Representation of immunofluorescence data for high-throughput analysis

G. Fontenay, G. Cong, and B. Parvin
Imaging and Informatics
Lawrence Berkeley National Laboratory
Berkeley, Ca 94720

One endpoint of integrated systems biology is protein localization studies, which are realized through different modes of microscopy and imaging. Through an appropriate mode of microscopy and molecular marker technology, imaging can reveal protein localization, tissue architecture, and cellular morphology under a variety of experimental factors and for different biological materials. Typically, a specific study may have several experiments associated with it, where each experiment targets a unique endpoint. Each experiment may utilize (1) a microscopy mode, (2) a specific imaging agent, (3) a model organism, or (4) a sample preparation protocol. This is very similar to microarray studies; however, in some areas, the data model can be truncated, and in other areas, the data model needs to be expanded. The data management requirements originate not only from grouping of experimental annotations, but also from large sample size that is often necessary for addressing biological heterogeneity in protein localization and tissue architecture.

Toward these objectives, analytical and informatics techniques have been developed to annotate experimental designs, convert images to quantitative features, and present different views of pertinent information through the Web. BioSig, a system that integrates these tools and techniques with web-accessible data import and storage, is in use at laboratories here at LBNL. We are working towards a new version of this system that leverages emerging new standards in biologically-relevant ontologies and ontology-related tools.

Ontologies and controlled vocabularies are required for effective and uniform annotation and querying of biological data. Ontologies and controlled vocabularies in use include NCICB databases, the MAGE/MGED ontology for microarray studies, and in-house-specific terminologies. The MAGE model and the MGED ontology, for example, are used to provide a concise way of defining the experimental factors (e.g., cellline, radiation types and dosage, and other treatments) and protocol associated with assay development (e.g., plating, incubation time with a reagent at a specific concentration, number of washout, and fixation). Experimental factors have values that may take the form of measurements or entries within an ontology. Coupling of experimental factors with ontologies can accommodate unpredictable complexity where new experimental factors can be developed and incorporated. Within such a controlled framework, experimental factors can be captured in machine-readable templates that allow for

reuse, modification, and extensibility of experimental designs. Such features can greatly facilitate annotation and design of biological experiments, but development of effective interfaces for experimental data entry can still be difficult. Towards improving interface functionality and usability, while decreasing development time, we have created a web-usable framework for data entry to different databases. This framework facilitates the development of customized semantic views that are both browsable and editable. From a functional perspective, one is not interested in the complete data model on the server side, but in browsing a series of easily sortable and constrained high-level views, utilizing efficient data entry and query-by-example forms. Users typically want features that automate repetitive tasks while allowing for considerable editing flexibility. Implementing these functions with the typical web browser scripting code is difficult to manage and becomes especially error-prone because of differences in browser implementation. To faciliate this functionality, the design incorporates Java applets using Protege. Protege has emerged as a popular Java-based software framework for ontology development, knowledge representation, and data population. The frame-based programming model and forms facility of Protege have been leveraged to construct client-side interfaces and customized forms for the semantic views. Protege allows for the development of plug-ins and modular pieces of code that can work both within a web browser as applets and as standalone applications. A Protege plug-in has been developed to interact with a backend database and allow for on-demand browsing and population of the forms. Interaction with a specific database is provided through a customized Java library that defines the semantic views and required database server statements. Subsequent data validation is provided within the Java-based framework. Additionally, Protege facilitates incorporation of controlled vocabularies and ontologies. For example, the MGED ontology, which supports the MAGE object model is programmatically accessible through Protege tools, and MGED ontology terms are now integrated into our system.

Protégé and related tools that leverage interoperability languages such as RDF/XML, will be used to add additional features and further development with standardization and extensibility in mind. As a machine-readable and universal format, RDF (Resource Description Framework) was designed to enable the representation of information about any type of resource. In this next iteration of BioSig, RDF will be used as a flexible way of creating and developing extensions of our data model and ontologies without modifying our database schema. Such extensions could involve, for example, the capture of unpredictable experimental data or the creation and development of new classification ontologies. We envision the creation and development of many new ontologies as research methods and collaborations grow. We will leverage Protege and related tools in order to manually and/or programmatically create, manage, populate and share ontological models.

These models can then used to annotate existing content and be populated with

references to internal or external ontology and database entries. These references will be in the form of Life Science Identifiers, a standard way of naming and accessing data and related resources. Resolution of these references involves returning the data and metadata associated with the reference within a networked RDF graph containing LSID references to other related data and metadata. LSID resolution can then allow intuitive browsing and drilling-down of related data and metadata. Leveraging previous work and experience in SVG interfaces, upcoming work on our system will allow for integrated LSID browsing and customized export of dynamically-created graphical LSID-populated documents based on ontology and database content. These interactive documents will providing an integrated view of experimental data and analysis, including images, annotation, and features. These tools will use the Protege API and utilities wherever possible, thereby facilitating development and working towards a standardized platform.

We have just begun to use Protege in our development and modeling efforts and it has proven to be effective in these contexts. Existing database interfaces through Protege were not sufficient, however, as we needed to create our own database tables and be able to create customized, constrained interfaces in certain cases. Through this current development effort we will determine how new components might be developed to facilitate the further abstraction of such time-consuming and error-prone tasks such as data entry interface development and database interaction as well as develop tools for visualization and navigation of experimental data.

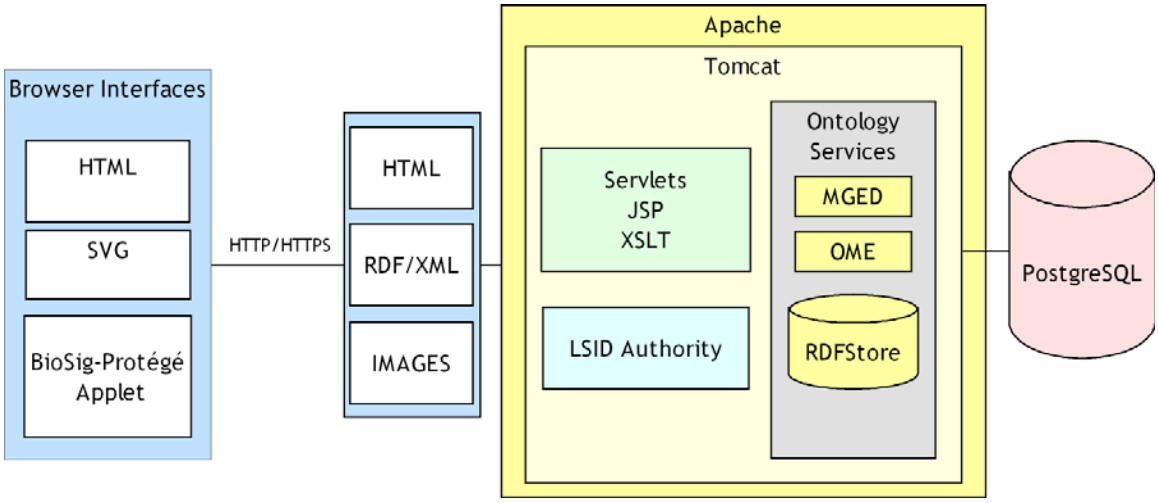

Figure: Presentation, service, and data layer for the BioSig Imaging Bioinformatics System