Metadata Based Management and Sharing of Distributed Biomedical Data

Fusheng Wang

Department of Biomedical Informatics,
Emory University,
Atlanta, GA, USA
Fax: +1 E-mail: fusheng.wang@emory.edu

Fusheng Wang

Department of Biomedical Informatics,
Emory University,
Atlanta, GA, USA
Fax: +1 E-mail: fusheng.wang@emory.edu

Peiya Liu

Department of Integrated Data Systems, Siemens Corporate Research
755 College Road East, Princeton 08540, USA

Abstract: Biomedical research becomes reliant on multi-disciplinary, multi-institutional collaboration, and data sharing is becoming increasingly important for researchers to reuse experiments, pool expertise and validate approaches. However, there are many hurdles for data sharing, including the unwillingness to share, lack of flexible data model for providing context information for shared data, difficulty to share syntactically and semantically consistent data across distributed institutions, and expensive cost to provide tools to share the data. In our work, we develop a Web-based collaborative biomedical data sharing platform *SciPort* to support biomedical data sharing across distributed organizations. SciPort provides a generic metadata model for researchers to flexibly customize and organize the data. To enable convenient data sharing, SciPort provides a central server based data sharing architecture, where data can be shared by one click through publishing metadata to the central server. To enable consistent data sharing, SciPort provides collaborative distributed schema management across distributed sites. To enable semantic consistency for data sharing, SciPort provides semantic tagging through controlled vocabularies. SciPort is lightweight and can be easily deployed for building data sharing communities for biomedical research.

Keywords: Metadata; Scientific Data Management; Data Sharing; Data Integration; Computer Supported Collaborative Work.

Reference to this paper should be made as follows: Rodríguez Bolívar, M.P. and Senés García, B. (xxxx) 'The corporate environmental disclosures on the internet: the case of IBEX 35 Spanish companies', *International Journal of Metadata, Semantics and Ontologies*, Vol. x, No. x, pp.xxx–xxx.

Biographical notes: Manuel Pedro Rodríguez Bolívar received his PhD in Accounting at the University of Granada. He is a Lecturer at the Department of Accounting and Finance, University of Granada. His research interests include issues related to conceptual frameworks of accounting, diffusion of financial information on Internet, Balanced Scorecard applications and environmental accounting. He is author of a great deal of research studies published at national and international journals, conference proceedings as well as book chapters, one of which has been edited by Kluwer Academic Publishers.

Belén Senés García received her PhD in Accounting at the University of Granada. She is a Lecturer at the Department of Accounting and Finance, University of Granada. Her research interests are related to cultural, institutional and historic accounting and in environmental accounting. She has published research papers at national and international journals, conference proceedings as well as chapters of books.

Both authors have published a book about environmental accounting edited by the Institute of Accounting and Auditing, Ministry of Economic Affairs, in Spain in October 2003.

1 Introduction

With increased complexity of scientific problems, biomedical research is increasingly a collaborative effort across multiple institutions and disciplines. Data sharing is becoming critical for validating approaches and ensuring that future research can build on previous efforts and discoveries. As a result, data sharing is often required by scientific funding agencies to share the data produced in grant projects. For example, the National Institutes of Health (NIH) of US requires data sharing for NIH funded projects of $500,000 or more in direct costs in any one year.

To support large scale collaborative biomedical research, NIH provides large-scale collaborative project awards for a team of independently funded investigators to synergize and integrate their efforts, and the awards mandate the research results and data to be shared NIH Statement on Sharing Scientific Research Data (http://grants.nih.gov/grants/policy/data_sharing/); Policy for sharing of data obtained in NIH supported or conducted genome-wide association studies (2008). The Network for Translational Research (NTR): Optical Imaging in Multimodality PlatformsNetwork for Translational Research (NTR): Optical Imaging in Multimodality Platforms (http://imaging.cancer.gov/programsandresources/specialized initiatives/ntroi) is one of such collaborative projects on the development, optimization, and validation of imaging methods and protocols for rapid translation to clinical environments. It requires not only managing the complex scientific research results, but also sharing the data across hundreds of research collaborators. As another example, Siemens Healthcare has research collaborations with hundreds of research sites distributed across the US, each providing Siemens marketing support by periodically delivering white papers, case reports, clinic methods, clinic protocols, state-of-the-art images, etc. In the past, there were no convenient methods for research partners to share data with Siemens, and mostly data were delivered through media such as emails, CDs and hard copies. This made it very difficult to organize, query and integrate the shared data.

Sharing biomedical research data is important but difficult Piwowar et al. (2008); Birnholtz et al. (2003). One major difficulty is the unwillingness to share, as investigators may restrict access to data to maximize the professional development and economic benefit Bekelman et al. (2003). With increased awareness of this issue, the government and funding agencies are developing strategies and policies to promote and enforce the sharing of data Piwowar et al. (2008); Data Sharing & Intellectual Capital (DSIC) Workspace (https://cabig.nci.nih.gov/working_groups/DSIC_SLWG); Getting Connected with caBIG (https://cabig.nci.nih.gov/getting_connected/). While the social issue of data sharing is challenging, this paper will mainly focus on the development of a biomedical data sharing system which makes data sharing becoming flexible and easy for humans on collaborative research.

1.1 Complexity of Biomedical Data

Biomedical data can be in heterogeneous formats such as structured data, standard based medical images such as DICOM Digital Imaging and Communications in Medicine (DICOM) (http://medical.nema.org/), non standard medical images such as optical images, raw equipment data, spreadsheets, PDF files, XML documents, and many others. Thus, the data are a mix of structured data (often deeply hierarchical) and files. Meanwhile, the data structures of biomedical research data can be very complex, ranging from different primitive data types, to lists and tables, from flat structures to deep nested structures, and so on. It is often difficult for investigators to create a sophisticated schema that can capture enough context information or metadata information, if the data types do not have a public, centralized, and well-recognized database. During our interviews with more than a dozen biomedical research investigators, we found that major investigators use spreadsheet like tools and manage their data through operating system folder organized files. To reduce the cost and complexity for data sharing, a data sharing system needs to provide an extensible architecture that can be easily customized by researchers with their own metadata models, even without programming skills.

1.2 Management of Data Sharing

Even with willingness to share data, in many cases an investigator would like to control when to share data, and who can access the data. An investigator may want to keep unpublished data only to his or her own group or collaborative partners, and selectively publish a subset of data for sharing. An investigator may also want to stop sharing certain data at any time, for example, when he or she finds a quality problem of the data. An investigator may also have new results added on existing shared data and want to keep the data current. A manageable data sharing system with convenient operations can potentially increase the willingness for users to share, as the effort for data sharing can be much reduced.

1.3 Data Sharing Architecture

There are three major types of data sharing architectures: i) Centralized, multiple datasets hosted at a single location in a common schema. For example, the Cancer Genome Atlas (TCGA) Data Portal The Cancer Genome Atlas (TCGA) Data Portal (http://cancergenome.nih.gov/dataportal) manages genome related data; National Biomedical Imaging Archive National Biomedical Imaging Archive (https://cabig.nci.nih.gov/tools/NCIA) manages DICOM based medical research data.

ii) Federated, with a virtual view of physically separate datasets. For example, Cancer Biomedical Informatics Grid (caBIG®) caBIG: cancer Biomedical Informatics Grid (http://caBIG.nci.nih.gov/) is a Grid based data federation infrastructure that supports a CQL query language across distributed data sources. iii) Distributed, physically and virtually separate datasets. Centralized approach is often limited to common data types. Biomedical research, however, generates complex data and often new data types. Distributed approach is often difficult to retrieve, interpret and aggregate results, and lacks data consistency between research sites. While caGrid is becoming widely used in biomedical research community, caGrid itself has a complex infrastructure and the effort is significant.

Considering the high dynamic nature of biomedical research and the need of cost effective data sharing, we develop a hybrid architecture which combines the benefits of the centralized approach and the distributed approach. In this approach, a data sharing central server is provided for multiple distributed data sources, and stores only published metadata (not raw data or images with large sizes) from distributed data sources. Users can have flexible management of data sharing through publishing or unpublishing data with a simple operation. The metadata contain context information of original data sources (including raw data) which are still managed at distributed research sites. The central server provides an integrated view of all shared data, and shared data can be easily aggregated and retrieved from the central server. This architecture not only is lightweight, but also provides support of consistent data sharing through collaboratively managing schemas and semantically tagging of data.

1.4 Inconsistency of Shared Data

Another difficulty on sharing data is the incompatible representation of data between sharing partners. Each site may use different schemas (e.g., data templates) and represent data in different formats; and even if multiple partners choose to use the same schema, the schema may evolve as research progresses. For example, a field name may be changed or additional constraints may be added in a schema. Data generated from different versions of a schema may not match when searched together. For example, if a field named “age” is changed to “patient_age”, these two field names will lead to structurally different fields and can not be searched together as a single field. Semantic consistency is on using the same vocabulary to represent knowledge. For example, one investigator may use “Malignant Tumor” to describe a lesion, and another investigator may use ‘cancer’. While semantics annotations can be added to data to enhance semantic interoperability between shared data, there is a lack of integrated tools and architecture that can support semantic annotations from controlled vocabularies.

1.5 Our Contributions

To meet the requirements of a data sharing system and overcome the challenges, we develop SciPort, a Web-based collaborative biomedical data sharing platform to support collaborative biomedical research. SciPort brings together the following essential components to support biomedical data sharing:

- Generic biomedical experiment metadata model and XML based biomedical data management for research sites to flexibly customize and organize their data (Section 2);

- Hybrid, lightweight data sharing architecture through a Central Server, and data sharing can be conveniently managed by investigators (Section 3);
- Easy sharing of schemas between distributed research partners (Section 4);
- Collaborative management of schemas and their evolution between distributed research sites to maintain syntactic data consistency (Section 5);
- Semantic tagging of data to achieve semantically consistent data sharing (Section 6).

This paper has been significantly extended from previous work Wang and Vergara-Niedermayr (2008), which discussed preliminary work and the technique aspect on distributed scientific data management. The work presents here represents latest results and software development, and has a major new focus on the human and sharing aspects of biomedical data. The paper also covers new work on metadata modeling for data sharing, collaborative tagging for data sharing, user experience on real world deployment, and so on.

2 Generic Biomedical Experiment Metadata Modeling and Management

As discussed in Birnholtz et al. (2003), metadata models serve as the abstraction of data, and are essential for understanding shared data. However, metadata are often incomplete, and important context information of data Chin and Lansing (2004) is often ignored. However, many domain specific applications or data models are specific to certain types of data and limited to a small set of context information. To precisely capture as sufficient metadata and context information as possible for biomedical data sharing, researchers themselves should be also to customize their own metadata models easily to meet their research need and data sharing need.

SciPort was first developed as a biomedical data management system with metadata modeling, authoring, management, viewing, searching and exchange.

2.1 Generic Meta Data Modeling for Biomedical Data

To meaningfuly represent each dataset, we develop a *SciPort document* model to represent the metadata and context information of the data. A SciPort Document can represent both (nested) structured data, files and images.

A SciPort document includes several objects: i) Primitive Data Types/Fields. Primitive data types are used to represent structural data, including *integer*, *float*, *date*, *text*, and Web-based data types such as *textarea*, *radiobutton*, *checkbox*, *URL*, etc.; ii) File. Files can be linked to a document through the file object; iii) *Reference.* A reference type links to another SciPort document; iv) Group. A group is similar to a table, which aggregates a collection of fields or nested groups. There can be multiple instances for a group, like rows of a table; v)Category. A category relates a list of fields, e.g., "patient data" category, "experiment data" category, etc. Categories are used only at the top level of the content, and categories are not nested.

Figure 1 shows an example SciPort document that describes image annotations on top of medical images. This document captures generic information such as the tile, description, author, creation date and modification date; patient information such as the age and gender; annotation data such as a number of annotated tumors marked up on top of the image, and the

Figure 1 A Sample SciPort Document

link to the image file from which annotations were generated. Note that in the example, we can easily represent nested complex information such as multiple tumors, multiple spatial coordinates and different data types such as images and files.

2.2 XML Based Implementation and the Benefits

The hierarchical nature of the data model fits perfectly with the tree based XML data model, and we take an XML based approach to implement the data model – we call it *SciPort Exchange Document*. Users can also easily define their own schemas which are internally represented with an XML-based schema definition language. Besides, To provide an intuitive way to present data, we also develop a hierarchical model based on XML to organize biomedical data, thus documents can be quickly browsed and identified through the hierarchy. For example, we can define a hierarchy with levels of "site"→ "patient" → "measurement", and attach documents at different folder levels. SciPort also provides fine grained access control at folder level for the data in the database Wang et al. (2009).

The self-describing and rich structure of XML makes it possible to represent arbitrary complex biomedical research data. By modeling the data as XML documents, we can take advantage of native XML database technologies to manage biomedical data, thus we can avoid complex data model and query translation between XML and RDBMS. This is especially beneficial since users can define arbitrary structured and nested data formats for their data. Moreover, powerful queries can be supported directly on XML databases with the standard XML query language XQuery W3C XML Query (XQuery) (http://www.w3.org/XML/Query).

A salient characteristic of SciPort is that the system is highly adaptable. SciPort provides a Web-based schema authoring tool to easily create complex hierarchical metadata schema model without any need of programming. The organizational hierarchy is also customizable with its hierarchy authoring tool. This allows users to configure their own biomedical data repository without requiring expensive and time-consuming services or software development effort.

3 Sharing Distributed Biomedical Data

While investigators can share their data simply through giving login information to collaborators, a more systematic approach for sharing data across research consortia or networks can provide more flexibility and benefits. SciPort comes with comprehensive data sharing capabilities: i) Convenience: data sharing is performed by a single action and data can be selectively shared; ii) Ownership of data: researchers own and manage their data by their own; iii) Flexible sharing control: data sharing can also be revoked by researchers at any time; iv) Up-to-date of shared data. As data are updated or removed, corresponding shared data also need to be synchronized accordingly to stay current; v) Consistently aggregated shared data from distributed sites; and vi) Lightweight. Sharing is manipulated through metadata, no copy of large volume data is needed.

These sharing capabilities are implemented through a lightweight, central server based approach, as discussed next.

Figure 2 The Central Server Based Architecture for Data Sharing

Figure 3 An Example of Publishing an Existing Document

Figure 4 An Example of Publishing a New Document

3.1 Sharing Data through a Central Server

SciPort provides a distributed architecture to share and integrate data through a Central Server (Figure 2). In this architecture, each research site will have its own Local SciPort Server which itself functions as an independent Server for data collection and management. In addition, there will be an additional Central Server upon which Local Servers are able to selectively publish their data (structured documents) (Figure 3). Images/files, which are often the major source of data volume, are still stored on corresponding Local Servers but are linked from the published documents on the Central Server. Once a user on the Central Server begins to download a document from the Central Server, actual data files are downloaded from the corresponding Local Server that holds the data.

Thus, the Central Server provides a global view of shared data across all distributed sites, and can also be used as a hub for sharing schemas among multiple sites. Since data are shared through the metadata (SciPort Documents), the integration is lightweight. Users on the Central Server will only have read access to the data.

Figure 2 illustrates an example SciPort sharing architecture formed by four Local Servers at four universities: UCI, UCSF, Dartmouth and Penn. Each Local Server is used for data collection and management of clinical trial data at its local institution. Since these clinical trials are under the same research consortium, they would share their data together by publishing their data (documents) to the Central Server located at UCI. Members at NTR research consortia are granted read access on such shared data through the Central Server. Once the user identifies a data set from the Central Server and wants to download the data, the user will be redirected to the corresponding hosting Local Server to download the data to the client.

3.2 Data Synchronization

Shared data may become outdated as new results or analysis are generated on the data. SciPort provides automated data synchronization on shared data through synchronization enforcement in the following operations: i) Create. When a document is created, the author or publisher has the option to publish this document (Figure 4). Once the document is published, a "published" status is added to the document. A user can also set up an automatic publishing flag so all new documents will be automatically published; ii) Update. When an update is performed on a published document, the document will automatically be republished to the Central Server; iii) Delete. When a published document is deleted, it will also be automatically removed from the Central Server; iv) Unpublish. A user can stop sharing a document by selecting the "unpublish" operation. Unpublished documents will be removed from the Central Server.

Figure 5 Sharing Data in Multiple Data Networks

Figure 6 Types of Information Sharing in SciPort

3.3 Security and Trust between Local Servers and the Central Server

Server Verification. The trust between Local Servers and the Central Server is implemented through security tokens. For a Local Server to be accepted into the network, it will be granted a security token to access the Central Server services. The token will be imported at the setup step. When a Local Server tries to connect to the Central Server, the Central Server verifies if the token matches.

Single Sign-on and Security. One issue for sharing data from distributed databases is that it is not feasible for Central Server users to login to every distributed database. When a user publishes a document, the user already grants the read access of the document (including the files linked to the document) to the users on the Central Server, thus another authentication is unnecessary. Therefore, users on the Central Server should be able to automatically access shared data from a Local Server in a transparent way. To support this, the Local Server Document Access Control Manager has to make sure that the remote download requests really come from Central Server users who are currently logging on. We develop a single sign-on method to guarantee the security of the data sharing, by verifying if an incoming data request to a Local Server comes from the Central Server and if the current request user is currently logged in on the Central Server.

3.4 Sharing Data in Multiple Data Networks

Data can be shared not only in a single data network through one Central Server, but also in multiple data networks through multiple Central Servers. One organization may want to share the same data in multiple networks, as demonstrated in an example (Figure 5). There are two networks, one centered at UCI and another centered at Stanford. UCI is collaborating with both networks and needs to share data with both networks. UCI will be granted as a partner site and its Local Server will be configured for both networks. A document can then be published to any of the Central Servers or both. This sharing architecture makes it possible for very flexible data sharing across different research networks.

Besides data sharing, SciPort also provides sharing of schemas between distributed research sites (Figure 6), discussed in Section 4.

3.5 The Benefits of Our Hybrid Data Sharing Architecture

The architecture of SciPort is a hybrid of centralized approach and distributed approach. Centralized approach can provide high visibility, easy retrieval, easy aggregation within the repository, but suffers from limited data types or flexibility of customizing new data types. Distributed approach allows users to flexibly manage, customize and control their data and data sharing, but often suffers from low visibility, difficulty on retrieval, interpretation, and aggregation, and lacks of data consistency. Our hybrid approach benefits from both. The Central Server provides high data visibility, easy retrieval and aggregation of all shared data from distributed sites. SciPort allows users to conveniently customize their data types through creating and updating schemas. Users can flexibly manage their data

Figure 7 An Example of Publishing Schemas from a Local Server

Figure 8 An Example of Importing Schemas to a Local Server

sharing through simple publishing or unpulishing operations. Data consistency is maintained through collaborative schema sharing and management (discussed in next two Sections). The hybrid approach is also lightweight, as it does not store directly original data such as raw data or medical images – which can have high data volumes, but only metadata or structured documents that describe and abstract the data.

4 Sharing Schemas

Schemas are used to define structures and constraints of documents. The former includes a mix of (possibly nested) object types defined in the data model, and the latter includes i) number of instance constraints for file and group types, ii) minimal and maximal value constraints; and iii) controlled values. Schemas are an essential component since they are used for i) data validation; ii) document authoring form generation; iii) data presentation – templates are defined based on schemas; and iv) search form generation.

Sharing schemas are critical for sharing data, since the Central Server glues data together using shared schemas to present and search data. How to keep data from multiple Local Servers coherent is also dependent on at what level and how schemas can be shared consistently.

4.1 Publishing Schemas

Schemas can be shared by publishing them to the Central Server. From a Local Server *Schema Management* menu, schemas created on the Local Server can be selectively published to target Central Servers, as shown in the example in Figure 7. When a new document is being published to the Central Server, the availability of the corresponding schema on the Central Server will be checked. If the schema is not present, then the schema will also be published together with the document.

The *owner of a schema* is defined as the Local Server on which the schema is first created. A schema is identified by its owner and schema ID. SciPort also provides comprehensive access control management Wang et al. (2009), and two roles are related to schema management: i) *organizer* role with privileges to author and update schemas and ii) *publisher* role with privileges to publish documents and schemas.

Once a schema is published, it can be shared through the Central Server. Other Local Servers can reuse schemas by importing schemas from the Central Server, as shown in Figure 8. A schema can be unpublished from a Central Server by a Local Server, thus the schema is not available on the Central Server for further sharing. Users on the Central Server with an "organizer" role can remove a schema from the Central Server if no document on Central Server is depending on this schema. This can be used to clean up non-used schemas.

4.2 Three Scenarios of Schemas Sharing

Based on the use cases, there are three typical scenarios of schema sharing:

Figure 9 Static Schemas Sharing

Figure 10 Uniform Schema Sharing

Figure 11 Multiform Schema Sharing

Static Schema A schema is fixed and will not be changed. For example, some common standard based schemas are not likely to change. Once a schema is authored and changed to the final status, it can be published to the Central Server to be shared by every Local Server (Figure 9). This is the simplest scenario, as schemas can be created once and shared directly through the Central Server.

Uniform Evolving Schema A Schema can be changed and a uniform version is shared by all Local Servers, thus data consistency is maintained across all sites. The schema is owned by its original creating Local Server, and the owner can make certain changes.

Multiform Evolving Schema A "seed" schema is first created and shared as public – every Local Server becomes the owner and can update the schema. The Central Server maintains a version of the schema that conditionally merges updates from Local Servers, thus all documents published on the Central Server will be compatible under this schema.

Next, we will discuss how to manage schemas and their evolution for the last two scenarios (static schema management is straightforward and ignored here.)

5 Collaborative Management of Schemas for Consistent Data Sharing

Since schemas can be shared and used across multiple distributed data sources, one challenging issue is how to manage schema evolution while keeping shared documents consistent and compatible with their schemas in a distributed environment. To solve this problem, we first define the following favorable rules for schema management and sharing in a distributed environment:

- *Minimal administration*. Human based manual schema management can be difficult, especially for schemas that may be updated by different sites. To minimize human effort for managing schemas across distributed sites, it would be ideal if the data sharing system can facilitate the management of schemas, for example, relying on an information exchange hub based on the Central Server.

- *Data consistency*. Schema evolution has to be backward compatible otherwise the integrity of documents will be broken.

- *Control of schemas*. To prevent arbitrary update of schemas, by default, only the owner (the user who creates the initial schema on a Local Server) of a schema can update a schema.

- *Current of Schemas*: The Central Server always has the up-to-date version of a schema if that schema is shared by the owner, to guarantee no outdated schemas are shared across sites.

- *Sharing Maximization*: Since shared schemas on the Central Server can be unpublished by local database users and removed from the Central Server, to promote sharing, only the last time publisher of this schema can unpublish a schema.

These rules will help to automate schema management in a distributed environment, minimize conflicts of updates, maintain coherent shared data on the Central Server, and reduce the effort from humans. To achieve this, we enforce the rules for schema operations on both Local Servers and Central Servers. Schema operations on Local Servers include create, update, delete, import, publish, and unpublish, and schema operations on Central Servers include remove.

5.1 Uniform Schema Management

In this scenario, a Schema can be changed and a uniform version is shared by all Local Servers, thus data consistency is maintained across all sites and the Central Server. Next we discuss how schema management rules are enforced in each schema operation.

Create and Delete

A schema can be created on a Local Server if the user has the organizer role. This Local Server will become the owner of this schema. A schema can be deleted if the user has the organizer role, and there are no documents using this schema on this Local Server. If the Local Server is the owner of the schema, and the schema is never published, deleting a schema will eliminate the schema forever. If the schema was once published, it may still be alive and used on other Local Servers.

Update

Incompatible update of a schema is the one that can lead to inconsistency between the new schema version and existing documents. Incompatible updates are not allowed unless the following conditions are met: i) only the owner of a schema can make updates to the schema; ii) there are no existing documents created based on this schema on this Local Server; and iii) the schema was never published, i.e., there will no other Local Servers using this schema to create documents.

Compatible update of a schema will not lead to inconsistency between the new schema version and existing documents. The following conditions are required for compatible update. i)*Ownership.* The Local Server is the owner of the schema, and the user is the organizer on this Local Server; ii)*Field Containment.* All the fields in the last schema are present in the new schema and belong to the same category or group, and new categories and fields added. iii)*Type Compatibility.* All the fields in the new schema have the same type or a compatible type, i.e., a more general type. iv)*Relaxed Value Range Constraints.* No new value constraints are permitted for existing fields which do not have any constraint. Value constraints can be updated with more relaxed ranges. Constraints on new fields are permitted. v)*Relaxed Controlled Values.* For field with controlled values, the extent is enlarged with more options. vi)*Relaxed Constraints on Number of Instances.* For group field or file field, there can be a constraint on the minimal and/or maximal number of instances. No instance number constraint is permitted on existing fields, and instance number constraint can be updated with more relaxed range. Change of a field's order within its sibling is not considered incompatible.

Once a schema is updated on the Local Server, it will be automatically republished to the Central Server (if any) onto which the schema has been published. This will ensure the Central Server always maintains up-to-date versions of schemas.

Publish and Unpublish

A schema can be published to one or multiple Central Servers if the user has a publisher role and there is no newer version of this schema on the Central Server. There are three scenarios of schema publishing: i) Schemas can be manually published from the schema publishing interfaces; ii) The schema of a document is automatically published when a document is published. When a document is published, the Local Server will check the Central Server if the schema is available or up-to-date. Otherwise the schema is republished; iii) If a schema was published and is then updated with compatibility, the new version schema will be automatically republished on the Central Server and replace the last version. This will keep the schemas on the Central Server up-to-date.

A schema can be unpublished by a Local Server with the following conditions: i) the user has the "publish" role on the Local Server; ii) there is no document associated with it on the Central Server, and iii) the Local Server is the last publisher of the schema. The last condition is necessary, otherwise if the schema is used at multiple Local Servers, every Local Server can easily stop the publishing which can be against the sharing goal of the last publisher.

Import

A Local Server can import a shared schema from the Central Server if the user has the organizer role. When there is a new version of a schema on the Central Server, the new version schema will be automatically detected by a Local Server when there is a document being published from that Local Server.

Remove from Central Server

Users on the Central Server with the organizer role can remove a schema from the Central Server if no document on Central Server is using this schema. This can be used to cleanup non-used schemas.

Update, or create? As a schema keeps on evolving, the number of updates on a schema can be many. When it reaches certain threshold, the latest schema may be very different from the original schema, the data between the two schemas may be very different, and the data consistency does not make much sense any more. In this case, it may be desirable to create a new schema. SciPort provides a functionality to view change history of a schema, and users can create a new schema from an existing schema in the Schema Editor tool.

An Example of Uniform Schema Management

A user with organizer privilege on a Local Server L1 creates a schema S(V1) (Figure 10). The user may later find the schema not accurate, and makes changes to the schema. Since no document has been created and the schema has never been shared yet, the user may make arbitrary changes, including incompatible updates. Once the user has a stable usable version of the schema, the user begins to author documents based on this schema, and publishes documents to the Central Server C. Schema S(V1) will be automatically published to the Central Server. After some time, the user may need more information for their data or adjust existing fields, and need to update schema S. Since there are existing documents using the schema, and the schema was also published, the user can make only compatible update to schema S(V1). (If compatible update is not sufficient, the user has to create a new schema.) Once schema S(V1) is updated as S(V2), it will be automatically propagated to the Central Server to replace the last version S(V1). A user with organizer privilege on a Local Server L2 imports schema S(V1) after S(V1) is published on the Central Server, and documents then are created on L2 on this schema. Later S(V2) replaces S(V1) on the Central Server, which is detected when a document of schema S(V1) is published to C. The user at L2 will be prompted to synchronize the schema, and S(V1) is replaced by S(V2) on L2.

5.2 Distributed Multiform Schema Sharing

Uniform schema sharing provides data compatibility through a single uniform version of schemas across all servers, and maintains the ownership of schemas and provides controlled schema evolution. There can also be cases that multiple sites need to adapt certain schema to their own needs, and a uniform schema may not be feasible. To provide flexible schema evolution at each Local Server and support data compatibility on the Central Server, we provide a multiform schema sharing approach.

As shown in Figure 11, a "seed" schema S will be first created and made as public owned, and then published onto the Central Server. Each Local Server will be able to import this schema, and adjust the schema for its local use (S_{L1}, S_{L2}, etc.) Schema updates from each site will be conditionally merged on the Central Server as S_M. The goal of the merging is to keep the schema version on the Central Server mostly relaxed, through the following merging conditions:

- If the changes are incremental structural changes, they will be merged. These include adding of new fields or new categories;
- If the constraints on the Central Server are more permissive than the new or updated constraints that the modification suggests, then the suggested changes are ignored by the Central Server;
- Structural removal changes such as field removal or category removal will be ignored and not merged.

In this way, each Local Server will maintain its local schema evolution while the Central Server will maintain a merged schema for shared data compatibility.

6 Collaborative Semantically Tagging of Data

Recently tagging has become a popular method to enable users to add keywords to Internet resources, thus to improve search and personal organization Marlow et al. (2006). In SciPort, data are hierarchically organized based on a tree based organizational structure. In practice, it can be very helpful that users can provide additional classification of data, by assigning tags from a controlled vocabulary to documents. Each document can be flexibly annotated with one or more semantic tags, and tags themselves can be shared by users at both Local Servers and the Central Server.

Semantically tagging adds additional semantics to documents, which not only provides semantics based query support, but also enhances semantic interoperability of shared data from multiple distributed sites.

6.1 Tagging from Controlled Vocabulary

Free tagging is used by most Web-based tagging systems, where users can define arbitrary tags. One issue for free tagging is that since there is no common vocabulary among these tags, there can be semantic mismatch between tags.

Instead, SciPort chooses to annotate data using semantic tags coming from predefined ontology or controlled vocabulary, such as NCI Enterprise Vocabulary Services (EVS)

Figure 12 Collaborative Tag Management

Figure 13 An Example of Automatic Tag Lookup

NCI Enterprise Vocabulary Services (EVS) (http://ncicb.nci.nih.gov/NCICB /infrastructure /cacore_overview /vocabulary). By standardizing tags using such controlled vocabulary, the system can then provide a controlled set of semantic tags. Thus data can be categorized into multiple semantic groups, which makes it possible to express queries based on common semantics. For instance, when a user authors a document of a study on breast cancer, he assigns a tag "Stage_II_Breast_Cancer" (ID: 18077), and another tag "Cancer_Risk" (ID: 7768). A problem arises that expensive lookup through vocabulary may simply prevent users from using it.

Next we show that with a collaborative tag management system, we can provide automatic tag lookup in a controlled vocabulary repository by caching previously retrieved tags in the Central Server with Ajax technology.

6.2 Collaborative Semantic Tag Management

Caching tags is to exploit locality inherent to the subset of the vocabulary that is used within a group of researchers. Since a collaborative research consortium often focuses on solving a single significant problem, the vocabulary is quite smaller than the standardized vocabulary. For instance, the NCI Thesaurus NCI Enterprise Vocabulary Services (EVS) (http://ncicb.nci.nih.gov/NCICB /infrastructure /cacore_overview /vocabulary) is 77MB in size. In SciPort, we provide a cached vocabulary repository on the Central Server, thus previous-retrieved tags from the standardized vocabulary is shared among all users. Cached vocabulary on the Central Server makes it very efficient to search for a tag in the vocabulary: instead of searching for a tag at a remote large vocabulary all the time, previous used tags in the tag cache repository can be searched first.

Figure 12 shows the architecture of collaborative tag management. At each Local Server, there is a tag repository that manages all tags on this Local Server; there is a remote NCI controlled vocabulary database from which users can dynamically search and retrieve tags. Once a tag is defined on the Local Server, it will be automatically cached at the Central Server tag repository. When a user wants to associate a tag to its data, he can dynamically search and select the tag, through automatic tag lookup, as discussed next.

6.3 Semantic Annotation of Documents with Tags

By taking advantage of the Ajax technology, we provide automatic tag lookup while a user wants to add a tag. As the user is typing in a keyword on a web page to search for a tag, there will be an automatic tag lookup from three resources: the local tag repository, the cached/shared tag repository on the Central Server, and the NCI controlled vocabulary. Implemented through Ajax, the lookup is performed in the background, and retrieved tags are displayed in the order of local tags, cached tags, and remote tags. The lookup will dynamically load a dropdown list of tags from which the user can pick up (Figure 13). By dynamically sending asynchronous tag queries to tag repositories, users can immediately select a desired tag to label his data, instead of opening multiple browser windows to do

Figure 14 A Screenshot of a SciPort Local Server Deployed at University of California, Irvine

separate searches. With this technology, we are able to support using tags from a controlled vocabulary or shared repository by providing a convenient interface.

6.4 Semantically Querying and Browsing Data through Semantic Tags

Once documents are annotated with semantic tags, it is now possible for users to semantically browse data by clicking on a semantic tag, or use tagged fields as additional constraints for specifying queries. For example, a user might want to search all documents tagged with "Stage_II_Breast_Cancer", and a type of "Breast Cancer" can generate a drop down list of semantic tags and the user can quickly use the right tag to specify queries. Semantic annotation can also help to classify data. Since semantic tags coming from an ontology are hierarchically organized in the ontology, it is possible to generate an ontology tree based view of all documents based on annotated ontology concepts on documents.

7 Implementation

7.1 Software

SciPort is built with J2EE and XML, running on Apache Tomcat servers and Oracle Berkeley DB XML database server (open source) or IBM DB2 XML database. The system is OS neutral and has been tested on Windows, Linux, and MacOS. It uses standard protocols, including XML, XSLT, XPath and XQuery, and Web Services. SciPort is a lightweight application and can be easily deployed and customized by users.

SciPort is a rich Web-based application, thus it is possible for users to use it at any place and at any time. SciPort supports major Web browsers including Internet Explorer, Firefox, Opera and Safari. Taking advantage of Web 2.0 technologies such as Ajax, SciPort provides rich application capabilities with smooth user experience.

A salient characteristic of SciPort is that the system is highly adaptable. Through customizable metadata schemas, hierarchy organization, and sharing architecture, SciPort can be easily setup for managing biomedical research data and providing a data sharing network.

Figure 14 shows the screenshot of the front page of a Local Server deployed at University of California, Irvine, while the Central Server is located at Siemens Corporate Research located at Princeton, New Jersey.

7.2 Deployment

SciPort was initially adopted by the Network for Translational Research on Optical Imaging (NTROI) for collecting and sharing data across several institutions (University of California, Irvine, University of Pennsylvania, Datmouth, Stanford and University of California, San Francisco. It is again adopted by the second cycle of Network for Translational Research (Washington University, University of Texas Health Science Center, Houston, University of Michigan, and Stanford University). SciPort is also used for sharing pathology images and their annotations for a large scale prostate cancer multi-modality study at University of Pennsylvania, the State University of New Jersey, Rutgers, and Siemens Corporate Research.

SciPort was also adopted for supporting Siemens Healthcare for collecting and sharing large scale biomedical research data from university and hospital research partners.

8 Discussion

The design and development of SciPort has been an iterative process and driven by numerous discussions with biomedical researchers and users. During the process, we have learned many lessons.

Usability. A critical requirement for software tools from biomedical researchers is the usability of the system. This includes intuitive user interfaces and workflows for data authoring, sharing and querying, and the ease of system setup. When the initial version of SciPort was released, the pilot users complained about multiple clicks needed to navigate through multiple web pages, and showed strong desires of simpler interfaces and fast response. Especially, a single click data publishing is desired, otherwise users gradually get frustrated to share their data. The Web based user interfaces then evolved from static web pages (jumping from one page to another) based interface to Ajax based interface, and then another Flex based interface is developed to provide even smoother user experience. Our conclusion is that smooth user experience of data sharing tools could much improve the interest for researchers to share their data.

Generalization and Customization. Although SciPort is currently a software platform and is not tied to specific applications, the initial version of SciPort (Version 1.0) was designed and developed for one single project with static data forms. Soon we realized that even for a single project, the schemas were dynamic, and new data forms were occasionally required. And researchers from other projects were using totally different data forms. Besides, the data are heterogenous, ranging from structural data to medical images, to files, etc. This motivated us to redesign SciPort as a platform that could provide easy and flexible customization. This is challenging as schema evolution is very difficult to support in traditional relational database management systems based on tables. The emergence of native XML database technologies makes it possible to support customizable schemas, as native XML DBMSs have a significant advantage on supporting schema evolution. Our data models, queries and data management are fully based on XML.

Integration with Other Systems. In the past, we also worked on extending SciPort so data managed in SciPort could be integrated into other infrastructures such as caBIG. We developed a SciPort-NBIA National Biomedical Imaging Archive (https://cabig.nci.nih.gov/tools/NCIA) bridge to support such integration.

9 Related Work

Computer supported collaborative work has been increasingly used to support research collaborations. A review of collaborative systems is presented in Bafoutsou and Mentzas (2002), and a review of taxonomy of collaboratory types is discussed Bos et al. (2007). SciPort belongs to the Community Data Systems based on this classification. Lee et al review collaboratory concepts relevant for collaborative biomedical research **?** and analyze the major challenges. In Myneni and Patel (2010), a collaborative information management system is discussed. A context-based sharing system is proposed in Chin and Lansing (2004)

to support more data-centric collaboration than tools oriented ones, and this motivates our metadata oriented approach for data modeling and sharing in SciPort.

With the increasing collaboration of scientific research, collaborative cyberinfrastructures have been researched and developed. Grid-based systems(such as caBIG® –cancer Biomedical Informatics Grid caBIG: cancer Biomedical Informatics Grid (http://caBIG.nci.nih.gov/), Biomedical Informatics Research Network (BIRN) Biomedical Informatics Research Network (http://www.nbirn.net/)) provide infrastructures to integrate existing computing and data resources. They rely on a top down common data structures. The iPlant Collaborative (iPC) Cyberinfrastructure for the Biological Sciences: Plant Science Cyberinfrastructure Collaborative (PSCIC) (http://www.nsf.gov/pubs/2006/nsf06594/nsf06594.htm) is a cyberinfrastructure project recently funded by US NSF. iPlant has a more focus on the human side of the infrastructure. myGrid **?** is a suite of tools for e-Science, with a focus on workflows. A review on cyberinfrastructure systems for the biological sciences is discussed in Stein (2008). Grid based systems are more used on sharing computing and storage resources, P2P is more used on sharing data Foster and Iamnitchi (2003). MIRC MIRC (http://mirc.rsna.org) is an example of P2P based data system for authoring and sharing radiology teaching files.

While Grid based systems are more used on sharing computing and storage resources, P2P is more used on sharing data Foster and Iamnitchi (2003). MIRC MIRC (http://mirc.rsna.org) is a popular pure-P2P based system for authoring and sharing teaching files.

A publish and subscribe architecture for distributed metadata management is discussed in Keidl et al. (2002), which focuses on the synchronization problems. In Taylor and Ives (2006), an approach of bottom-up collaborative data sharing is proposed, where each group independently manages and extends their data, and the groups compare and reconcile their changes eventually while tolerating disagreement. Our approach takes an approach in between the bottom-up and top-down approaches, where each group manages their data, but also achieves as much agreement on schemas as possible through controlled schema evolution.

In Rader and Wash (2008), influence on tag choices is analyzed from del.icio.us. Our approach focuses on semantic tagging from a controlled vocabulary instead of free tags. In Pike and Gahegan (2007), different approaches for representing scientific knowledge is discussed.

Extensive work has been done in data integration and schema integration Halevy et al. (2006); Doan and Halevy (2005). In Beynon-Davies et al. (1997), a collaborative schema integration system is discussed for database design. Our system takes a proactive approach where schema and data consistency is enforced during data authoring and schema authoring.

10 Conclusion

Contemporary biomedical research is moving towards multi-disciplinary, multi-institutional collaboration. These lead to strong demand for tools and systems to share biomedical data. This drives the development of SciPort – a Web-based data sharing platform for collaborative biomedical research. To support meaningful abstraction of data, SciPort provides a generic metadata model for users to conveniently define their own metadata schemas. SciPort provides an innovative lightweight hybrid data sharing architecture which combines the benefits of the centralized approach and the distributed approach. Investigators

are able to flexibly manage their data and schema sharing, and quickly build data sharing networks. To enable data consistency for data sharing, SciPort provides comprehensive collaborative distributed schema management. Through semantic tagging, SciPort further enhances semantic interoperability of shared data from distributed sites.

References

Myneni, S. and Patel, V.L. (2010) 'Organization of biomedical data for collaborative scientific research: a research information management system', *International Journal of Information Management*, Vol. 30, No. 3, pp.256–264

Bekelman, J.E., Li, Y. and Gross, C.P. (2003) 'Scope and impact of financial conflicts of interest in biomedical research: a systematic review', *JAMA*, Vol. 289, No. 19, pp.454–65

Stein, L.D. 2008 'Towards a cyberinfrastructure for the biological sciences: progress, visions and challenges', *Nature Reviews Genetics*, Vol. 9, No. 9, pp.678–688.

NIH Statement on Sharing Scientific Research Data, http://grants.nih.gov/grants/policy/data_sharing/

Policy for sharing of data obtained in NIH supported or conducted genome-wide association studies, http://grants.nih.gov/grants/guide/notice-files/not-od-07-088.html

Network for Translational Research (NTR): Optical Imaging in Multimodality Platforms, http://imaging.cancer.gov/programsandresources/specializedinitiatives/ntroi

Piwowar, H., Becich, M., Bilofsky, H. and Crowley, R. (2008) 'PLoS medicine', *Sept, No. 9, Towards a Data Sharing Culture: Recommendations for Leadership from Academic Health Centers*, Vol. 5.

Birnholtz, J.P. and Bietz, M.J. (2003) 'Data at work: supporting sharing in science and engineering', *GROUP*, pp.339–348.

Data Sharing & Intellectual Capital (DSIC) Workspace, https://cabig.nci.nih.gov/working_groups/DSIC_SLWG

Getting Connected with caBIG, https://cabig.nci.nih.gov/getting_connected/

Digital Imaging and Communications in Medicine (DICOM), http://medical.nema.org/

The Cancer Genome Atlas (TCGA) Data Portal, http://cancergenome.nih.gov/dataportal

National Biomedical Imaging Archive, https://cabig.nci.nih.gov/tools/NCIA

caBIG: cancer Biomedical Informatics Grid, http://caBIG.nci.nih.gov/

Biomedical Informatics Research Network, http://www.nbirn.net/

Cyberinfrastructure for the Biological Sciences: Plant Science Cyberinfrastructure Collaborative (PSCIC), http://www.nsf.gov/pubs/2006/nsf06594/nsf06594.htm

MIRC, http://mirc.rsna.org

Foster, I. and Iamnitchi, A. (2003) 'On death, taxes, and the convergence of peer-to-peer and grid computing', *IPTPS'03*.

Keidl, M., Kreutz, A., Kemper, A. and Kossmann, D. (2002) 'A publish & subscribe architecture for distributed metadata management', *ICDE*.

Taylor, N.E. and Ives, Z.G. (2006) 'Reconciling while tolerating disagreement in collaborative data sharing', *SIGMOD*.

Rader, E.J. and Wash, R. (2008) *CSCW*, pp.239-248, Influences on tag choices in del.icio.us.

Halevy, A., Rajaraman, A. and Ordille, J. (2006) *VLDB, Data integration: the teenage years*, http://portal.acm.org/citation.cfm?id=1182635.1164130

Doan, A. and Halevy, A.Y. (2005) Semantic Integration Research in the Database Community: A Brief Survey, *AI Magazine*, Vol. 26, No. 1, pp.83–94.

Beynon-Davies, P., Bonde, L., McPhee, D. and Jones, C.B. (1997) 'A collaborative schema integration system', *Comput. Supported Coop. Work*, Vol. 6, No. 1, Norwell, MA, USA, pp.1–18, issn = 0925-9724.

Wang, F. and Vergara-Niedermayr, C. (2008) 'Collaboratively Sharing Scientific Data', *CollaborateCom*, pp.805–823.

Chin Jr., G. and Lansing, C.S. (2008) 'Capturing and Supporting Contexts for Scientific Data Sharing via the Biological Sciences Collaboratory', *CSCW*, ISBN 1-58113-810-5.

W3C XML Query (XQuery), http://www.w3.org/XML/Query

Wang, F., Hussels, P. and Liu, P. (2009) 'Securely and flexibly sharing a biomedical data management system', *SPIE*.

Marlow, C., Naaman, M., Boyd, D. and Davis, M. (2006) 'Position Paper, Tagging, Taxonomy, Flickr, Article, ToRead', *Collaborative Web Tagging Workshop*.

NCI Enterprise Vocabulary Services (EVS), http://ncicb.nci.nih.gov/NCICB /infrastructure /cacore_overview /vocabulary

NCI Enterprise Vocabulary Services (EVS), http://ncicb.nci.nih.gov/NCICB /infrastructure /cacore_overview /vocabulary

Bafoutsou, G. and Mentzas, G. (2002) 'Review and functional classification of collaborative systems', *International Journal of Information Management*, Vol. 22, No. 4, pp.281–305.

Bos, N., Zimmerman, A., Olson, J., Yew, J., Yerkie, J. and Dahl, E. et al. (2007) 'From shared databases to communities of practice: A taxonomy of collaboratories', *Journal of Computer-Mediated Communication*, Vol. 12, No. 2.

Myneni, S. and Patel, V.L. (2010) 'Organization of biomedical data for collaborative scientific research: A research information management system', *International Journal of Information Management*, Vol. 30, No. 3, June, pp.256–264.

myGrid, http://www.mygrid.org.uk/

Pike, W. and Gahegan, M. (2007) 'Beyond ontologies: Toward situated representations of scientific knowledge', *International Journal of Human-Computer Studies*, Vol. 65, No. 7, pp.674–688.