

An Ontology-based Architecture for Integration of Clinical Trials Management Applications

Ravi D. Shankar, MS¹, Susana B. Martins, MD, MSc¹, Martin O'Connor, MSc¹,
David B. Parrish, MS², Amar K. Das, MD, PhD¹

¹Stanford Medical Informatics, Stanford University, CA, USA;

²The Immune Tolerance Network, Pittsburgh, PA, USA

Abstract

Management of complex clinical trials involves coordinated-use of a myriad of software applications by trial personnel. The applications typically use distinct knowledge representations and generate enormous amount of information during the course of a trial. It becomes vital that the applications exchange trial semantics in order for efficient management of the trials and subsequent analysis of clinical trial data. Existing model-based frameworks do not address the requirements of semantic integration of heterogeneous applications. We have built an ontology-based architecture to support interoperation of clinical trial software applications. Central to our approach is a suite of clinical trial ontologies, which we call Epoch, that define the vocabulary and semantics necessary to represent information on clinical trials. We are continuing to demonstrate and validate our approach with different clinical trials management applications and with growing number of clinical trials.

1. Introduction

Personnel involved with conducting complex clinical trials employ a myriad of software applications that meet the demands of managing clinical trials. Authoring tools, operational plan builders, study site management tools, participant and specimen tracking applications, and trial data analyzers address the functionality requirements at different stages of the trial —specification, planning, implementation and analysis. These applications vary widely in complexity, are generally autonomous, use very distinct trial knowledge representations, and generate enormous amount of data during the course of a trial. Many times clinical trial information is exchanged among these applications using less rigorous methods such as textual documents and email messages. The heterogeneity of the applications when combined with weak sharing of clinical trial semantics leads to variances in trial specification that percolate through the course of the trial and eventually manifest as gross inconsistencies in clinical trial data. Considerable time and effort are then spent in cleaning up and validating

the collected data before any analysis can even begin. Therefore, semantic integration of the varied applications becomes vital to improve the efficiencies of trial data collection and to ensure the quality of collected data.

Many model-based systems have been proposed to automate different areas in the domain of clinical trials and clinical guidelines. As we discuss later in Section 4, these systems do not amply address the need to integrate heterogeneous software applications that clinical trials personnel use. We have built an ontology-based architecture to support semantic interoperation of clinical trial software applications. At the core of our architecture is a suite of ontologies that we call Epoch that encodes knowledge about the clinical trial domain that is relevant to trial management applications. The ontologies define the vocabulary and semantics necessary for formal representation of clinical trials. The Epoch framework has software components that facilitate the sharing of the ontological formalisms among heterogeneous applications thus promoting semantic interoperability among these applications. We are tailoring our approach to meet the requirements in the management of clinical trials at the Immune Tolerance Network, or ITN [1], which develops new therapeutics for immune-mediated disorders. Our work is borne out of the collaborative research efforts of Stanford Medical Informatics (SMI) and the ITN in addressing the informatics needs of collecting, managing, integrating and analyzing clinical trial and immunoassay data. In this paper, we will discuss our ontology-based architecture and its components especially the Epoch ontologies. We will illustrate our framework by describing a usage scenario where applications interoperate to facilitate specimen tracking. We will further discuss how our approach supports the complete life-cycle of clinical trials management, and differs from prior and ongoing work in ontological solutions, especially in the areas of guideline based care and clinical trials management.

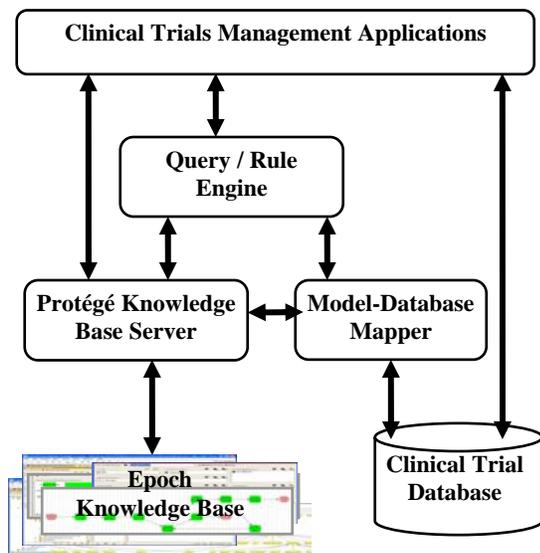


Figure 2 An ontology-based architecture that supports clinical trials management applications

The Epoch ontologies thus provide a common nomenclature and semantics required to support an integrated and consistent clinical trials management.

We have developed these ontologies in OWL (the Web Ontology Language proposed by W3C) [<http://www.w3.org/TR/owl-features/>] by building hierarchies of *classes* describing concepts in the ontologies and relating the classes to each other using *properties*. We use SWRL (the Semantic Web Rule Language) [<http://www.w3.org/Submission/SWRL/>] to specify rules that validate the constraints specified using the constraint expression ontology. Protégé [2] is a software tool that supports the specification and maintenance of terminologies, ontologies and knowledge-bases. We used Protégé to create the ontologies in OWL. We then encoded specific clinical trial protocols using Protégé's knowledge-acquisition facilities.

We have built an ontology-based architecture (Figure 2) that broadly supports three types of methods that clinical trials management applications can use. The knowledge acquisition methods allow users to encode specific protocols and related operational elements, and thus, to create the protocol knowledge base. Ontology-database mapping methods integrate the protocol and biomedical knowledge with clinical trial data including clinical results and operational data stored in the ITN data repository. Concept-driven querying methods support integrated data management, and can be used to create high-level abstractions of clinical data during analysis of clinical results. At the center of all these methods and the

applications that use these methods is the Epoch clinical trial ontologies. The *Epoch Knowledge Base* contains the Epoch ontologies and specific instantiations of the ontologies for different clinical trials. A *Knowledge Base Server* provides access to the contents of the ontology repository. We have developed a tool to generate XML renditions of the OWL knowledge base based on custom XML Schema. In the next section, we show how we employed this tool to configure a data collection application with information in the knowledge base. We are building other utility tools to support querying and rule execution. These tools will eventually be integrated with the knowledge base server. The *Clinical Trial Database* is a relational database system that stores data related to the implementation and execution of clinical trials. The types of data include participant enrollment data, specimen shipping and receiving logs, participant visits and activities, and clinical results. The *Model-Database Mapper* facilitates runtime access to the clinical trial database via a mapping ontology that connects data model concepts to database entities. The *Inference / Rule Engine* executes temporal and non-temporal constraints – that have been expressed as SWRL rules – in Epoch ontologies. The interface with the Model-Database Mapper allows SWRL rules to be executed on data stored in the clinical trial database. We are currently using JESS [<http://www.jessrules.com/>], a production rule-engine, to execute the rules. The *Clinical Trials Management Applications* interoperate via the Epoch components at syntactic, structural and semantic levels to support the management of clinical trials.

3. Semantic Interoperation among Applications Implementing Specimen Tracking

Biological specimens are collected from participants at different visits based on clinical assessments and mechanistic studies planned in the protocol. These specimens are then stored in pre-determined containers and shipped to bio-repositories. The specimens (or portions of them) are shipped to the core laboratories where specific assays are performed on the specimens. The ITN has contracted with Cimarron Software, Inc. to build a specimen workflow system called ImmunoTrak based on Cimarron's Laboratory Workflow Systems product. Clinical trial personnel at the sites will use the system to log participant's visit, specimen collection, shipping and receiving of bar-coded specimen containers, etc. ImmunoTrak can be configured using a graphical user interface or via an XML specification. The

configuration parameters include, the participant visit flow, the specimen container specification, list of participants, list of clinical and laboratory sites, and specimen workflow. The system should also be configured with the container manufacturer's report on the empty specimen containers shipped to the collection sites. The specimen tracking data that is collected by the system during the course of the trial is stored in a relational database. A Specimen Tracking application can then access the database to monitor the status of the specimen collection and processing.

Figure 3 shows the usage scenario employing the Epoch framework to specimen tracking. The first step is to specify the specimen workflow in the Protégé-OWL editor using relevant Epoch ontologies. Next, the *Knowledge Base Server* uses an XML Schema file to generate the configuration file for ImmunoTrak, the Specimen Workflow System. The Specimen Container application generates container specifications that form the basis of the manufacturer's report. During the course of the clinical trial, research coordinators at different clinical sites access ImmunoTrak to enter specimen collection data which is stored in the Clinical Trial Database. The Specimen Tracking Application employs the Model-Database Mapper to access the data via the Epoch data models. It can then satisfy user queries for specific specimen processing status, specimen collection inventory. It can also execute any validation rules or temporal constraints as specified in the ontologies on the tracking data using the production rule engine.

All the applications in this example work on the same set of semantic descriptions of specimen workflow

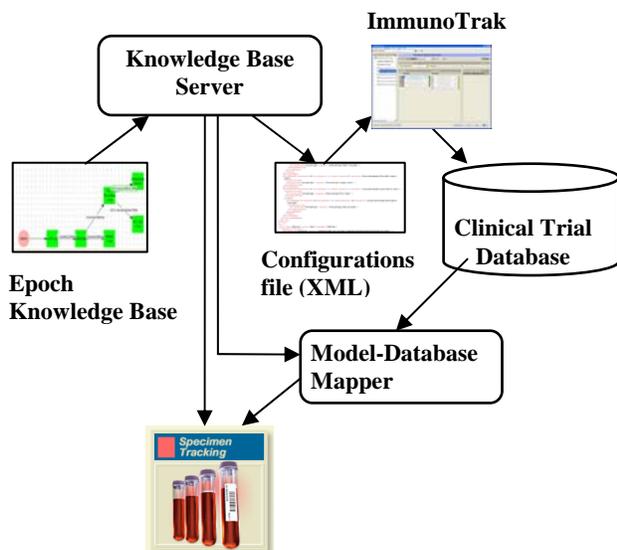


Figure 3 Epoch architectural components facilitate specimen tracking.

concepts found in the Epoch knowledge base. The applications are built by different vendors independent of the Epoch effort, and are pressed into service at different stages – specification, execution and monitoring – of the clinical trial. Using semantic mapping methodologies that we described earlier, these disparate applications share the clinical trial specifications that are created with the Epoch ontologies. The semantic chaining of the applications, from protocol specification, to data collection, to data analysis can improve standardization, data integrity, data integration and data analysis.

4. Discussion

We have built an ontology-based software architecture that supports the consistent specification and implementation of clinical trials undertaken by large, distributed research consortiums, such as the ITN. Many groups, internal and external to ITN, collaborate in facilitating the specification and implementation of the trials and related biological assay studies. We have encoded three ITN clinical trial protocols, and counting, using the Epoch ontologies. We have performed semantic mapping between the ontologies and the configuration formats of two applications that support specimen tracking and data visualization. We are continuing to implement other clinical trials management services such as site management, and study tracking. We are enhancing our ontologies to capture more complex temporal and logical constraints. We are investigating different ways of mapping ontologies to database elements in order to improve optimization of queries on clinical trial data. Currently, we use the Protégé-OWL editor to build the Epoch ontologies, and for domain specialists to enter a specification of a protocol. Unfortunately, domain specialists find it cumbersome and non-intuitive to use the generic user interfaces. We are building custom graphical user interfaces that hide the complexities of the knowledge models, and that facilitate guided knowledge-acquisition. The Epoch ontologies are publicly accessible from the Open Biomedical Ontologies (OBO) library hosted by the National Center for Biomedical Ontology (NCBO) [<http://www.bioontology.org/>]. We are also collaborating with NCBO with its Ontology of Clinical Investigations initiative.

In the past few years there has been a spurt of activities in designing computable formalisms of the domain of interest, with the aim of building software applications that will use these formalisms to provide improved, easily maintainable, and scalable services. The healthcare informatics community has made great

strides in the area of computer-interpretable clinical guidelines. Formal guideline models [3] such as EON, PROforma and SAGE support clinical decision-support systems that provide guideline-based clinical advisories at the point-of-care. In all these modeling frameworks, the formalisms vary in many dimensions – syntactic structure, semantic representation, domain range, reasoning complexity and computational technique. In most of these frameworks, the guideline model is at the core, providing a way to represent clinical knowledge found in clinical guideline documents. A guideline model interpreter applies patient data to the contents of the formal guideline knowledge base to generate patient specific clinical advisories. Thus in an automated guideline based care system, there is a tight coupling between the guideline model and the advisory application. Therefore, the semantic integration requirements in such a system are much simplified compared to the requirements with heterogeneous clinical trials management applications that we described earlier.

In the area of clinical trials, there have been sporadic initiatives to automate focused areas of clinical trials management. The Asgaard [4] framework uses a time-oriented, intention-based, skeletal-plan specification language to represent clinical protocols. The temporal plans can then be used to create treatment plans in accordance with the protocol specification. TrialBank [5] captures data on randomized clinical trials using a formal model (RCTSchema) that can then support systematic reviewing of the specifications and the results of the trials. FastTrack [6] provides a model-based infrastructure for optimizing operational design and management of clinical trials. It has modeling constructs to capture protocol issues that tend to cause operational difficulties. Clinical Data Interchange Standards Consortium (CDISC) is a multidisciplinary organization that is establishing new standards for electronic acquisition, exchange, storage, archival and submission of clinical trials data and metadata for medical and biopharmaceutical product development. A spin-off from the CDISC effort is the BRIDG project [<http://www.bridgproject.org/>] that is mainly sponsored by National Cancer Institute's caBIG initiative and is in partnership with HL7. The BRIDG project's main aim is to develop a comprehensive model that defines functions and behaviors throughout standard clinical trials so as to bridge standards within the clinical research and healthcare domains. The use of the model in production clinical trial applications is tightly coupled with NCI's infrastructure components such as Common Data Elements (CDEs) and Enterprise Vocabulary Services (EVS). It is not clear

how the BRIDG model will influence applications built outside the NCI fold. Also, the BRIDG model is underrepresented in computable formalisms to specify temporal annotations and logical constraints that are the core of clinical trial protocols.

The Epoch architecture employs semantic web technologies to facilitate integration of heterogeneous applications that share the semantics of the clinical trials and not necessarily the representation formalisms. There are several efforts [7,8] from the semantic web community that propose similar ontology-based architectures to integrate distributed information resources. The ITN applications that we are integrating have been built by different software vendors using their proprietary information representations. Thus our approach can scale with newer applications, and, for the same reasons, can be adapted to other clinical trial organizations.

Acknowledgement

This work is supported in part by the Immune Tolerance Network, which is funded by the National Institute of Health under the grant NO1-AI-15416).

References

- [1] Rotrosen D, Matthews JB, Bluestone JA. The immune tolerance network: a new paradigm for developing tolerance-inducing therapies. *Journal of Allergy and Clinical Immunology*, 2002 Jul;110(1):17-23.
- [2] H. Knublauch, R.W. Fergerson, N.F. Noy, and M.A. Musen, "The Protégé OWL Plugin: An open development environment for semantic web applications", *Proc Third ISWC (ISWC 2004)*, Hiroshima, Japan, 2004, pp. 229-243.
- [3] M. Peleg, S. W. Tu, et al. Comparing Computer-Interpretable Guideline Models: A Case-Study Approach. *J Am Med Inform Assoc.*,(JAMIA), 10(1):52-68. 2002
- [4] Shahar, Y., Mirksch, S., Johnson, P, The Asgaard Project: a Task-Specific Framework for the Application and Critiquing of Time-Oriented Clinical Guidelines, *Artificial Intelligence in Medicine*, 14:29-51, 1998.
- [5] I. Sim, B. Olasov, and S. Carini, "The Trial Bank system: capturing randomized trials for evidence-based medicine", *Proceedings of the AMIA Annual Symposium*, 2003, pp. 1076.
- [6] M.G. Kahn, C.A. Broverman, N. Wu, W.J. Farnsworth, and L.Manlapaz-Espiritu, A Model-Based Method for Improving Protocol Quality, *Applied Clinical Trials*, 2002.
- [7] L. Zhang, J. Gu. Ontology based semantic mapping architecture. *Proc Fourth International Conference on Machine Learning and Cybernetics*, Guangzhou, 2005.
- [8] R. Vdovjak, G. Houben. RDF based architecture for semantic integration of heterogeneous information sources. *International Workshop on Information Integration on the Web*, 2001.