[6] A Guide to the operating principles and administration of the SDS Server, http://www.insilicodiscovery.com/v2/index.php?option=com_kb&Itemid=149&page=articles&articleid=46
[7]http://www.insilicodiscovery.com/v2/index.php?option=com_content&task=view&id=82&Itemid=151
[8] http://www.w3.org/TR/grddl/
[9] GATE: A General Architecture for Text Engineering, http://gate.ac.uk