Semantic Representation and Querying of caBIG Data Services

E. Patrick Shironoshita[1], Ray M. Bradley[1], Yves R. Jean-Mary[1], Thomas J. Taylor[1], Michael T. Ryan[1], and Mansur R. Kabuka[1,2]

[1] INFOTECH Soft, Inc
9200 S. Dadeland Blvd, Suite 620 Miami, Florida 33156
[2] University of Miami, Coral Gables, Florida, 33124
{patrick,rbradley,reggie,thomas,mryan,kabuka}@infotechsoft.com

Abstract. A computational grid infrastructure for biomedical research, called caGrid, is under development by the National Cancer Institute (NCI) as part of the cancer Biomedical Informatics Grid (caBIG) Initiative. In this paper we present a model that enables users to query an integrated view of caBIG data services at a conceptual semantic level. The model is based on semCDI, a formulation to generate an ontology view of caBIG semantics and pose queries against this view using the SPARQL query language complemented with Horn rules. We present here a mechanism to process these queries algebraically using our semQA query algebra extension for SPARQL, in order to create sub-expressions for each data service. We then show how resulting graphs from these sub-expressions are then merged using Horn rules.

1 Introduction

As the amount of publicly available bioinformatics and genetics data grows larger, the opportunities available to cancer biology researchers increase immensely. However, with these opportunities come additional challenges with respect to storing, retrieving, and analyzing this data [2]. It becomes essential to utilize semantic representation of the information stored in multiple data sources in order to define correspondence between entities, resolve conflicts between sources, and automate the integration process [8]. Interoperability for data representation and management can be improved as is necessary by using knowledge representation techniques that can describe the semantics of the data [13].

At the forefront of advancing the technology and implementation of semantic representations and collaborative environments is the National Cancer Institute (NCI) through its cancer Biomedical Informatics Grid (caBIG) program. caBIG is developing standards and guidelines along with data and analytical services that can be accessed and utilized through open-source software tools, all within a grid-based architecture referred to as caGrid [9].

We have developed semCDI as a model that enables users to query an integrated view of caBIG data services at a conceptual semantic level [10]. It allows researchers to utilize a single conceptual representation of the data instead of the various and

A. Bairoch, S. Cohen-Boulakia, and C. Froidevaux (Eds.): DILS 2008, LNBI 5109, pp. 108–115, 2008.
© Springer-Verlag Berlin Heidelberg 2008

distinct domain models defined by each of the underlying data services. In this paper, we present an extension to semCDI to demonstrate the manipulation of queries into sub-expressions for specific caBIG data services using semQA, a SPARQL query algebra extension [11], and the merging of results from these sub-expressions through the application of Horn rules.

Table 1. SPARQL query for genes related to the EGF pathway within caBIG

```
SELECT ?symbol ?gene_name
  ?org_name ?chr_map ?cluster_id
FROM cabio, cafe
WHERE {
 ?gene rdf:type :Gene;
  :geneSymbol ?symbol;
  :genePathway ?pathway;
  :geneOrganism ?org .
  OPTIONAL {?gene
    :geneName ?gene_name }
  OPTIONAL {?gene
    :geneLocation ?chr_map }
  OPTIONAL {?gene
    :geneClusterId ?cluster_id }
  ?org :organismName ?org_name .
  ?pathway
    :pathwayName ?pathn .
  FILTER (?pathn="h_egfPathway")
}
```

Table 2. Horn Rules for organism and gene equivalence

```
<!—rule 1-->
Organism_has_Scientific_Name
 [ rdf:type ->
 owl:InverseFunctionalProperty]
<!—rule 2-->
Forall ?x ?y
 ?x = ?y :-
 And(
  ?x [ rdf:type -> :Gene ]
  ?y [ rdf:type -> :Gene ]
  Exists ?z ?w (
   And (
    ?x [ :geneOrganism -> ?z ]
    ?y [ :geneOrganism -> ?w ]
    ?z = ?w
   )
  )
  Exists ?a ?b (
   And (
    ?x [ :geneSymbol -> ?a ]
    ?y [ :geneSymbol -> ?b ]
    ?a = ?b
   )
  )
 )
```

2 Background

Interoperability is addressed by caBIG using a design consisting of a syntactic layer and three semantic layers. At the syntactic layer, interface integration is handled. The semantic layers are organized as follows: first, the controlled terminology layer is maintained in the NCI Thesaurus [3], a reference terminology published by the Enterprise Vocabulary Service (EVS). It includes a list of all concepts that the caBIG semantic structure recognizes. Each of these concepts is tied to one or more common data elements (CDEs). A CDE identifies a property that can be associated with a concept; it also assigns a value restriction or value domain to that property. The third semantic layer of caBIG, the domain model layer, is data source driven, meaning that it is a collection of UML models of the caBIG compliant data services. These models are used to bind the data source metadata to caBIG's concepts and CDEs. The domain models and CDEs are contained within the cancer Data Standards Repository (caDSR). The caBIG program supports an increasing number of tools and datasets of interest to cancer research. Data services in caBIG are compatible because the elements of the underlying data sources are mapped into domain object models that are annotated to offer semantic meanings as provided in the caDSR and EVS [9]. The

data services also require querying to be performed using CQL, an XML-based caGrid query language.

semCDI is a query formulation that defines an ontology view of caBIG semantics, where terminology concepts and domain model classes are modeled as ontology classes, associations between doman model classes are represented as object properties, attributes encoded in CDEs are modeled as datatype properties, and data objects are modeled as OWL individuals members of the corresponding domain model class. semCDI then uses the SPARQL query language [6] as the formulism to pose queries against this ontology view. Table 1 presents the SPARQL representation of a query for pathway objects representing the EGF signaling pathway, and interrelated gene and organism objects.

Table 3. SPARQL algebra representation of simple query indicating source of data

```
LeftJoin(
  Join(
    Union(
      {?gene cabio:geneSymbol ?symbol},
      {?gene cafe:geneSymbol ?symbol})
    Union(
      {?gene cabio:geneName ?gene_name},
      {?gene cafe:geneName ?gene_name})),
  {?gene cabio:geneClusterId ?clusterId},
  true)
```

Table 4. SPARQL expression with zero results

```
Join(
  {?gene cabio:geneSymbol ?symbol},
  {?gene cafe:geneName ?gene_name}))
```

This merging of individuals is indicated in the semCDI query formulation using *definite Horn rules*, which define *a priori* conditional statements that are not explicitly asserted by the ontology extracted from caBIG. By design, they are defined outside of the query, to allow the use of the same rules with multiple queries independently. Two Horn rules are shown in Table 2: the first one uses the OWL inverse-functional property type to indicate that two organisms are the same if their scientific names are equal; the second rule indicates that two genes are the same if their organisms and gene symbols are equal. In these rules we use the presentation syntax derived by the Rules Interchange Format (RIF) Working Group from W3C [1].

3 Query Processing

In order to process a query formulated against the ontology view of caBIG, it is necessary first to represent the query in terms of the data sources to be queried. This is achieved by using OWL subclasses and subproperties distinguished by namespaces. Table 3 shows a partial representation of the query in Table 1, denoted using the query algebra defined in the standard [6]. Note the use of the SPARQL Union operator where data is obtained from multiple sources. Note also that the cluster ID is only obtained from caBIO, as caFE does not contain this information.

The graph pattern expression of this rewritten query then needs to be divided into sub-expressions that obtain data from each specific data source. This is achieved by utilizing our semQA query algebra extension for SPARQL, which defines properties and equivalences between graph patterns. semQA is detailed elsewhere [11]; in summary, it substitutes Union by an idempotent-disjunction Or operator that can be distributed over both Join and LeftJoin. In particular for semCDI, algebraic manipulation is simplified by the fact that the Join of two SPARQL triple patterns from different data sources, as in Table 4, will result in zero solutions. With this, then, the query in Table 1 is transformed into the algebraic expression in Table 5.

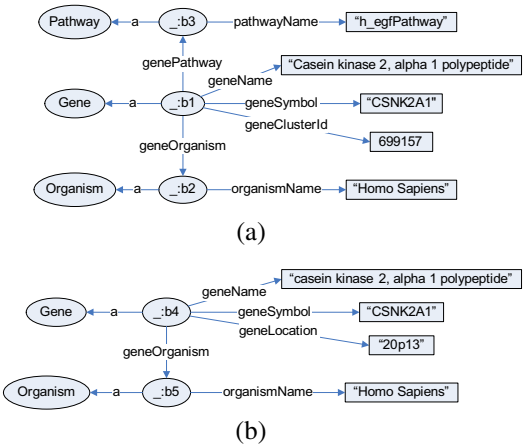

Fig. 1. RDF graph of query result from (a) caBIO and (b) caFE

Table 5. SPARQL algebra representation of query in Table 1 with sub-expressions by data source

```
or(
  Join(
    {?gene rdf:type cabio:Gene},
    {?gene cabio:geneSymbol ?symbol},
    {?gene cabio:geneName ?gene_name},
    {?gene cabio:geneClusterId ?clusterId},
    {?gene cabio:geneOrganism ?org},
    {?org cabio:organismName ?org_name},
    {?gene cabio:genePathway ?path},
    Filter((?pathn = "h_egfPathway"),
      {?path cabio:pathwayName ?pathn})),
  Join(
    {?gene rdf:type cabfe:Gene},
    {?gene cafe:geneSymbol ?symbol},
    {?gene cafe:geneName ?gene_name},
    {?gene cafe:geneLocation ?chr_map},
    {?gene cabio:geneOrganism ?org},
    {?org cabio:organismName ?org_name}}
)
```

Each sub-expression of a query so transformed is then converted into an equivalent query using CQL, the query language for caBIG; this conversion is straight-forward, and involves obtaining the domain model descriptor for each data-source specific property and class. The results from the set of CQL queries are then combined into RDF solution graphs for the original query, as described in the following section.

4 Result Merging

Each CQL query results in a set of individuals that can be represented as an RDF graph. For example, the query in Table 5 results in individuals such as the ones illustrated in

Figure 1(a) for caBIO and (b) for caFE. Other similar results can be obtained from other sources that model the concept "Gene". Note that RDF blank nodes are used to denote the individuals returned.

Table 6. Summary of the Horn rule merge operation

```
INPUT RDF graphs g,h, query q, Horn rules r[]
m = RDF-merge(g, h)
FOR EACH variable v in query q
  FOR EACH individual x in m bound to v
    FOR EACH individual y in m bound to v
      IF x = y by some Horn rule r[i]
        z = new label not in m
        FOR EACH triple in m referring to x
          replace x by z
        FOR EACH triple in m referring to y
          replace y by z
```

Such solutions must be combined according to the Horn rules associated with the query. This is achieved through the Horn-rule merge algorithm illustrated in Table 6. This algorithm takes as input two RDF graphs, *g* and *h*, a query *q*, and a set of Horn rules *r*; the two graphs are two solutions to the original query coming from different data sources. The algorithm begins with an RDF merge [7] of *g* and *h*. Next, it verifies whether two individuals within the resulting merged graph are equivalent by testing the Horn rules; if so, it merges these individuals by creating a new blank node.

The application of this algorithm to the results in Figure 1 is illustrated in Figure 2. In part (a), rule 1 from Table 2 is applied, substituting _:b2 and _:b5 as _:m1. The second part applies rule 2 to further combine _:b1 and _:b4 into _:m2. Note that in this latter combination, the ?gene_name variable could not be combined, as the lexical match between the terms is not exact due to a difference in letter case; thus, when projected into a table of variable bindings, each gene object will result in two rows, one for each name.

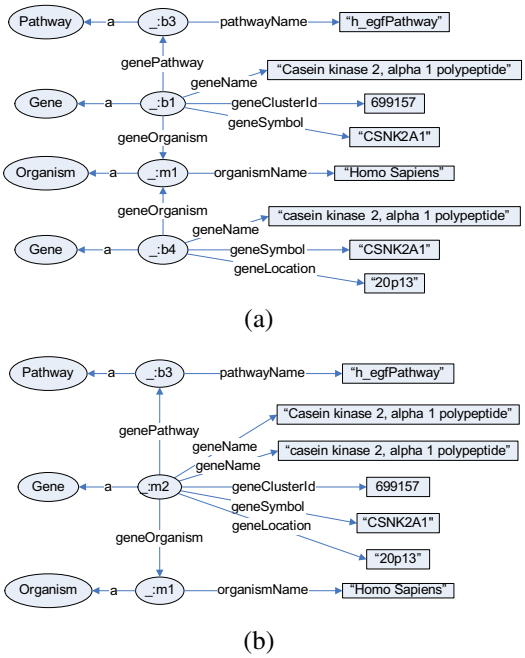

Fig. 2. RDF graph after the application of (a) rule 1 and (b) rule 2

5 Experimental Results and Discussion

Java prototypes of the major model components have been created and interfaced with the caGrid services. The query in Table 1 was run against caBIO and caFE; other sources could not be used due to unavailability through caBIG. Table 7 shows a

Table 7. Partial result for the query in Table 1

symbol	?gene_name	?org_name	?chr_map	?cluster_id
FOS	V-fos FBJ murine osteosarcoma viral oncogene homolog	Homo sapiens	14q24.3	25647
FOS	v-fos FBJ murine osteosarcoma viral oncogene homolog	Homo sapiens	14q24.3	25647
CSNK2A1	Casein kinase 2, alpha 1 polypeptide	Homo sapiens	20p13	701971
CSNK2A1	casein kinase 2, alpha 1 polypeptide	Homo sapiens	20p13	701971
EGF	Epidermal growth factor (beta-urogastrone)	Homo sapiens	2q42-q43	419815
EGF	epidermal growth factor	Homo sapiens	2q42-q43	419815

listing of the first six of the 44 results obtained. We have shown a more extensive set of queries and results obtained in [10].

Several grid-specific issues were encountered. First, many of the caBIG data sources had service outages ranging from a few hours to a few days. Second, as many of these queries involve large data sets, the time it takes to formulate an ontology or receive query results ranged from two minutes to twenty minutes. Third, several of the caDSR domain models contain internal inconsistencies, for example, the field 'name' in the caFE domain model is associated with both 'organismName' and 'organismScientificName' in caDSR, causing the ambiguity to be carried over into the solution graph. As the caBIG project continues to mature and more scientists use the grid for their research, it is likely that many of these issues will be resolved by the caBIG community.

There has been recent work within the caBIG community to develop an infrastructure of data identifiers that would be used to uniquely identifying concepts on the grid. This includes work for the caGrid Identifier Services Framework, which will be capable of supplying globally unique names for caBIG domain objects. Once these additions have been implemented, our models can be extended to leverage these identifiers as a standard Horn rule applied to the result sets.

We are exploring the use of alternative caGrid query interfaces to reduce the overhead of object serialization. Also, the query model will be evaluated against more complex query patterns, and will be expanded to incorporate additional semantics available through NCI Thesaurus. Finally, a visual query interface will be designed to assist cancer researchers with query formulation.

6 Related Work

As a part of caBIG, the cancer Translational Research Informatics Platform (caTRIP) project achieves integration of data from a predefined set of grid services using an object-oriented design. caTRIP performs queries through a federated query engine by using Distributed CQL (DCQL), which is an extension to the CQL query language used by caGrid. In order to merge data from multiple sources for a single query, users must link the attributes for data elements together within the query designing portion of the user interface; the model presented here, on the other hand, uses the relationships

established through Common Data Elements to define the conditions for merging data based on the semantics of caBIG.

The use of SPARQL to execute queries over distributed data nodes in a grid is proposed in [4]; unlike semCDI, this is based upon the use of relational algebra rather than algebras on SPARQL. Other mechanisms for distributing queries expressed in SPARQL and other RDF query languages include [7], [12]; these mechanisms are concerned with queries on RDF data stores. None of the systems or proposals explored in the literature utilize both SPARQL and Horn rules to enable the merging of results obtained from querying.

7 Conclusion

We have presented a model that enables users to query an integrated view of caBIG data services at a conceptual semantic level, allowing researchers to utilize a single conceptual representation of the data instead of the various and distinct domain models defined by each of the underlying data services. We have showed the application of the semCDI query formulation to generate an ontology view of caBIG semantics and define queries in SPARQL complemented with Horn rules; the use of the semQA algebra extension for SPARQL to manipulate queries and obtain sub-expressions for each data service; and the mechanism for the combination of the results from these sub-expressions according to Horn rules. We have presented examples and experimental results showing the working of the model, and we have discussed the direction of future work.

Acknowledgements

This work is supported by NIH grant 1R43CA132293. The authors also wish to acknowledge the insights given by Dr. Thomas Deisboeck at Massachusetts General Hospital, Drs. Robert Clarke and Stephen Byers at Georgetown University, and Dr. Timothy Kuzel at Northwestern University.

References

[1] Boley, H., Kifer, M.: RIF Basic Logic Dialect. W3C Working Draft 30 (October 2007), `http://www.w3.org/TR/rif-bld/`

[2] Collins, F.S., Green, E.D., Guttmacher, A.E., Guyer, M.S.: US National Human Genome Research Institute. A vision for the future of genomics research. Nature 422, 835–847 (2003)

[3] Hartel, F., Coronado, S., Dionne, R., Fragoso, G., Golbeck, J.: Modeling a Description Logic Vocabulary for Cancer Research. Journal of Biomedical Informatics 38, 114–129 (2005)

[4] Hayes, P. (ed.): RDF Semantics. W3C Recommendation 10 (February 2004), `http://www.w3.org/TR/rdf-mt/`

[5] Langegger, A., Blochl, M., Wob, W.: Sharing Data on the Grid using Ontologies and distributed SPARQL Queries. In: 18th International Workshop on Database and Expert Systems Applications, pp. 450–454. IEEE, Los Alamitos (2007)
[6] Prud'hommeaux, E., Seaborne, A.: SPARQL Query Language for RDF, http://www.w3.org/TR/rdf-sparql-query/
[7] Quilitz, B.: DARQ – Federated Queries with SPARQL (1/2008), http://darq.sourceforge.net/
[8] Rodriguez, M.A., Egenhofer, M.J.: Determining Semantic Similarity among Entity Classes from Different Ontologies. IEEE Trans. on Knowledge and Data Eng. 15, 442–456 (2003)
[9] Saltz, J., et al.: caGrid: design and implementation of the core architecture of the cancer biomedical informatics grid. Bioinformatics 22, 1910–1916 (2006)
[10] Shironoshita, E.P., Jean-Mary, Y.R., Kabuka, M.R.: semCDI: A Query Formulation for Semantic Data Integration in caBIG. JAMIA (accepted for publication)
[11] Shironoshita, E.P., Kabuka, M.R.: semQA: SPARQL Algebra and Complexity Analysis. IEE Transactions on Knowledge and Data Engineering
[12] Stuckenschmidt, H., Vdovjak, R., Broekstra, J.: Index Structures and Algorithms for Querying Distributed RDF Repositories. In: Proc. Of WWW Conf. 2004, pp. 17–22 (May 2004)
[13] Wang, X., Gorlitsky, R., Almeida, J.S.: From XML to RDF: how semantic web technologies will change the design of 'omic' standards. Nat. Biotechnol. 9, 1099–1103 (2005)