Building Semantic Information Search Platform with Extended Sesame Framework

Tao Chen*, Yongjuan Zhang*, Shen Zhang, Chengcai Chen, Heng Chen
Shanghai Institutes for Biological Sciences/ Shanghai Information Center for Life Sciences, CAS
Shanghai, China 200031

ABSTRACT
Semantic Web provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries. Therefore, this technology can be used to integrate disparate libraries resources in the field of library located in lots of different web sites. Based on Sesame framework, this article aims at building an extended platform which not only be used to convert web pages to RDF, but also provide a common interface to query semantic data among multiple data repositories. System architecture of the extended application is presented firstly, and how to define data and clue extraction rules for collected data from web pages with tools are introduced subsequently. Convert web data to the uniform RDF triple format data is another key point discussed in this paper. How to merge multi-core Solr and SIREn with Sesame system is also an important problem to be solved. Finally, a simple case is also given to prove the solution proposed in this paper. How to publish the linked data of this paper on the web is the advanced task which will be performed in the near future.

Keywords
Semantic Web, RDF, Ontology, Sesame, Solr, SIREn.

1. Introduction
It is apparent that all that library resources are scattered in different web sites which can cause some disadvantages in digital library. On one hand, the most obvious disadvantage is that these resources cannot be shared in one access entrance, due to a lack of widely-accepted standards for describing those sites and contents. On the other hand, there are many kinds of relationships between those resources. For example, one resource possesses much information described in several web sites, or one entity has different definitions. The problem is made more acutely by the increasing requirement to integrate information from multiple sites. Therefore, how to integrate those information is a key point to research in recent years.

Semantic Web provides a common framework that allows data to be shared and reused across application, enterprise, and community boundaries[1-2]. Resource Description Framework (RDF) is a directed, labeled graph data format for representing information on the semantic web. It is often used to represent, among other things, personal information, social networks, metadata about digital artifacts, as well as to provide a means of integration over disparate sources of information[3-5]. Recently, Semantic Web has been successful used in several domains and applications, such as Uniprot[6] and DBpedia[7]. Therefore, how to convert library resources to structured RDF data and publish them on web is another hot discussed issue in digital library.
Furthermore, some potential relationships and knowledges cannot be found only with traditional document web pages which is not a web of data. On the Semantic Web, data is modeled as a set of relationships between resources, and with inference, new relationships based on the data and some additional information will be generated automatically with vocabularies and a set of rules.
Based on these advantages of Semantic Web, paper aims at building an extended platform for web pages conversion and semantic data query. First, the system architecture is introduced in section two which is extended with new *Indexer* and *Rdfizer* packages. In section three, how to design domain ontology and extracting Biological Science Information Express (BSIExpress, not-for-profit website, aims at collecting and reporting research trend and policy orientation in domestic and international life sciences)[8] and use GooSeeker[9] tools are made clear. The details of how to convert extracted information and index them with integrated Solr and SIREn will be described in section four. A case of application with the extended system is provided in section five. Finally, summaries and the future research directions are described in the conclusion section.

2. System Architecture

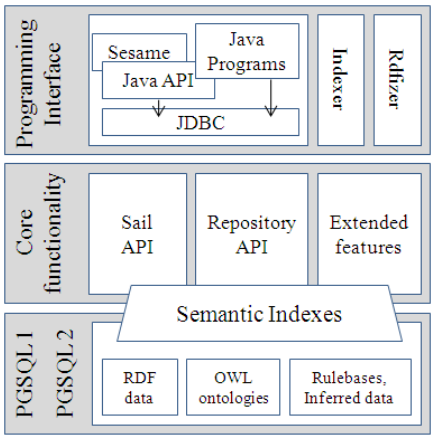

Figure 1. Extended System Architecture

Sesame[10-11] is a popular standard framework for RDF data processing, which includes parsing, storing, inferencing and

---

* Tao Chen and Yongjuan Zhang contribute equally.
§ Corresponding authors：E-mail: yesonme@gmail.com(Tao Chen), zhangyj@sibs.ac.cn(Yongjuan Zhang), hengchen@sibs.ac.cn(Heng Chen).

I-SEMANTICS 2012, 8th Int. Conf. on Semantic Systems, Sept. 5-7, 2012, Graz, Austria

querying of/over such data. It offers an easy-to-use API that can be connected to all leading RDF storage solutions. Sesame supports RAM, disk, or RDBMS storage. Additionally, the central APIs of Sesame are storage-independent and are supported by many third-party RDF database vendors.

The PostgreSQL is chosen as the persistent storage container storing RDF data, OWL ontologies and inference rules, which is shown in figure 1. It is also revealed in this figure that Sesame has two main communication interfaces: the Sail API and the Repository API. The Storage and Inference Layer (Sail) API is a low level system API for RDF stores and inferences. The Repository API is a higher level API and is meant to be the main API against which people can program. It offers various methods for uploading data files, querying, and extracting and manipulating data.

The paper aims at integrating two different resources, biological science information express and bibliographies, which are stored in separated sites without semantic format. In the Sesame framework, some improvements are made to support cross data querying. The main work includes as follows:

a. Multi-database is used to store RDF data according to their sources in order to maintain data conveniently.
b. Sesame features of RDF operations are extended, such as add, modify, remove, delete, etc.
c. Data conversion package is provided-*Rdfizer*, which can convert unstructured data to RDF format that can be reused by other applications.
d. Solr and SIREn are merged into the Sesame framework to index and query RDF data.

3. Data Extraction

3.1 Ontology Design

Heterogeneous data formats provided from different web resources make universal interoperability becoming more difficult but it is really not impossible. The goal of data integration is to gather data from different sources, combine and present them in such a way which makes them being a unified whole. Going beyond a data model, the Semantic Web approach relies on using a standard ontology to integrate different databases. An ontology is a shared specification of the conceptualization of knowledge in a particular domain, which can also be reused by different kinds of applications. It consists of a collection of classes, properties and optionally instances[12-13].

Table 1. Properties Definition of Application

Item	*Property*	*Item*	*Property*
Title	dc:title	Issued	dcterms:issued
Author	dc:creator	Month	swrc:month
Date	dc:date	No.	swrc:number
Publisher	dc:publisher	Vol.	swrc:volume
Category	bie:hasCategory		
Sub-Category	bie:hasSubCategory		

In the application, articles in BSIExpress should be converted with several attributes to RDF. These attributions and the data properties of ontology are listed in table 1. The left column is the properties of article information, and journal details are listed in right one. All the most of attribute terms have been stated in common ontologies, such as these namespaces: dc: <http://purl.org/dc/elements/1.1/>, dcterms: <http://purl.org/dc/terms/>, and swrc: <http://swrc.ontoware.org/ontology#>. However, category and subcategory properties should be extended for meeting project requirements. Category, an inseparable part and main classification of express article, contains fundament research, biotechnology, serious infectious diseases, and others in application. Fundamental search category has Stem Cell, Clone, Genomics and Metanomics, etc. In design, rdfs:subClassOf property is used to describe the relationships between category and subcategory.

Properties, prefixed with 'bie' which is a namespace in internal environment, are designed for BSIExpress domain. It is just namespace that can resolve ambiguity allowing two terms in different ontologies to share a local name.

3.2 Data Extraction

The site source contains a large number of diverse detail data, required of ontology. These unstructured data cannot be reused by semantic web directly; therefore, how to extract and structure them is the key point for semantic data preparation.GooSeeker, focusing on data schema modeling and data extraction, provides some useful tools to extract data, such as MetaStudio and DataScraper. MetaStudio is a tool describing data schemas of target web pages, it is appropriate for defining data schema and generating data extraction rules for express and bibliography, and provides many validation facilities to find if the defined data schema and extraction rules can work as that we expect. DataScraper, another used tool, is applied to continuously extracting data from the web, which is instructed by the data extraction rules generated by MetaStudio. The resulted XML-formatted files are stored onto DataStore server.How to define Data and Clue Extraction Rules (DCER) are the important two steps in schema design. The rules are made up of a series of XSLT, XPath and proprietary XML commands. Data extraction rule appoints where and how to extract data snippets from a target page, whereas clue extraction rule specifies where and how to extract new clues from a target page.

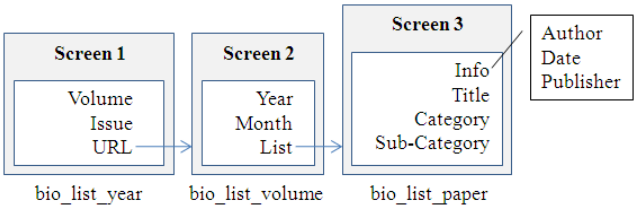

Figure 2. Multi-level Data Extraction

Figure 2 indicates the multi-level data extraction rules for express. All the information is not included in a single screen: in the first screen, volume and issue data can be extracted, as click a special volume URL, screen will display the volume detail information, in which list all articles of this volume. Then, only the article page is accessed, its title, author, category and others data can be acquired. Pattern clue is an important clue extraction method in different ongoing screens. This clue is a series of clues which can be extracted within a scope of target HTML document according to patterns of URL's character strings. In the examples, URL is the address identifier for express articles.

After the extraction rules are defined, all the needed data will be extracted by the DataScraper tool. Comparing with the disparate data listed in figure 2, the extracted data are structured in XML format, shown in following segment.

```
<!-- BSIExpress paper information -->
<Detail>
  <Item>
    <Info>update：2012-03-06  source：
Sciencedaily</Info>
    <Title>The American Journal of Pathology：Protein That
Functions in Normal Breast May Also Contribute to Breast
Cancer Metastasis
    </Title>
    <Subcategory>zhuti_1.asp?Y_1=Cancer</Subcategory>
    <Category>
      zhuti.asp?Y=Fateful Chronic Illnesses
    </Category>
  </Item>
</Detail>
```

4. Data Management

4.1 Data Conversion

The BSIExpress domain ontology, having been designed in part II, should be imported to application with Sesame workbench firstly. Then, journals and articles can be imported to system with features extended in Sesame, which class diagram is displayed in figure 3, in which *CommonOntologies* class defines the public namespace and their properties, like dc, swrc, and dcterms. And class *ModelSchema* declares private domain properties. These properties in these two classes will be recalled by *VolumeModel* and *PaperModel*, which can be used to parse volume and article information to RDF with its *processContent()* method. This method returns *RepositoryResult<Statement>* result including all the n-triples of the converted data.

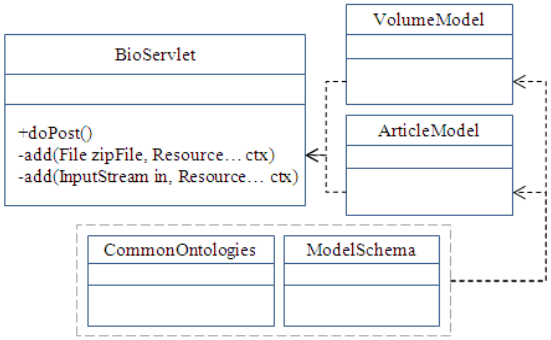

Figure 3. Data Conversion Diagram

BioServlet is the concrete servlet file to interpret input stream uploaded from web interface. This servelet has two “add” methods with different parameters, the one with *File zipFile* parameter can parse zip package and zip more xml files, the other includes *InputStream* in parameter, which can parse one xml file every time. Parameter *Resource ctx* means whether binding context value. Using context, you can think of as a way to group sets of statements together through a single group identifier which can be a blank node or a URI. On the contrary, if this parameter is set to blank, the sets will be not grouped.

The disparate data can be mapped to several triples as converted, in which each triple expresses a simple proposition. For index and search, blank node is avoided to be used as object and subject as much as possible, since blank nodes simply indicate the existence of a thing, without using, or saying anything about, the name of that thing. Moreover, they will complicate the lives of data consumers, especially does so if the data changes.

In conversion, all the statements of one subject should be stored in *Map<String,ArrayList<String>>*, in which the first parameter stands for subject name, and the *ArrayList<String>* is the array of statements. Index method for handling triple data should be added in every logical transaction, such as inserting, modifying, clearing and others.

4.2 Data Index

Efficient and large handling of semi-structured data (including RDF) is a gradual important issue to many web and enterprise information reuse scenarios. SIREn[14-15] (Semantic Information Retrieval Engine) is a Lucene plugin to efficiently index and query RDF, as well as any textual document with an arbitrary amount of metadata fields. Solr is “the popular and blazing fast open source enterprise search platform” which can provides distributed search and index replication, and powers the search and navigation features[16].

Multiple PostgreSQL databases are used to store data of different web sites, and multi-core Solr is taken for maintaining them here. Each core is a completely independent search index; therefore the database data can be re-indexed solely without impacting other data.

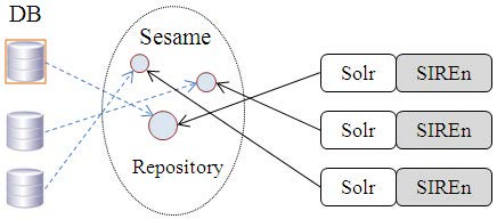

Figure 4. Sesame, Solr and SIREn Integration

Merging Solr and SIREn into Sesame framework is another important improvement in project. From figure 4, this improvement can be understood precisely and intuitively. With Sesame repositories, semantic data can be accessed and maintained conveniently. Solr core will be mapped in every repository connecting to real RDMS. In Solr configuration file, SIREn must be registered to query parser so as to explain RDF statements. All the statements of one document are bound to a field-ntriple-in SIREn. In order to accomplish this architecture, several servlet actions of Sesame must be rewritten in order to support index with Solr. Besides this field, there is another filed named “url” in Solr configuration, which records the subject values of statements.

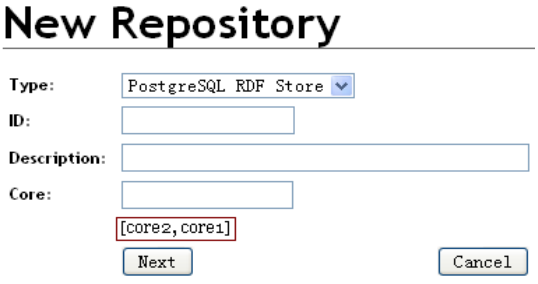

Figure 5. Add Solr core in creating new repository

Another improvement is merging IK analyzer, an open-source package for Chinese word segmentation, to support Chinese character in free text query. In developing, because only object will contains Chinese words, so the IK tokenizer only be added in string data type of ntriple field.

As adding new repository in Sesame, the Solr core identifier is also added in it, shown in figure 5, which will store indices of every repository node in different index place. The identifier will be removed if it is used, but it can also be reused as the repository removed from Sesame. After added, the used core will list in repository table, for example core0 is attached to 'sibs' repository, shown in figure 6. It's don't recommend that bind core and change anything in default repository, SYSTEM.

Figure 6. Solr core in sesame framework

The extended system has a window to maintain indices, which also provide more power ways to rebuild indices according to different granularities.

a. Article is the minimal document unit to index, so we can rebuild the special article through the object value of RDF triple.
b. Context is the option setting for Sesame, therefore, a group of sets with same context can be rebuilt.
c. The same predicate of n-triples can also be rebuilt too.
d. All the data in one repository can be rebuilt wholly if the index condition blank is set.

5. Data Search

The BSIExpress articles in the last four years have been stored in different repositories and databases for testing. In Sesame workbench, these data can be managed and maintained solely, but the newly extended system provides a same access entrance to query them cross these repositories. All repositories are listed in search field with a checkbox setting, shown in figure 7. In this way, people can query all data if he selects all repositories, and also can select one interesting repository to search, which will reduce responding time in querying. Also it is easy to reduce repository node if one repository is in maintaining or not be used again. Furthermore, system can merge other data repositories to become a dynamic data integrating hub.

Figure 7. Simply Search for Multi-repositories

RDF metadata of every article are counted in result lists, which can also be exported alone for reuse by other semantic applications with multiple formats, N3, RDF/XML, turtle.

6. Conclusion

Data integration via shared semantic standards is critical to the digital library. In general, this extended system possesses some advantages compared with traditional data integrating model as follows:

a. Scalability. Obviously, it is convenient to extend more repositories and databases for querying with this framework.
b. Dynamically. Data with same subject can be automatically integrated even scattered in multi-sources after converted to RDF standards using one of semantic technology.
c. Flexibility. Through integrating multi-core Solr and SIREn, one repository's RDF triples can be indexed and queried solely.

Certainly, there are still many aspects needed for improvement in the future, for example, the semantic data in this paper will be published to web as linked data which describes a method of publishing structured data; building federated query to connect another published SPARQL endpoints, such as UniProt, PubMed, etc.

7. Acknowledgment

Thanks to funding from Shanghai Planning Office of Philosophy and Social Science, an administration institution for Shanghai social science research created in 1992.
We also shall extend our thanks to the staff of this project for all their hard work and help in system research and application realization of main modules. Last, I'd like to thank all my friends for their encouragement and support.

8. References

[1] Alexandre Passant, "Semantic Web Technologies for Enterprise 2.0," *ISO Press*. 2011.
[2] Sven Groppe, "Data Management and Query Processing in Semantic Web Databases," *Springer-Verlag*, 2011.
[3] Giovanni Sartor, Monica Palmirani, Enrico Francesconi, etc., "Legislative XML for the Semantic Web: Principles, Models, Standards for Document Management," *Springer*, 2011
[4] Rajendra Akerkar, "Foundations of the Semantic Web," *Oxford: Alpha Science*, 2009.
[5] "RDF – Semantic Web Standards, " http://www.w3.org/RDF.
[6] http://www.dbpedia.org.
[7] http://www.uniprot.org.
[8] http://www.bioexpress.ac.cn.
[9] http://www.gooseeker.com/.
[10] J. Broekstra, A. Kampman, and F. Van Harmelen., "Sesame: an Architecture for Storing and Querying RDF and RDF Schema," *Proc. 1st Int'1 Semantic Web Conf.* (ISWC 2002), LNCS 2342, Springer, 2002, pp.54-68.
[11] Mark Watson, "Practical Semantic Web and Linked Data Applications," 2011.
[12] Aldo Gangemi, "Ontology Design Patterns for Semantic Web Content," Proc. *4th Int'4 Semantic Web Conf.* (ISWC 2005), LNCS 3729, Springer, 2005, pp. 262-276.
[13] David Taniar, Johanna Wenny Rahaju, "Web Semantics Ontology," Idea Group, 2006.
[14] Renaud Delbru. *Searching MWeb Data: an Entity Retrieval model*. Doctoral Thesis at Digital Enterprise Research Institute, National University of Ireland, Galway. September 2010.
[15] R. Delbru, N. Toupikov, M. Catasta, etc., "A Node Indexing Scheme for Web Entity Retrieval," *In Proceedings of the 7th Extended Semantic Web Conference* (ESWC). 2010.
[16] "Apache Solr," http://lucene.apache.org/solr/.