Model-Based Information Integration in a Neuroscience Mediator System

Bertram Ludäscher* Amarnath Gupta* Maryann E. Martone‡

*San Diego Supercomputer Center, UCSD {gupta,ludaesch}@sdsc.edu
‡Department of Neurosciences, UCSD mmartone@ucsd.edu

1 Overview

We present the information mediator prototype called KIND[1] [GLM00], recently developed as part of an integrated Neuroscience workbench project at SDSC/UCSD within the NPACI[2] project. The broad goal of the workbench is to serve as an environment where, among other tasks, the Neuroscientist can query a mediator to retrieve information from across a number of information sources, and use the results to perform her own analysis on the data.

The KIND mediator is an instance of a novel *model-centered* mediator architecture that extends current XML-based mediator approaches by incorporating a semantic *model* of an information source as an integral part of the mediation process. Thus, by model we mean a combination of (i) a *conceptual model* of the source data, for example a UML or EER model, and (ii) any additional conceptual-level *knowledge* about the source as expressed through IDB (intensional database) rules. While current mediators for information integration address and solve the problems of *syntactic* and *structural* heterogeneities amongst sources using a semistructured data model like XML [GMPQ+97, CDSS98, GMW99], the *semantic* integration problem remains and a mediation engineer has to come up with integrated view definitions on top of the different XML views and DTDs exported by sources. In contrast to this structure-centered approach, the model-centered mediator architecture provides a framework to the mediation engineer in which the integrated domain model (i.e., both its conceptual schema and data instances) can be defined in terms of the domain-level semantic models of sources using a high-level declarative language.

Our development of a model-based mediator was driven by the need to integrate scientific databases like those of the Neuroscience workbench, where source data comes from different "semantic worlds", often sharing few if any attributes. Thus, in contrast to the more traditional "one world" integration scenario (say integrating information from online book-shops) where despite differences in the local schemas, the sources share the same domain of discourse, here we need to integrate across different domains like *neuroanatomy*, *protein properties* and *ion-currents in nerves*. Since

Proceedings of the 26th VLDB Conference, Cairo, Egypt, 2000.

[1] ***K**nowledge-based **I**ntegration of **N**euroscience **D**ata*

[2] ***N**ational **P**artnership for **A**dvanced **C**omputational **I**nfrastructure*, http://www.npaci.edu

these sources are scientifically related in the physical world through (expert and common) knowledge, the integration process using a mediator becomes feasible if those associations between objects from different domains are definable at the level of the domain model.

The system we demonstrate includes the following salient features:

- We use XML as the uniform format for exchanging instance data (relational, object-oriented, semistructured) and *model data* (like UML and EER schemas). In particular UML models of sources are represented in XML using XMI [XMI99].

- The model-integration language is F-Logic (short: FL) [KLW95], a rule-based, object-oriented language that allows to represent, query, and reason with not only semistructured data like XML trees but also conceptual-level information, i.e., object-oriented data and *schema*. Query evaluation at the mediator level is based on the FLORA engine [LYK99, YK00], enabling a *virtual integrated view* approach, and on the MIX*m* system for querying XML-enabled sources via the XML query language XMAS [BGL+99, LPV00].

- For sources that export a conceptual model CM (in XMI) to the mediator, a semantics-preserving DTD_{CM} is derived. At runtime, the mediator can automatically construct CM (i.e., populate the classes of CM and validate against the integrity constraints of CM) and then integrate across different CMs, provided the sources export XML data which is valid wrt. DTD_{CM}. Otherwise, declarative FL rules are used to map instances from the source-specific DTD to DTD_{CM} and thus again to instances of CM. In this way, sources are not only queryable as labeled ordered trees (as is the case with pure XML-based query languages), but as *domain-level object-bases*; in particular information about *generalizations* (class hierarchy), *properties of relationships* (cardinalities, relationship types like aggregation and composition), and *application-specific integrity constraints and rules* all become accessible for defining the integrated view at the mediator.

2 Demonstration

The demonstration presents a Neuroscience application where the KIND mediator integrates three data sources and two domain knowledge sources (cf. Figure 1). The central component of the architecture is the KIND mediator which uses the FLORA system [LYK99] for executing the integration rules. The mediator has several "plug-in" modules for interfacing with the runtime environment. For example, the module XML_2FL maps XML to equivalent FL objects. For sources like CAPROT which export a conceptual model CM, there is a XMI_2FL plug-in, which takes the conceptual model CM(S) of source S and produces an FL model of it. As we will show in the demo, the plug-in can generate from CM(S) (which is here given in the XMI syntax for UML) the FL equivalent of CM(S), i.e.,

- a *class signature* augmented with a set of *integrity constraint rules* for capturing the semantics of CM(S), and

- a set of *instantiation rules* which are used at runtime to populate CM(S) from the XML data exported by S.

For sources that do *not* export a CM, the mediation engineer has to reverse-engineer the CM from the exported XML DTD. Based on this, the instantiation rules for CM are derived. The integrated view INSM (Figure 1) is exported to the user or application as XML; it is defined in the declarative FL rule language on top of the CMs of all involved sources. It is important to notice that the mediator does *not* materialize the complete CMs but computes the relevant CM instances on the fly at query

evaluation time, based on the user query against the integrated view. This *virtual integration approach* results from the use of the top-down query evaluation engine FLORA. In addition, for integration scenarios where materialization of CM(S) is advantageous, we may use the bottom-up FL engine FLORID [FLO, LHL+98].

XML sources are accessed through the MIX*m* mediator system which can evaluate complex XML queries expressed in XMAS[3]. The "structure-level" XML mediator MIX*m* is mainly used to produce *on demand* virtual XML views on top of potentially huge XML sources [LPV00]. Here, "on demand" means that the construction of the virtual XML view is driven by the navigation of the client (i.e., the KIND mediator) within the view.

The demonstration involves the following sources:

- PROLAB stores results of image analysis from protein labeling experiments performed on one or more regions of the brain. They contain measurements of protein concentration in different *segments* of light and electron microscope images. Segments are organized into collection objects called *anatomical structures*. An anatomical structure may contain other anatomical structures.

- DENDREC is a database of volumetric reconstructions of nerve compartments called *dendritic spines* and *dendritic shafts*. The database records a number of measurements performed on these volumetric objects. As in PROLAB, dendritic shafts and spines are spatially composed into larger volumetric objects called *dendrites*. Both PROLAB and DENDREC are implemented on an ORACLE8i database and wrapped to produce XML output.

- CAPROT is a Web-accessible database of calcium binding proteins.[4] In contrast to PROLAB and DENDREC whose conceptual models had to be reverse-engineered, CAPROT publishes a conceptual model (EER) of its data which we have mapped into an UML/XMI encoding so it is accessible to the mediator through the FL_2XMI plug-in. CAPROT contains protein properties, the organisms and tissues and cells where they are found, as well as the known functions of the proteins.

- TAXON is a database containing the scientific classification of the animal kingdom, along with the common names of the animals. This database is an example for a knowledge source that allows to eliminate "semantic holes" of otherwise unrelated source models. Here, it bridges the gap between the scientific names used in CAPROT and the common names used in the other data sources. As shown in the demonstration, this also allows users to query on animals by higher order group names (e.g, mammals instead of humans and mice).

- Finally, ANATOM defines a hierarchy of biological organ names starting from the brain down to cells and subcellular components. It also contains a set of rules defining different groupings and classification hierarchies of organs. For example, it states how the set of brain regions can be grouped structurally and functionally. It also specifies transitive properties of *is-a* and *has-a* relations which are beyond the expressive power of most XML query languages.

An example user query spanning different sources (and which, therefore, could not be answered without the mediator) is:

- *"Find the cerebellar distribution of all rat proteins with more than 90% amino acid homology with the human NCS-1 protein"*, or

- *"Like before, but give the distribution of this protein or its homologs in other rodents."*

[3] ***X**ML **M**atching **A**nd **S**tructuring Language* [BGL+99]

[4] `http://structbio.vanderbilt.edu/cabp_database`

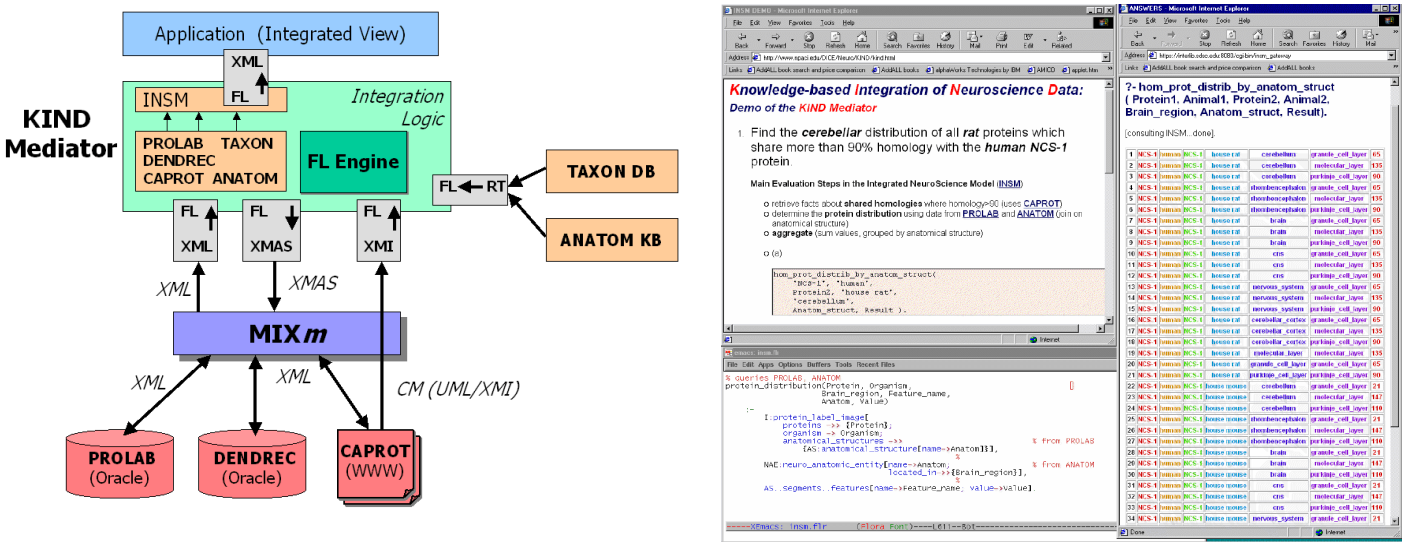

Figure 1: **left:** architecture of the KIND mediator; **right:** snapshot of the Web interface

As part of the demonstration, we will show the different system components involved in the evaluation of such queries, along with the corresponding data and schema definitions and the mappings between them. A screenshot of the current Web interface is shown on the right in Figure 1.

References

[BGL+99] C. Baru, A. Gupta, B. Ludäscher, R. Marciano, Y. Papakonstantinou, P. Velikhov, and V. Chu. XML-Based Information Mediation with MIX. In *ACM Intl. Conference on Management of Data (SIGMOD)*, Philadelphia, PA, 1999. exhibition program.

[CDSS98] S. Cluet, C. Delobel, J. Simeon, and K. Smaga. Your Mediators Need Data Conversion! In *ACM Intl. Conference on Management of Data (SIGMOD)*, pp. 177–188, 1998.

[FLO] FLORID Homepage. `www.informatik.uni-freiburg.de/~dbis/florid/`.

[GLM00] A. Gupta, B. Ludäscher, and M. E. Martone. Knowledge-Based Integration of Neuroscience Data Sources. In *12th Intl. Conference on Scientific and Statistical Database Management (SSDBM)*, Berlin, July 2000. IEEE Computer Society.

[GMPQ+97] H. Garcia-Molina, Y. Papakonstantinou, D. Quass, A. Rajaraman, Y. Sagiv, J. Ullman, V. Vassalos, and J. Widom. The TSIMMIS Approach to Mediation: Data Models and Languages. *Journal of Intelligent Information Systems*, 8(2), 1997.

[GMW99] R. Goldman, J. McHugh, and J. Widom. From Semistructured Data to XML: Migrating the Lore Data Model and Query Language. In *ACM SIGMOD Workshop on the Web and Databases (WebDB)*, pp. 25–30, Philadelphia, 1999.

[KLW95] M. Kifer, G. Lausen, and J. Wu. Logical Foundations of Object-Oriented and Frame-Based Languages. *Journal of the ACM*, 42(4):741–843, July 1995.

[LHL+98] B. Ludäscher, R. Himmeröder, G. Lausen, W. May, and C. Schlepphorst. Managing Semistructured Data with FLORID: A Deductive Object-Oriented Perspective. *Information Systems*, 23(8):589–613, 1998.

[LPV00] B. Ludäscher, Y. Papakonstantinou, and P. Velikhov. Navigation-Driven Evaluation of Virtual Mediated Views. In *Intl. Conference on Extending Database Technology (EDBT)*, LNCS 1777, Konstanz, March 2000.

[LYK99] B. Ludäscher, G. Yang, and M. Kifer. FLORA: The Secret of Object-Oriented Logic Programming. Technical report, State University of New York, Stony Brook, June 1999. see also `www.cs.sunysb.edu/~sbprolog/flora/`.

[XMI99] XML Metadata Interchange (XMI). `www.omg.org/cgi-bin/doc?ad/99-10-02`, 1999.

[YK00] G. Yang and M. Kifer. FLORA: Implementing an Efficient DOOD System Using a Tabling Logic Engine. In *6th International Conference on Rules and Objects in Databases (DOOD)*, 2000.