Title
Health Ontology Mapper (HOM): Semantic Interoperability at Scale

Author
Wynden, Rob

Publication Date
2013

Peer reviewed|Thesis/dissertation

The Health Ontology Mapper (HOM) Method
Semantic Interoperability at Scale

by

Rob Wynden

DISSERTATION

Submitted in partial satisfaction of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Biological and Medical Informatics

in the

GRADUATE DIVISION

of the

[redacted]

Copyright

Acknowledgements

I deeply appreciate the support of the following people and institutions.

Dr. Donna Hudson, a trusted advisor and informatics sage
Dr. David Avrin, for his unwavering support
Dr. Maurice Cohen, for his inspiration and leadership in informatics
Dr. Russ Cucina, for his support, expertise and career advice
Dr. Ida Sim, for her inspiring leadership on the Human Studies Database
Dr. Yao Sun, for his expertise and advice in refining my approach

Special thanks to the following people and institutions:
Dr. Nick Anderson
Dr. Mark G. Weiner
Marco Casale, M. S.
Davera Gabriel, R. N.
Maggie Massary, M. A.
Dr. Ketty Mobed
Prakash Lakshminarayanan, MBA.
Shannon Hastings, M. S.
Hari Rekapalli, M. S.
Aaron Abend, MBA
Dr. Nigam Shah
UCSF
UC Davis
University of Pennsylvania
Stanford University
University of Rochester
Ohio State University

The HOM Method (Health Ontology Mapper):
Semantic Interoperability at Scale

Rob Wynden

Abstract

The Health Ontology Mapper (HOM) method is a proposed solution to the semantic gap problem. The HOM Method provides the following functionality to enable the scalable deployment of informatics systems involving data from multiple health systems. The HOM method allows a relatively small population of biomedical ontology experts to describe the interpretation and analysis of biomedical information collected at thousands of hospitals via a cloud based terminology server. As such the HOM Method is focused on the scalability of the human talent required for successful informatics projects. The HOM promotes a means of converting UML based medical data into OWL format via a cloud-based method of controlling the data loading process. HOM subscribes to a means of converting data into a HIPAA Limited Data Set format to lower the risk associated with developing large virtual data repositories. HOM also provides a means of allowing access to medical data over grid computing environments by translating all information via a centralized web-based terminology server technology.

An integrated data repository (IDR) containing aggregations of clinical, biomedical, economic, administrative, and public health data is a key component of research infrastructure, quality improvement and decision support. But most available medical data is encoded using standard data warehouse architecture that employs arbitrary data encoding standards, making queries across disparate repositories difficult. In response to these shortcomings the Health Ontology Mapper (HOM) translates terminologies into formal data encoding standards without altering the underlying source data. The HOM method promotes inter-institutional data sharing and research collaboration, and will ultimately lower the barrier to developing and using an IDR.

Table of Contents

List of Figures

Introduction

To compute on clinical data from multiple clinical environments for either research or clinical purposes it is necessary to access that data using standard medical vocabularies, ontologies and constrained value sets. But there are very few (only about 300) terminologists in the USA with the skills required to guide these data translation projects and over 5000 hospitals all with unique source data requirements. Standards can help with moving data around but are unlikely to lead to meaningful semantic interoperability within the near future because clinical data is collected within commercial software environments that do not subscribe to these standards and because these source data systems are often based on UML technology that is difficult to bridge to OWL based medical data standards. Additionally much of the clinical data generated is unstructured text. These problems led to semantic gaps in our ability to properly interpret and compute on clinical data.

Problem Statement

Medical data is messy and not clearly defined. Medical data is encoded using an inconsistent mix of structured clinical data using locally defined data dictionaries, financial and claims information and unstructured full-text notes and dictations. There have been many different means described over the years of encoding medical data in standard formats but none of those methods have been adopted universally and even a consistent interpretation of those standards cannot be relied upon.

The skills required for the interpretation of medical data are difficult to obtain. Typically even the basic "coding" of financial claims data requires training and the interpretation of medical data across clinical environments requires a much higher caliber of expertise equivalent to a masters in nursing informatics or biomedical informatics. Additionally once medical data has been aggregated centrally, advanced medical knowledge is often required to formulate a meaningful query or even to interpret the results.

To make matters even worse, significant legal challenges exist to the aggregation of medical data at scale. HIPAA (Health Insurance Portability and Accountability Act) legislation has made it very difficult to obtain permission for aggregating patient data at scale. To combine all medical data at multiple institutions one must obtain permission from multiple overlapping authorities. Although these laws were intended to protect patient privacy, these complex approvals make the centralized storage and interpretation of massive amounts of medical information practically impossible.

Finally, even if all medical data could somehow be "cleansed" using available skilled labor and aggregated using legally attainable methods into a large centrally-maintained classic data-warehouse; there is no clearly defined means of storing that medical information such that it may be subsequently reused for any purpose. Researchers in different disciplines require medical knowledge to be represented in different ways. Clinical decision support information must be encoded differently depending on the medical problem

domain. The competing concerns of how financial information is used to bill insurance carriers makes the accurate "coding" of medical finance a challenge. Even the very definition of what constitutes a medical "outcome" may vary depending on the context of the question asked. All of these challenges have one thing in common: the required representation of the data is not known until it is time to use that data for a particular purpose. In other words the requirements for representing the medical data are not known when the data warehouse is first constructed.

Clinical Scenarios

Examples of semantic gaps [7,8,9] include the usage of abbreviations in medical text. For example let's say that the term "MS" is used within a clinical finding. Assuming that this is a cardiology finding we would conclude that MS stands for "mitral stenosis". If it were a finding related to Anesthesia we would assume it stands for "morphine sulfate". Etc. In fact there are hundreds if not thousands of these domain specific interpretations of clinical data that exist. Additionally multiple possible biological

pathways or forms of environmental stress may cause the exact same “clinical phenotype”. For example, an ITP patient may have low platelet counts due to Graves Disease or due to exposure to Heliobactor Pylori Bacteria and can sometimes be originally detected following abnormal serum liver tests and Grave’s patients often have liver disease. If the original clinical findings are on the topic of Grave’s disease or a bacterial infectious disease then within which domain should the term “ITP” be later interpreted? The term “ITP” does not necessarily mean the same thing within these clinical domains as it sometimes refers to “Inosine triphosphate” which is associated with gene defects leading to SAE’s after liver transplants. Also, there are questions regarding episodes of care. Patient encounters sometimes overlap and the data collected within them contributes to the computed health of the patient. But can the data collected for two overlapping clinical encounters be assumed to be related or not, and given that ambiguity what is the domain within which the data should be interpreted? The nature of clinical information is ambiguous and it can be very difficult to interpret

automatically. There are many examples of these ambiguous problems that must be addressed.

New methodologies are required that address these gaps. Approaches that rely primarily on traditional NLP or strong AI as a multiplier could have significant legal ramifications and may lack social acceptability as strong AI cannot actually explain WHY it reaches its conclusions as these systems are based on weights and graphs. Please note that there are some AI methods that have explanation algorithms although further development is needed to make them a practical reality for clinical usage. This proposal suggests an alternative method for clinical data usage based on collaborative web based access.

Research Benefit

The standardization of clinical data into common ontologies across multiple clinical and hospital systems has direct application to new research in quality improvement and decision support. Often in research the phenotype of a patient group may become very specific

which leads to small patient populations. These specific subsets of patients must be identified across multiple hospital systems in order to obtain a sufficient statistical power calculation for use in quality improvement. An example of a specific patient phenotype could be patients that have experienced myocardial infarction and have had a particular brand of stent and have been previously infected with periodontal disease via the bacteria Treponema Denticola. To mine clinical data across multiple clinics and hospitals and perform a query of this sort would require access using standardized vocabularies such as UMLS, an ontology based view of dental data such as the Ontology for Dental Research, a means of identifying the bacteria such as HumanCyc, standardized claims data using ICD-10 and financial information translated into an ontology of FDA Devices to identify the stent in use. But data is not entered into clinical information systems in the vocabularies required for this sort of query. By providing a system that can enable researchers to identify, in an ad-hoc fashion, patient populations across multiple clinical systems, we could enable the study of more specific patient phenotypes than is presently possible.

Platform for further development

The challenges described here have acted to slow the subsequent development of biomedical informatics. If the challenges of aggregating and understanding biomedical data can be overcome then population statistics can be applied at scale for applications in quality improvement research, decision support applications, financial modeling, biomedical knowledge management and outcomes based reporting. What is proposed here is a platform on which these higher goals could more feasibly be approached.

Challenges in knowledge management

This HOM (Health Ontology Mapper) Method is concerned with a formal description of metadata that describes the translation of biomedical information using only further metadata as input. This method does not require the direct access to the biomedical information in question but only access to metadata that describes the biomedical information. A means of annotating that metadata is described to define the potential usages of the clinical information

and the entire system described here is remotely managed via a web-based terminology server.

Goals of HOM

The goals of the HOM Method are:

1) Provide a means of standardizing all biomedical information using ontologies even though the data is often collected using technologies based on UML and not on OWL.
2) Provide a means by which a small number of highly trained experts may manage the translation and interpretation of biomedical information remotely through the usage of a web-based terminology server.
3) Provide a means by which biomedical data may be described and translated that does not violate the legal requirements of HIPAA and that allows the manipulation of all biomedical data as a HIPAA Limited Data Set.

4) Provide a means by which biomedical information annotated with the HOM Method may be remotely accessed using query-based grid computing platforms.

Background

There have been many approaches to resolving the semantic interoperability that exists between biomedical and biological databases. These approaches can be categorized into specific types of general approach such as the Common Data Model approach, Mediator Architectures, Federated Database Systems, Intermediate Representation, and XML Transformation.

Common Data Model Approach

In the Common Data Model [14-20] approach each site must either adopt the model as it's native representation for data, or their local data must be mapped into the common model. This is possibly the most popular approach taken and certainly the oldest. In this approach often a data warehouse is employed to house the common model. Then data is transformed and loaded into the warehouse. Each site must maintain its

own mapping of information into the common model.

One of the most difficult problems to address when using the Common Model is agreement on which models to support. Medical researchers and quality improvement staff do not frequently agree on the same set of data models. For example should the representation of drug data use RxNorm or NDF-RT, or The Enhanced Therapeutic Classification System by First Databank, or perhaps the AHFS (American Society of Health System-Pharmacists)? Depending on the clinical or research problem to be addressed one or more of these systems may be more useful than the others.

Mediator Approach

Mediator architectures [22-29] attempt to dynamically generalize or tighten each query when specific answers cannot be returned. Often mediators are delivered in classes (such as "relaxers") within specific contexts and can transfer information in object models thereby hiding the underlying database. But these systems are very difficult to deploy and counter intuitive for local hospital IT staff to maintain.

Bridge Ontologies

Intermediate Representation [30-33] systems assume that a portable "bridge" ontology can be created to carry the information without information loss from source to target. This method requires extremely experienced terminologist staff and also requires data translation from the local hospital systems into the portable format.

Federated Database Approach

Federated Data Systems [35-40] often expose a common set of interfaces without actually translating the data at rest. Instead the data is translated at point of query by the common interfaces employed. These systems require a great deal of IT system maintenance so that common interfaces are updated as hospital IT environments are updated and replaced.

XML Approach

XML (eXensible Markup Language) [41] Transformation uses the transformation language capabilities of XML (such as XSLT) to ease the translation of data to and from specific hospital environments. Data is translated into a Common Data Model or into specific Reference Ontologies

[3]. It may be possible to use this method to create "adapters" much like those used within traditional IT (Information Technology) integration systems. XML Transformation based adapters could work within very well defined source software environments and for very specific use cases. However a means of defining each XML tag must still be supplied for each hospital. In my opinion this method is essentially just another variant on the Common Data model approach but in a different format and with updated technology. With this approach, the same human resources are required at each hospital including both advanced IT programmers and biomedical terminologists.

Significance of the HOM Method

Each of the approaches discussed so far all have a single characteristic in common. Each of these approaches requires skilled staff at each hospital to perform work to resolve semantic incompatibilities. These skilled staff includes terminologists that understand the meaning, interpretation and transformation of medical data as well as highly skilled computer programmers. But highly skilled IT (Information Technology) staff are expensive and skilled biomedical terminologists are extremely rare.

For example, the Common Data Model approach requires that medical databases be first encoded to the standards ascribed to the model. If the source databases cannot be written to the common model then data-warehousing approaches that use a common data model instead force ETL (Extract, Transform and Load) programming tasks onto IT staff that often have little knowledge of medical terminology. Similarly, the Federated Database System requires local translation of the query into the local schema, the Mediator Architecture requires each local system's schema to expose a common interface, and Intermediate Representation and XML Transformation requires that information be translated at each hospital. All of these methods require both local terminologist and local programming expertise. Using these methods the skills required at each hospital to resolve semantic incompatibilities is daunting even for an advanced teaching hospital and they are completely out of reach for almost any community hospital.

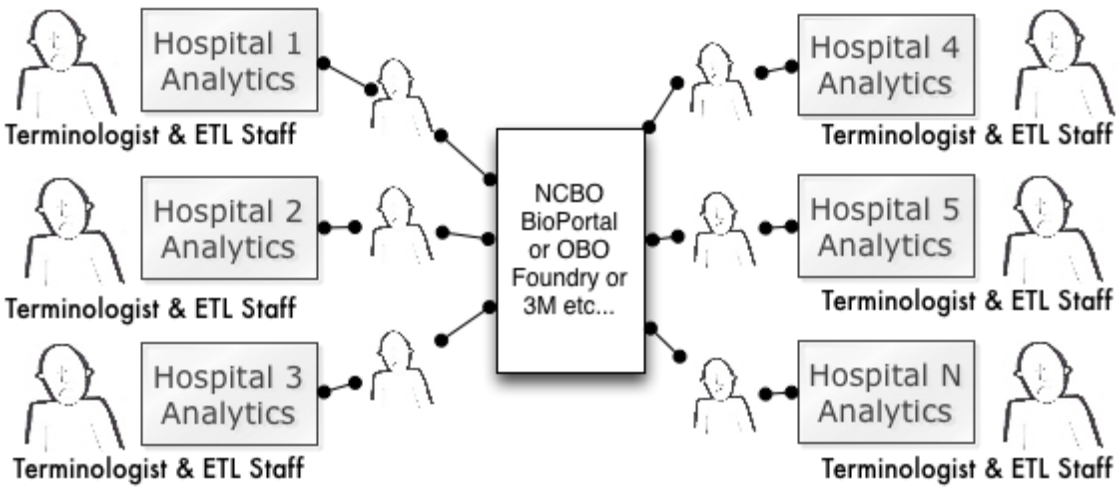


Figure 1. Standard method: The standard method of interpreting hospital data involves the engagement of skilled personnel at each hospital. These staffs are expensive and very difficult to find.

This dissertation proposes a new approach to the problem that deals primarily with the scarcity of the human terminologist resources that are required. In this approach (called The Health Ontology Mapper - HOM Method), distributed data warehouse architecture is employed. In the HOM Method a standard method of encoding and describing the translation between medical data is employed. Each hospital then uses local software that implements the HOM method identically. Each hospital interprets these terminology and ontology maps in an identical fashion that requires little local knowledge of biomedical terminology to operate. Each hospital then connects to a single, centralized terminology server for access to biomedical terminologies, relations and mappings. By accessing this common approach to terminology mapping each hospital can then respond

consistently to distributed-queries using any biomedical terminology described on the centralized terminology server.

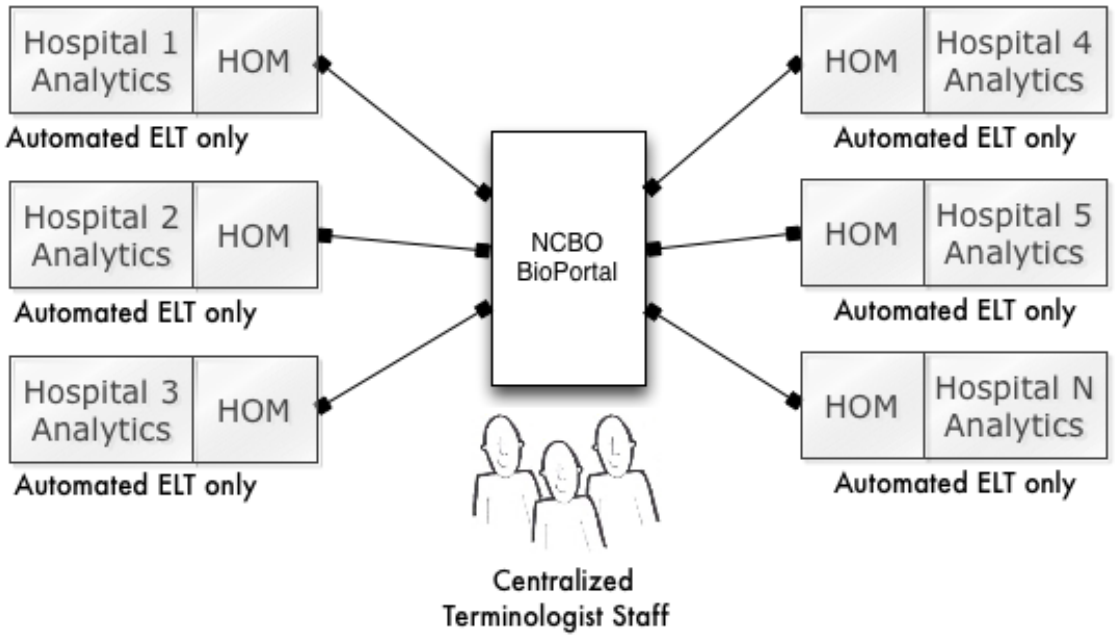


Figure 2. HOM Method: Skilled staff collaborate using a single web-based terminology server enabling remotely accessed analytics expertise for many hospitals and outpatient clinics.

By moving all information regarding the mapping of biomedical data into a single commonly accessed terminology server the staffing requirements of participating hospitals and outpatient clinics is greatly reduced. Instead the existing pool of trained ontologists would focus their combined efforts on the centralized terminology server environment allowing them to service a much larger population of hospitals in a manner that allows reuse of mapping information across multiple institutions. Each participating hospital requires little local staff to operate the system while providing a

consistent means of interpreting biomedical information.

Hypothesis

The HOM Method of employing an online collaborative approach to terminology access, interpretation and instance data mapping will provide an efficient and scalable solution to the massive aggregation of medical data, using formal terminologies and ontologies as well as instance data maps, to solve the “semantic gap” problem in a scalable manner.

The mass aggregation of medical terminologies as well as clinical instance data maps on a terminology server is both scalable and a sound means of resolving the semantic gap problem. Establishment of an online collaborative approach including common terminology and synthesis methods will permit data discovery for the development of disease models, assist medical centers in controlling costs and expand the capacity for personalized healthcare.

Contribution to Informatics

This HOM Method resolves the scaling of access to biomedical data by focusing on the limited availability of human terminologist resources within

hospital environments. By deploying identical hospital system configurations (each interpreting terminology data in an identical fashion) at each hospital, and by not requiring each hospital to employ local terminologist nor local IT ETL (Extract,Load,Transform) staff, it is feasible to build distributed access to hundreds if not thousands of hospitals, resulting in the maximum possible patient populations for study and quality improvement.

This system provides a highly efficient means of translation for all local hospital data (instance data) across the semantic gap. This system can automatically discover what data each hospital has regardless of the particular mix of local software systems and determine what maps are available based on previous works at other sites. It can then automatically map a hospital's local data based on that prior work. Any subsequent work at each new hospital can then be leveraged at all subsequent hospitals such that instance data maps of a particular type are only defined once across many hospitals. This is a web-based collaborative approach to medical instance data mapping that utilizes a central terminology server as a content management framework for the automated mapping of a large

number of client sites. This approach also requires only a single cohesive team of terminologists for the mapping of instance data at all hospitals that all reference the same set of terminology data and instance data maps.

Additional Benefits

This system can be used to magnify the effects of other informatics projects. For example, if the eMerge computable phenotypes [42] were implemented via the HOM Method then eMerge could be deployed nationally with a minimum of local site IT and terminologist involvement, thereby dramatically increasing it's scalability while lowering the cost of deployment.

HOM off-loads all terminologist involvement to a centralized web-based terminology server. This lowers the cost of ownership for each hospital. This architecture also helps local hospital IT departments that are much more capable of maintaining traditional IT systems and who lack local terminologist or ontologist expertise.

HOM System Design

Specifically, a system comprised of a single web-based collaborative terminology server platform (NCBO BioPortal [10]) for the curation, exposure, control and access of terminology along with HOM (Health Ontology Mapper) [1] provides a multiplier on the efficiency of terminologists, such that they can manually define both point-to-point and automated ontology mappings for addressing domain specific interpretations of clinical data in a scalable manner.

HOM enables the dynamic, just-in-time interpretation of warehoused clinical data.

The HOM instance data translation system was first invented within the CTSA (Clinical Translation Science Awards) program and the HOM project took 5 years to complete with test implementations that have run at several CTSA sites including but not limited to UCSF, Stanford, UW, UC Davis, Rochester U, and U. Penn. This method can be implemented on multiple software platforms and to date HOM compatible software

environments have been implemented in both java and perl.

Methods

The HOM Method leverages a single terminology server to allow multiple hospitals to translate clinical information by leveraging the same definitions of clinical terminology, the same information models representing source clinical software environments and the same set of instance maps used to translate clinical information into standard clinical terminology. When this project was first begun the terminology servers available did not provide any means of local hospital instance data mapping. Instead terminology servers (such as the NCBO Bioportal [10]) had focused on the mapping of relations between biomedical terminologies. No thought had been given to the specific hospital software environments from which instance data is extracted. This project included the novel use of Bioportal API's (Application Program Interface REST Services: REpresentation State Transform) to support instance data maps, which has allowed multiple hospitals to interrogate the BioPortal for information regarding the nature of their local hospital instance data.

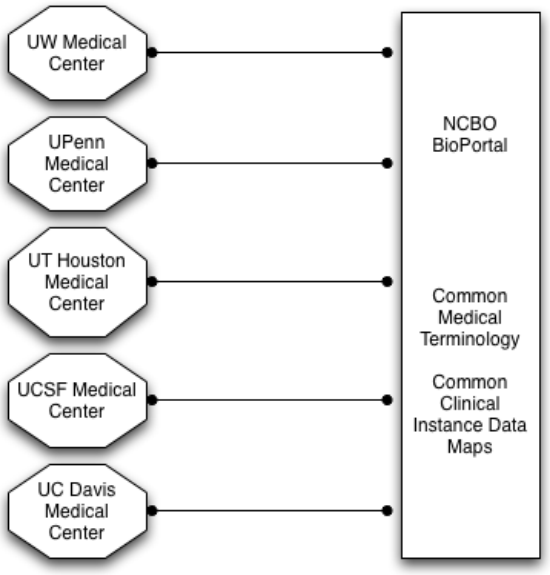


Figure 3. Multiple hospitals connected to single terminology server.

An instance of HOM installed at each hospital connects to the terminology server using an API (Application Program Interface) based on BioPortal REST services. These REST services have been extended to support HOM queries for clinical instance data maps. HOM can query these services in a dynamic fashion allowing the application of instance maps to clinical data to occur after the data has already been loaded into a warehouse.

HOM Method Tags

To achieve this common representation of hospital instance map definition HOM implements a series of terminology server "tags" that annotate source information in a standardized fashion across multiple hospitals.

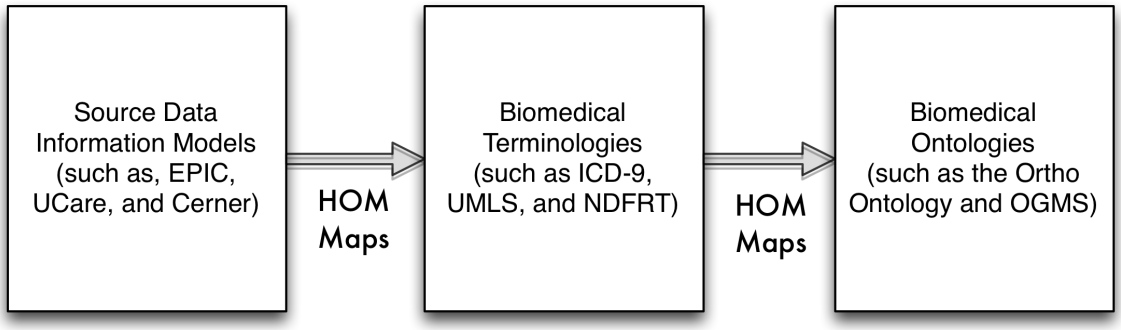


Figure 4. HOM Maps represent a means of repurposing clinical content for re-use in a more standardized manner.

HOM Tags are applied to elements in all three types of content housed within the terminology server. Those include source information model content that describes source hospital information systems (for vendor systems such as EPIC or Cerner), Biomedical terminologies/taxonomies (such as ICD-9 or ICD-10), and Biomedical ontologies (such as ICD-11 or OGMS) that attempt to model the relationships between biomedical terms. HOM Tags are also used to define the mappings between these 3 types of terminology server content.

In practice, when used with BioPortal, HOM Tags can either be created within Protégé [2,4] or CMap Tools or they can be dynamically created and edited on BioPortal using Web-based Protégé. However, in principle, the HOM Tags described in this method could be created within any ontology

tools and could be used with any terminology server environment.

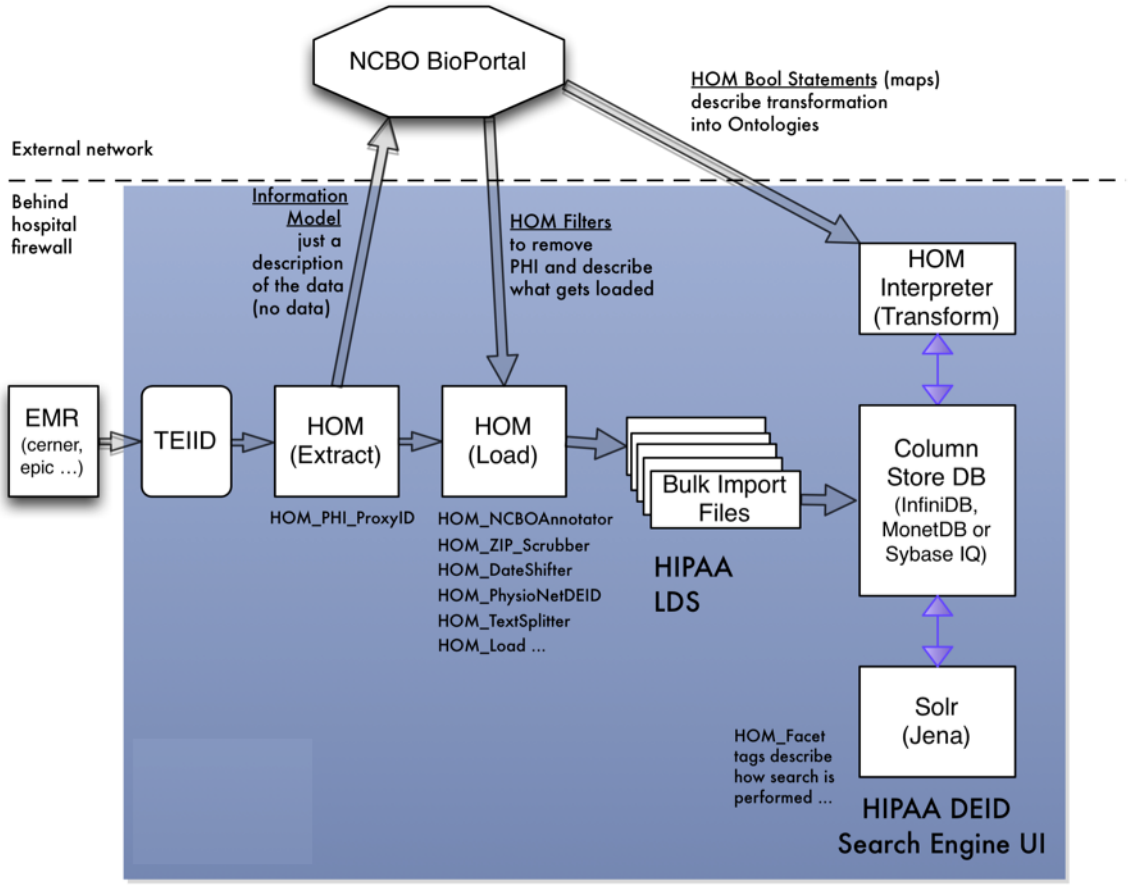


Figure 5. Shows the HOM Method tags and how they are used to describe the standard processing of medical data via a terminology server. This system is to be implemented on a commercial Appliance and used to measure the performance of the HOM Method in a chart review study that is described below.

HOM Information Models

The first step in the processing of HOM Method tags starts with the generation of OWL Information Model files from the source database. To accomplish this goal we first use a database virtualization system to hide the vendor database platform (such as Microsoft SQL or Oracle) and

thereby eliminate any vendor specific coding or data transformation. The virtualization system chosen for that task is named TEIID. The TEIID system provides a generic JDBC interface that looks exactly the same regardless of the source database platform. Next, a derivative of the NCBO Datamaster system is called (named GenerateRDBInformationModel) which creates an OWL Information Model file that contains no instance data but that represents all constructs from the source database schema in OWL format [46,47]. This allows the OWL representation of the source schema to contain all table, column and foreign key relationships as well as the data types of the individual columns. The information model is then extended so that all database columns are represented as an OWL Class instead of just using a meta-class instance. Representing database columns as an OWL Class was necessary to allow HOM to leverage the existing BioPortal "Term Maps" as a means of implementing HOM 1-to-1 instance maps.

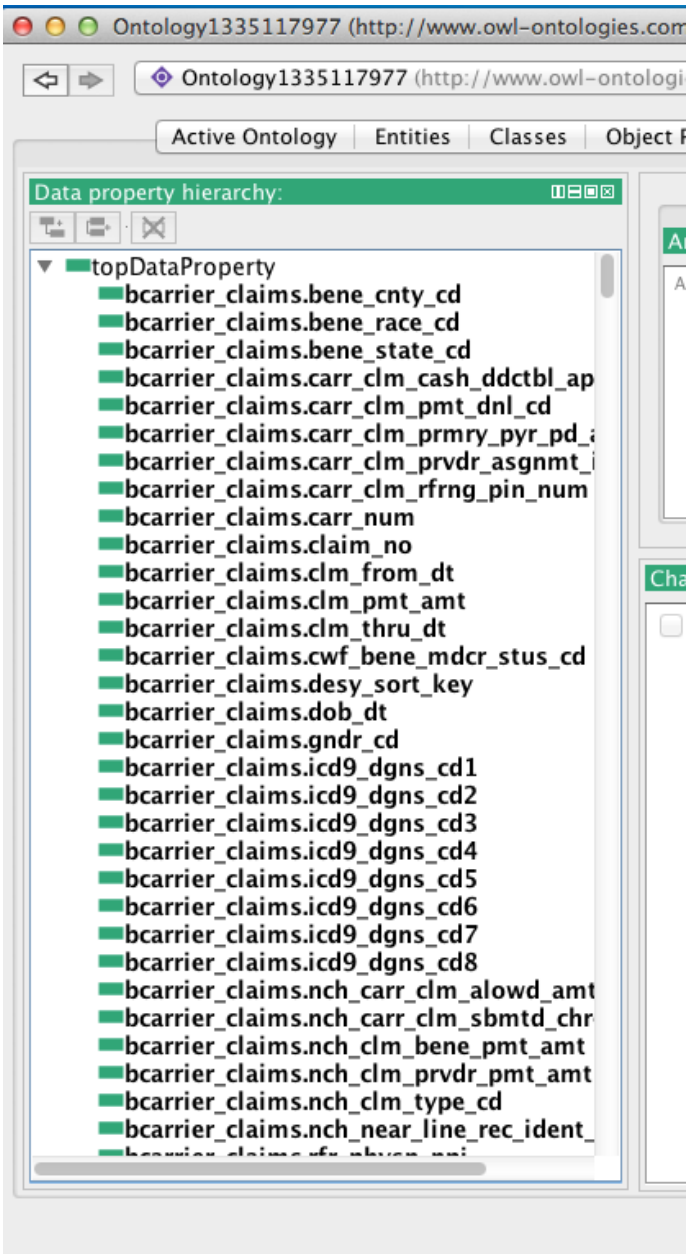


Figure 6. A Source Information Model of the CMS data used for the bundled payments initiative.

Once the information model has been generated it is then automatically posted to BioPortal via BioPortal REST services and an email list of ontologists is alerted to inform them that new content is now available. Centralized ontologist staff, that work via the NCBO BioPortal instead of at

the local hospital, then apply other HOM tags to the information model to describe how instance data mapping should be properly processed. Please note that this is essentially a web-based content management approach to describing the instance data mapping between biomedical and biological ontology terms. Then, the instance data processing is performed automatically at each local hospital site. The following diagram provides an outline of the basic process of generating OWL information models in the HOM Method.

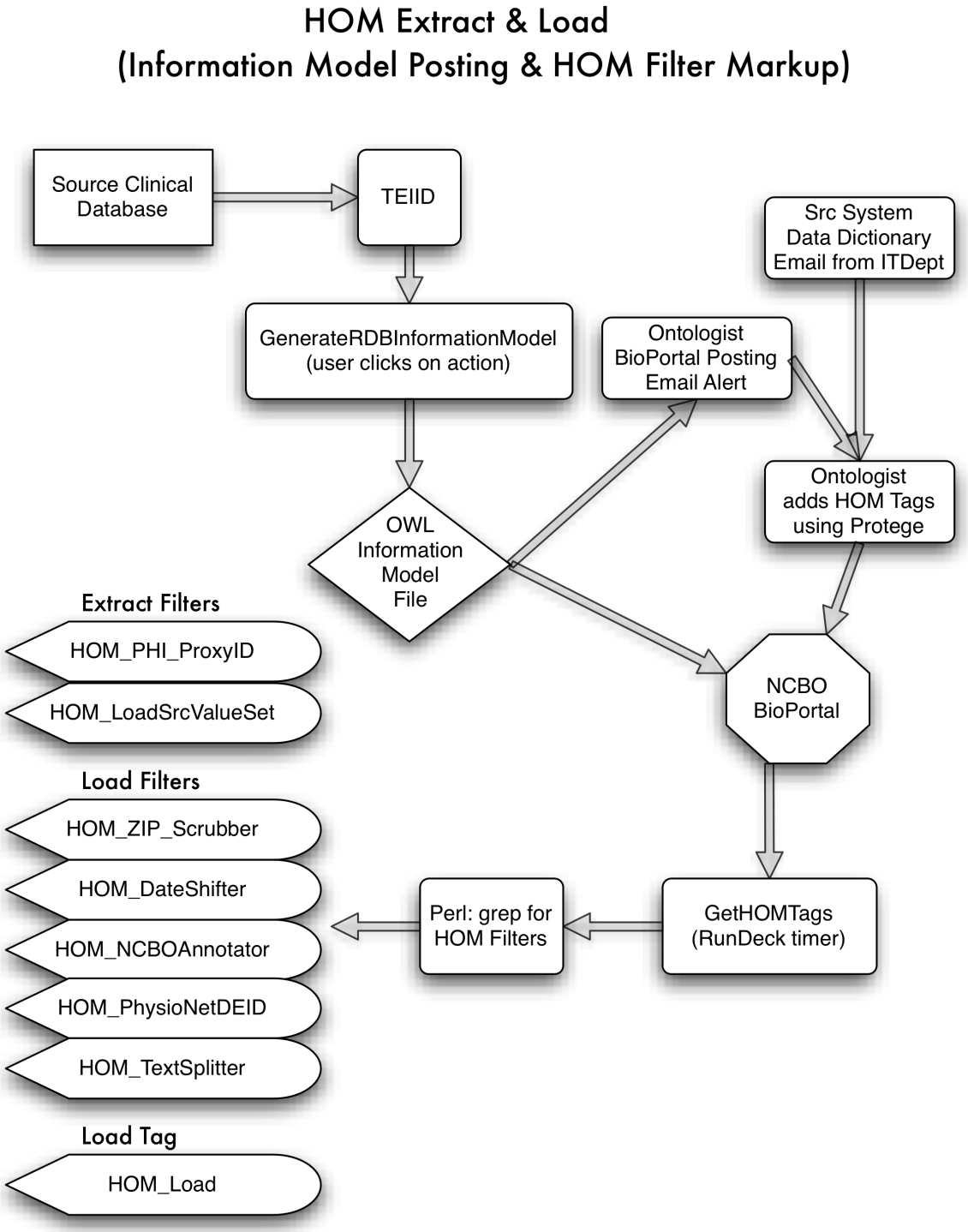


Figure 7. HOM Annotation tags delivered from the NCBO BioPortal.

Once the information model has been posted on BioPortal ontologist staff are then free to apply HOM Extract, Filter and Load tags to describe how it

can be processed. Since the information model is generic these HOM tags are re-usable across multiple institutions. For example, a set of HOM tags that describe how Cerner EMR data is processed would be greatly beneficial to any hospital that is using Cerner as their EMR platform and it is expected that the vast majority of Cerner HOM tags are reusable by any hospital that uses Cerner.

HOM_PHI_ProxyID

HOM Extract, Filter and Load tags are applied to the source information model as a means of describing how data should be identified and loaded. The first tag typically used is the HOM_PHI_ProxyID tag. HOM_PHI_ProxyID replaces any incoming data within the specified column with a random number. If the same source information is later encountered then the same random number is returned. This allows patient sensitive identifiers to be eliminated from the source data without losing the ability to use that data for linking other data elements. For example, this tag could be used to remove the patient's social security number from the source EMR database during data extraction.

Replace a string value from any source & type with a random unsigned 64 bit integer proxy value. If the same value is encountered then the same proxy is returned.

The primary purpose of this tag is to populate the ProxyTable.

Note: the location of ProxyTable must be system configurable. It's possible that the hospital may decide to house ProxyTable outside of the Appliance in some cases. So the admin interface needs to allow the MedCtr to provide credentials for an external ProxyTable and they will need a copy of the DDL to create the table in our documentation.

Defaults: If no parameters are supplied then "Source" is set to the name of the ontology. If no Type is supplied then please set that to "PHIProxyID".

Example usage: HOM_PHI_ProxyID EPIC MRN

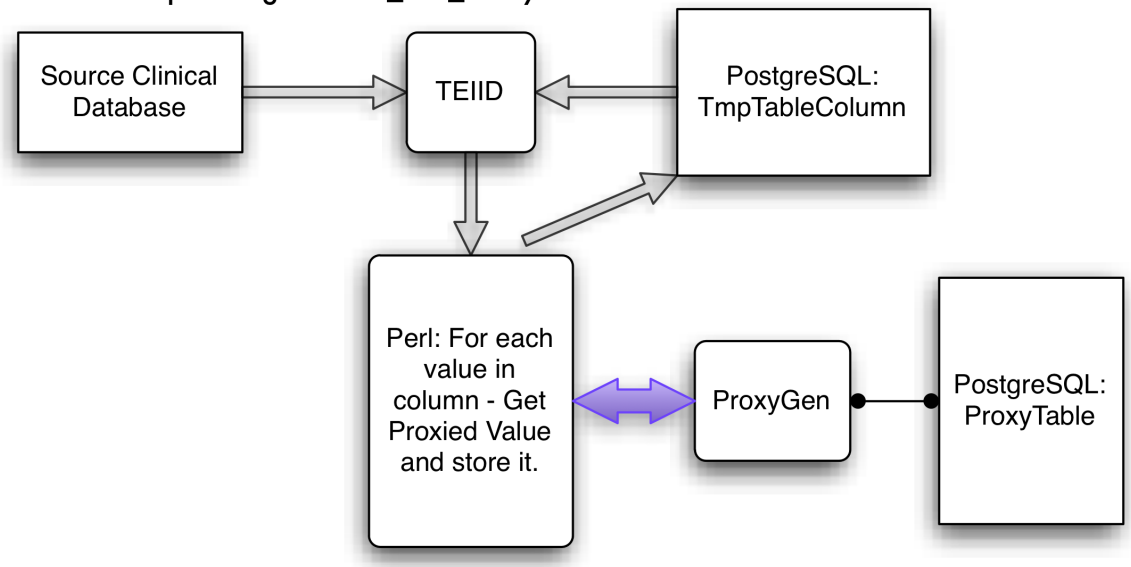


Figure 8. Creating proxies for sensitive patient data.

The ontologist applies the HOM_PHI_ProxyID tag to any information model data that contains HIPAA sensitive patient identifiers such as Name, MRN or SSN. Please note that if that mapping of proxy identifiers is retained,

within a separate and secure sub-network, then the re-identification of data remains possible even though the dataset in use remains a HIPAA Limited Data Set (HIPAA LDS).

The HOM Method also employs a second tag associated with the data extraction process from the source clinical database named the HOM_LoadSrcValueSet tag. HOM_LoadSrcValueSet will cause the loading of all unique instance data values for the specified column. It does this by performing a "select distinct" against the specified column and uploading the resulting set. This is useful for the assignment of simple 1-to-1 maps. In the HOM Method 1-to-1 maps are normally used to map patient demographic information out of the source information model. For example, the source information model may represent the patient's sex as 0, 1, and 4 and those values may need to be mapped to a medical terminology that represents sex as "m", "f" or "o".

Creates an new (derived) OWL file that includes a value-set list and posts it to BioPortal and sends an alert to the ontologist. This tag is optionally added to source information model columns by the ontologist to create column specific value sets.

Please note: this new OWL posting does not have any dependencies on other calls to GenerateRDBInformationModel. No race-conditions will occur if you allow processes to run in parallel. This tag will set the HOM_ParentOntology attribute in the new OWL file to the ontology ID of the information model within which the tag was placed.

Defaults: There are no defaults because there are no parameters
Example usage: HOM_LoadSrcValueSet

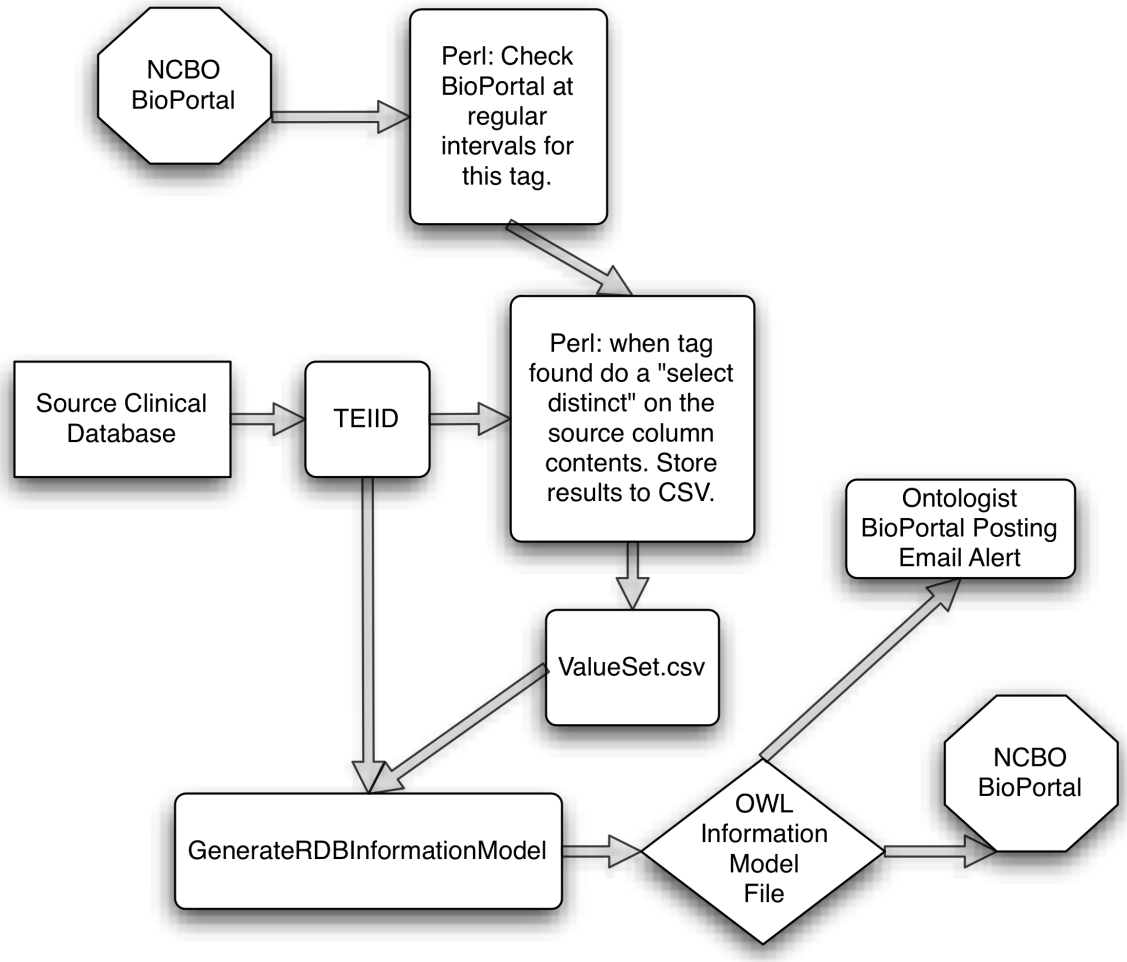


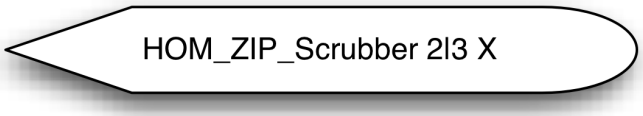
Figure 9. Using HOM to generate information models and value sets.

Once the new OWL file is generated it's HOM_ParentOntology tag is set to

the URI of the original OWL information model. The HOM_ParentOntology tag implements inheritance for instance data mapping and will be discussed below.

HOM Filter Tags

The HOM Method also defines 5 Load Filter tags that further assist in the handling of sensitive patient information. The purpose of the HOM Filter tags is to produce data for loading into the data warehouse that is a HIPAA Limited Dataset. By enabling the warehousing of data as a Limited Dataset the potential legal liability of institutions using the system is greatly reduced. This architecture makes it possible to store information in a warehouse with very high security and relatively low risk of institutional exposure to data theft [45]. The first of those tags, the HOM_ZIP_Scrubber tag, describes how zip codes can be modified for loading into the warehouse.



Remove the last 2 or 3 digits from a ZipCode and replace them with the assigned character and then store the result for later.

Defaults: If no parameters supplied then defaults to 2 digits when the patient population as published by the US Census is above 20,000, otherwise it will remove 3 digits. The default "strikeout" character is "X".

Example usage: HOM_ZIP_Scrubber 2 X

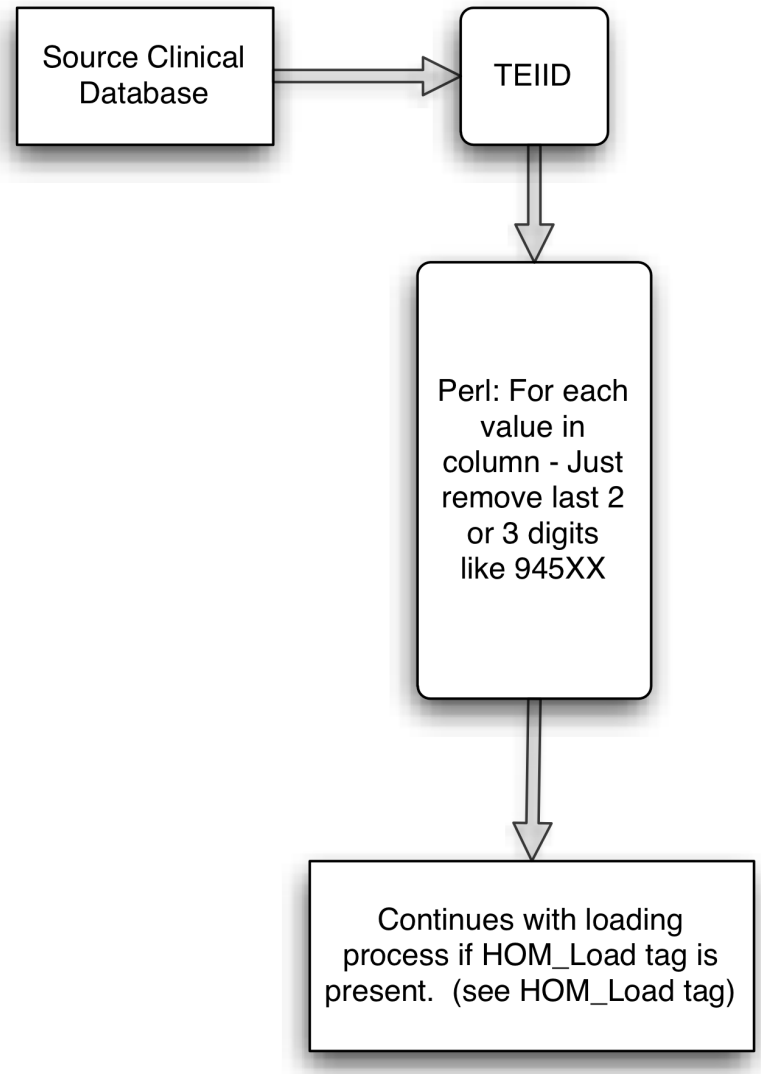


Figure 10. HOM_ZIP_Scrubber for handling zip codes.

HOM_ZIP_Scrubber removes the bottom 2 or 3 digits of a zip code as appropriate for the geographic location of the patient.

The next tag is the HOM_Dateshifter tag that shifts patient related dates forward or backward by random amounts to further de-identify the patient record.

Shift a date either forward or back a random amount within the specified range and then store the result for later. So this can be used to date shift each patient encounter uniquely within the range provided. Date shifts should never cross year-quarterly boundaries so that "seasons" are still valid.

Note: this tags especially this tag can appear multiple times for the same data allowing multiple interpretations for different use cases. DOB may need age-at-encounter calculations or current-age-of-cohort selection (or likely both). A date-shifted DOB should never cross a year-birthday boundary in any event.

Note: HIPAA LDS does allow for limited dates-of-service. (We should just do what we can.)

Defaults: If no parameters are supplied then default range to 5 days.

Example usage: HOM_DateShifter 3
(shift the dates randomly either back or forward by 3 days.

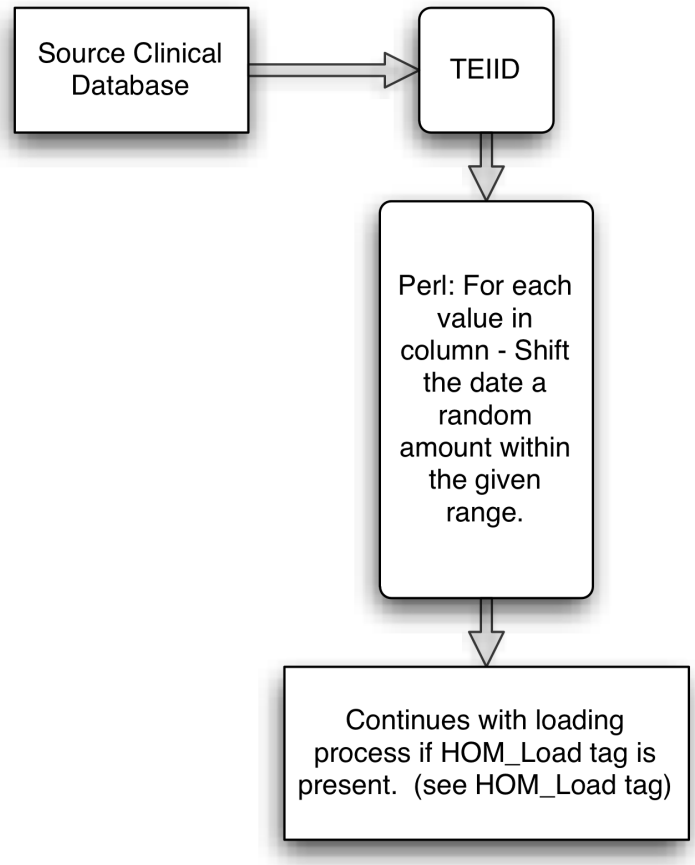


Figure 11. HOM_DateShifter for obscuring sensitive dates.

The HOM_Dateshifter tag does not cross year quarterly boundaries to keep the basic “season” intact. Also age at encounter may need to be calculated shortly after the loading process is completed.

Text Processing

The HOM_NCBOAnnotator call transforms full-text clinical notes into biomedical terms. (Alternatively this feature could be implemented by using the Apache cTakes system). Since the terms selected must already exist on BioPortal, and since BioPortal does not contain PHI, this also results in the de-identification of the full-text.

HOM_NCBOAnnotator TerminologyList (ontology ID's/accession#,cuiPropName)

Mark up the given field with medical terms from the list of terminologies provided.
Parameters: list of pairs (OntID,cuiPropName), (OntID,cuiPropName), (Acc#,cuiPropName)...
(no default parameters are allowed for this tag)
(IDs are either ontology ID's on latest version or accession numbers, and name of cuiProperty)
Example usage: HOM_NCBOAnnotator 45231, "CUI"

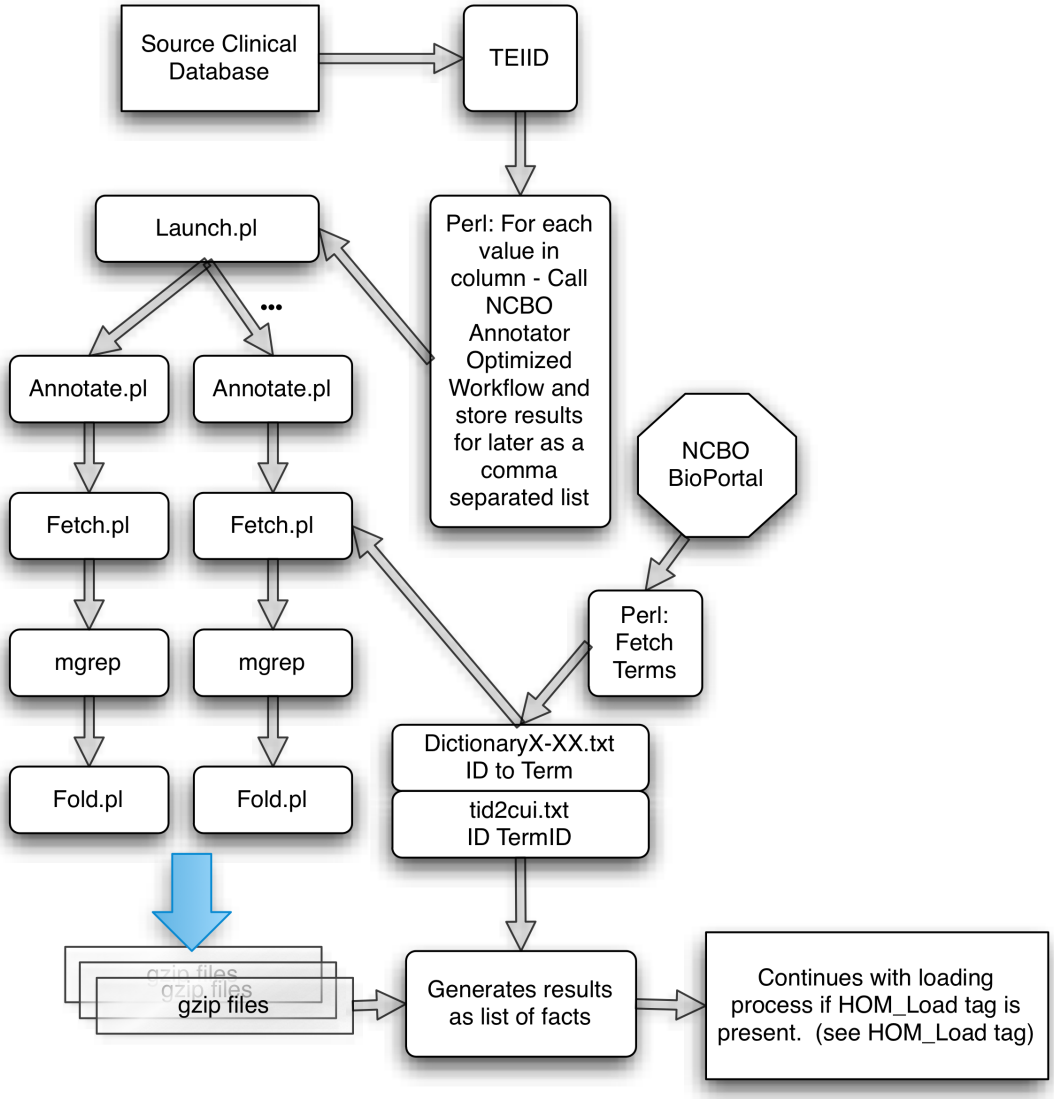


Figure 12. The annotation of unstructured text.

The means by which Annotator [12,13] is used within HOM is completely medical terminology neutral. Unlike the default Annotator [12,13]

implementation, when used with this HOM tag, Annotator can be assigned any biomedical terminology through the application of HOM annotations (HOM tags) on the source information model. For example, UMLS contains a limited set of medical abbreviations. To better implement instance maps on BioPortal the HOM team has also created our own biomedical abbreviations ontology called ABBS. The same biomedical text can be interpreted via this tag twice, once via UMLS, and a second time using ABBS through the application of 2 HOM tags. This allows the ontologist to rapidly identify and adjust for new biomedical abbreviations without the need to wait for updates to UMLS and without writing custom code for each hospital.

The HOM method also provides a means of delivering clinical text for human consumption with the HOM_PhysioNetDEID tag. That can be useful when medical personnel wish to review the results of HOM maps so that they might suggest improvements. This is also useful for driving retrospective data driven chart reviews. Such chart reviews are to measure the output of maps generated with the HOM method.

HOM_PhysioNetDEID OutputName, Type,Type,Type (list of Types)

For the give list of PHI Types (in ProxyTable) get all PHI strings and remove them from the input text. The Types are all PHI Types that have previously been added to the ProxyTable.

The first parameter is the new name of the resulting deidentified text CLOB.

Defaults: If no parameters are supplied then set type to "PHIProxyID"

Example usage: HOM_PhysioNetDEID "PHIProxyID"

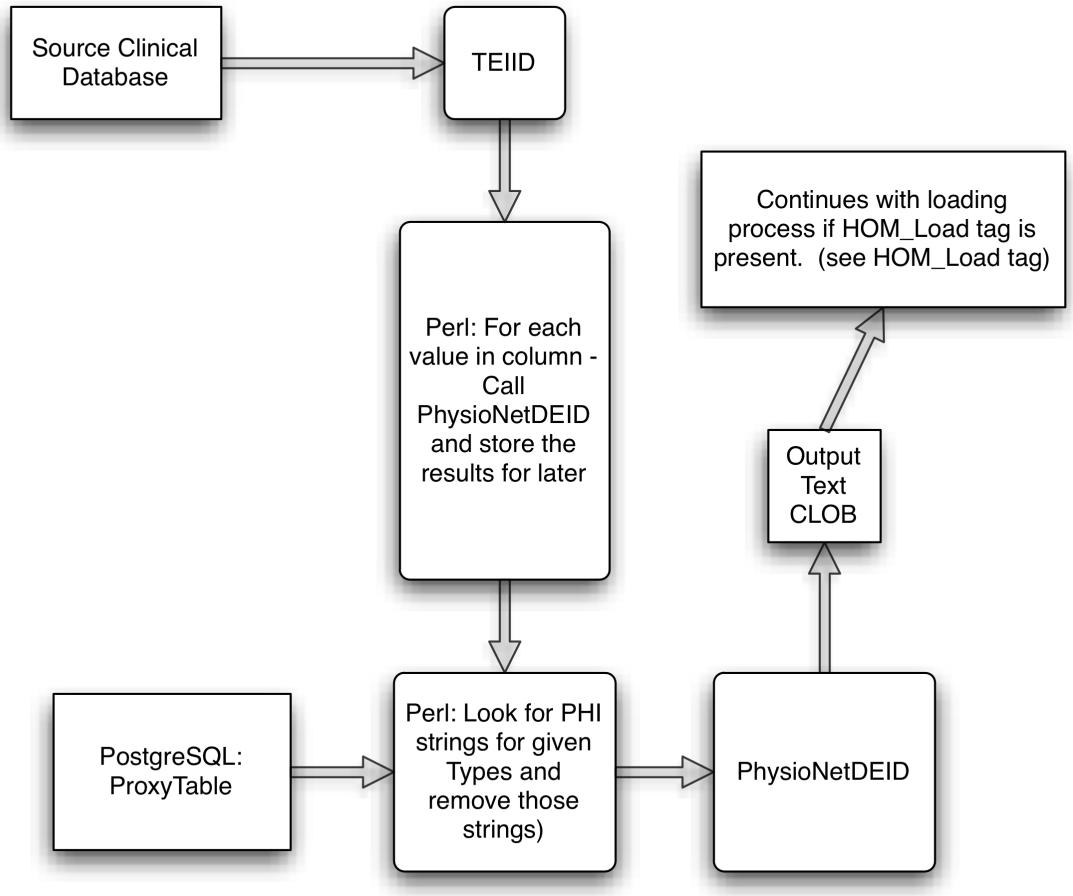


Figure 13. The DEID of text meant for chart review.

PhysioNet DEID can be used to scrub PHI from clinical notes and other

unstructured data. It does this by looking for PHI that has previously been identified within the source EMR system. Since the HOM_PHI_ProxyID tag generates a ProxyTable with that same information it can subsequently be used to drive PhysioNet DEID and de-identify unstructured clinical text. When applied to the source information model by ontologists this tag directs the removal of PHI from clinical notes to make them accessible to researchers and quality improvement staff that need to efficiently evaluate the output of HOM or to enable staff to perform other chart review related tasks.

HOM also provides a means of treating "semi-structured" clinical notes. Some EMR (such as EPIC for example) clinical notes have internal structures that are denoted using annotations, units of measure and xml tags. These semi-structured notes can be used to break up large text fields into smaller fragments such as "patient history" or "admit finding". By breaking up these larger text fields in a consistent fashion they can later be handled in a more precise fashion via the usage of subsequent HOM tags. In the HOM Method that is accomplished by using the HOM_TextSplitter tag.

Break the input text field up into smaller subfields and store them. The name of the resulting elements are the same as the source except they have the corresponding "OutputName" string appended to them.

Note: HOM_Filter tags can appear multiple times for each leaf node. For example, text splitting could be followed by physionetDEID and Annotator filters for specific subsections of the text. Or the same text may be broken up multiple times into different subsets. Tags execute in the order given.

Defaults: no defaults are allowed with this tag.
Example usage: HOM_TextSplitter "^---Next_Part" RadiologyFinding

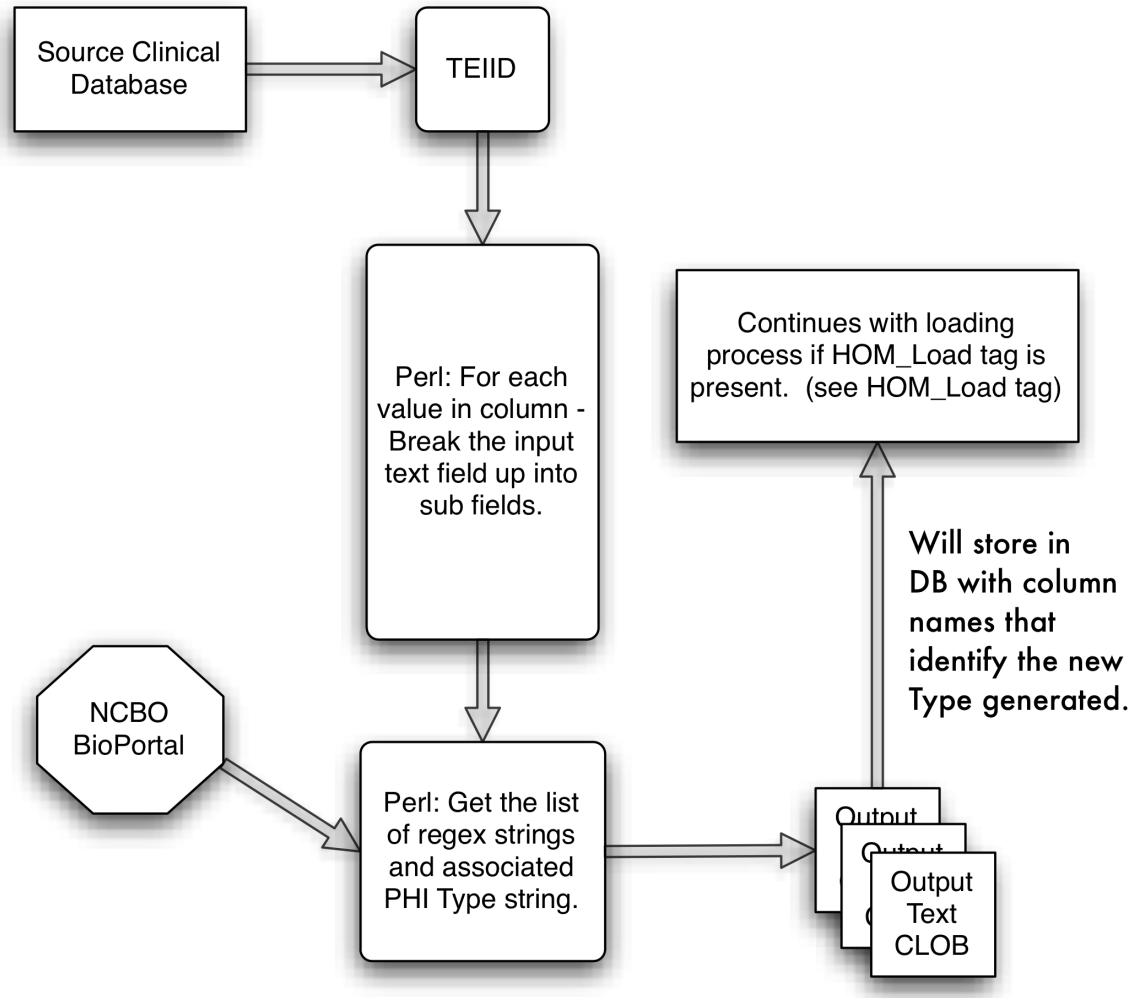


Figure 14. Splitting up text fields before further processing.

HOM Filter tags are used to describe the proper handling of HIPAA sensitive information. In addition to HOM Extract and Load Filter tags there is also a HOM_Load tag that describes how information should be organized before loading into any data warehouse system.

HOM_Load sourceSystem bulkImportDelimeter

Generate the bulk import files used to load data to the column store required for transformation. This tag is applied to the SOURCE Information Model columns that are to be imported and effectively "activates" a source data element for subsequent loading via HOM. The sourceSystem parameter is set to the name of the source data system and should be coded consistent with the Source parameter passed to the HOM_PHI_ProxyID tag. When given the delimiter character to be used for the bulk import files this will generate all fact table entries for the column store. To improve efficiency and to enable restarting jobs each bulk file has a max of 1 million records. After 1 million records another bulk import file is created automatically with file names that denote the source data imported followed by an auto increment number.

An MD5 hash of the entire input row is generated and used to populate the encounter_num, patient_num, provider_id, modifier_cd and instance_num fields. The source URI is used to populate the URI field, start_date, update, import and download dates are all set to current date, concept_cd is set to BioPortal ontology accession number+"_"+(term ID), valType_cd is set to "T" and all data is loaded as text into tval_char. Data that results from PhysioNetDEID is placed in the observation_blob field. The sourcesystem_cd field is set via it's parameter.

Defaults: parameters default to: EPIC ','
Example: HOM_Load EPIC '^'

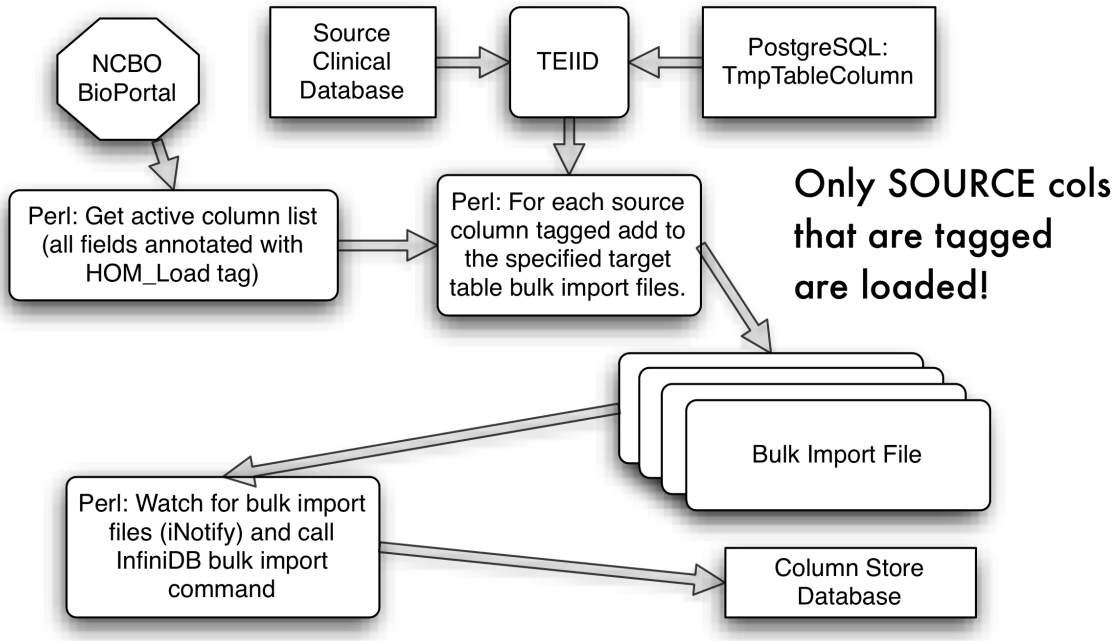


Figure 15. Bulk import file generation.

The HOM_Load tag is a database bulk import file generation tag that is

completely database vendor neutral. As such it will generate bulk load files for any database vendor and can be used to generate fact table entries for any data warehouse platform.

Once data is loaded into the data warehouse information is represented in a fashion that can be further manipulated with HOM maps. This allows the automated creation of any required lookup tables (sometimes referred to as dimension tables) that are required by the schema of the data warehouse. For example, when loading to i2b2 [6], this system supports the automated generation of the Patient Dimension table from the loaded facts. Prefabricated dimension table generation scripts are supplied that will generate i2b2 dimension table records.

In normal ELT (and ETL) data warehouse processing dimension tables are generated during the data loading process as records are generated thereby maintaining the referential integrity of the dimension tables on a continuous basis. But HOM provides an innovation that make it possible to forego the generation of dimension table entries during the loading process. In HOM all facts are represented by URI's that are defined on a

terminology server. Multiple URI's can be assigned to the same fact table entries multiple times allowing facts to become members of infinite ad-hoc sets. These sets can be defined to include the contents of the dimension tables and therefore dimension table data can be regenerated from the fact table entries after the loading process is complete. In a HOM based data warehouse referential integrity of the data warehouse schema is deactivated during bulk loading and a batch process can then regenerate dimension table entries regardless of the schema employed by the data warehouse vendor. This allows HOM to define a single bulk import file format that can be used to load data into any data warehouse system (including systems like i2b2, Cognos and Pentaho).

This use of URI based fact table loading to enable the generic generation of bulk load files that are compatible with any database vendor (such as MonetDB or Sybase IQ) and which are also compatible with any data warehouse schema (such as i2b2 and Business Objects) is novel.

After data has been Extracted and Loaded (EL) it must next be Transformed (ELT). The HOM interpreter provides the methods for data transformation.

HOM_Bool

HOM Translate : Translate the column store data into multiple ontologies

The HOM Interpreter generates new column store facts based on previously stored facts. Input data can be presented from source information models, biomedical terminologies or ontologies or any combination of the three.

The HOM Interpreter generates new facts by evaluating either BioPortal "Term Mappings" (which are always 1-1 maps from source to target; or the interpreter can generate facts via the evaluation of HOM_Bool statements which are multivariate many-to-1 maps that are tagged on the target ontology. By evaluating sets of either type of mapping, HOM can translate data in complex many-to-many mapping sets. For example, HOM can be used to map into ICD-10 by evaluating maps from SNOMED, ICD-9, Demographics, LOINC and RxNorm.

Note: The HOM_Interpreter will use HOM_Bool statements within HOM_ParentOntology's as well (inheritance) unless those statements are overridden in the child ontology.

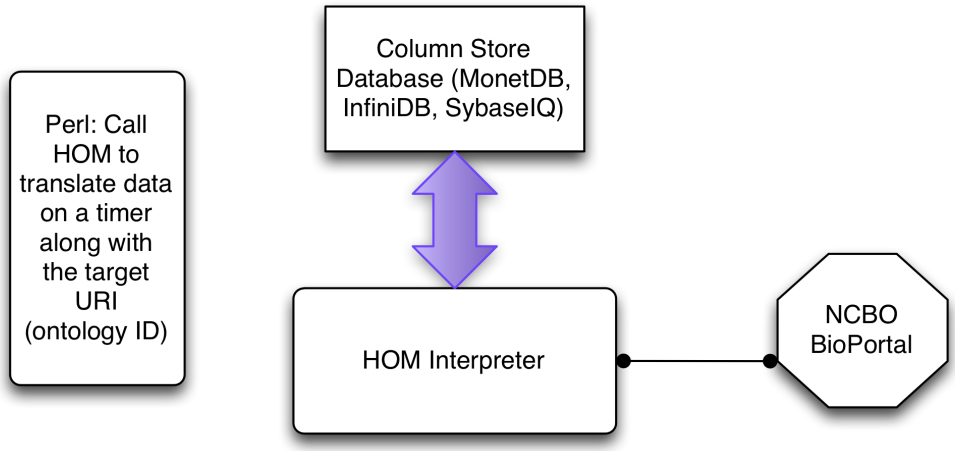
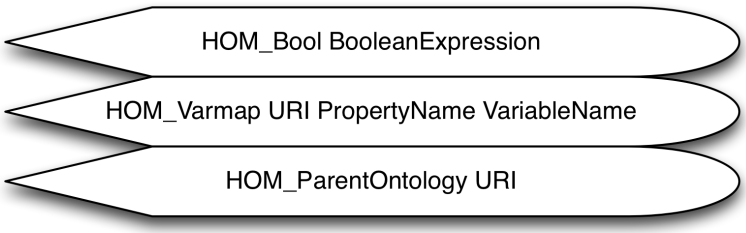


Figure 16. The execution of HOM_Bool statements.

In addition to supporting simple 1-to-1 BioPortal “Term Mappings” the

HOM Method also defines three mapping tags. HOM Map tags are applied

to the Target ontology and are used to define the conditions under which target ontology terms are generated as new facts within the data warehouse.



HOM Bool expressions evaluate any SQLite Expression and when that expression evaluates to TRUE the resulting target ontology term is created and instance data is generated for that target term. HOM_Bool statements are multivariate and can contain expression terms from multiple source information models and terminologies. For example, a HOM_Bool statement could map into ICD-10 from a multivariate boolean statement that uses ICD-9, SNOMED, Demographics, LOINC and RxNorm as inputs. HOM_Bool statements use 1 or more HOM_Variable statements as a means of expressing shorthand notation to make the boolean statements easier to read and curate. *These HOM tags can either be entered into BioPortal as properties via Protege or alternatively they can be dynamically entered using Web-base Protege.*

Note: The HOM_Interpreter will use Bool statements within HOM_ParentOntology's as well (inheritance) unless those statements are overridden in the child ontology. When HOM_ParentOntology is used that node and all of its children are inherited into the child node. Child taxonomy paths that are identical to the parent override existing behavior.

Note: HOM_Bool statements use the SQLite expression syntax that is extended to include access (as SQLite Functions) to the jFuzzyLogic, R and Weka, and RxNav API's.

Example usage:
HOM_Bool1 U.C0195620 OR (U.C2451470 AND U.C0185115)
HOM_Bool2 U.C0185115 OR U.C0391874
HOM_Varmap http://purl.bioontology.org/ontology/SNOMEDCT/ 'Umls Cui' U [accession]
HOM_ParentOntology http://purl.bioontology.org/ontology/ICD10PCS

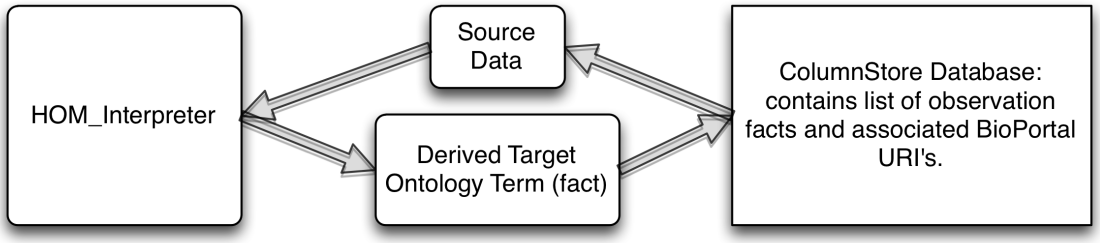


Figure 17. HOM_Bool statement, inheritance and URI based fact IDs.

HOM_Bool statements provide a multivariate mapping definition. The HOM Interpreter calls the terminology server (the NCBO BioPortal over REST services) and determines what mapping rules are available for use within the data warehouse. It then reads in the raw clinical data and previously mapped medical terminology as input and writes back derived fact table entries. When a HOM_Bool expression evaluates to "TRUE" that is a signal to the data warehouse that the specified target ontology term should be added to the data warehouse as an additional derived fact. HOM_Bool statements are complex and often include variables from multiple source information models and terminologies.

The HOM_Bool statements are an implementation of the SQLite Expression Syntax. SQLite is used as the standard expression syntax in many (100+) open source projects such as MySQL; and SQLite is a notation system that is understandable [21] by most people. Instead of referencing database columns however, the HOM implementation of SQLite Expression Syntax references terminology server URI's allowing this expression syntax to be used as a mapping descriptor. Additionally the SQLite function interface can be used to call external resources such as R and Weka for statistics and

machine learning and to allow calls to the RxNav server for handling RxNorm data transformations. In the future, this capability could also be extended on the function call level to access a fuzzy logic library.

Since terminology server OWL information is referenced via URIs (using purlz) a shorthand notation for URI's was provide with the HOM_Varmap statement. The HOM_Varmap statement allows any URI property name to be associated with an alphabetic letter. For example, the URI for UMLS and its "CUI" property could be identified by the letter "U" within the HOM_Bool statement. This shorthand notation makes HOM_Bool statements shorter and easier to read.

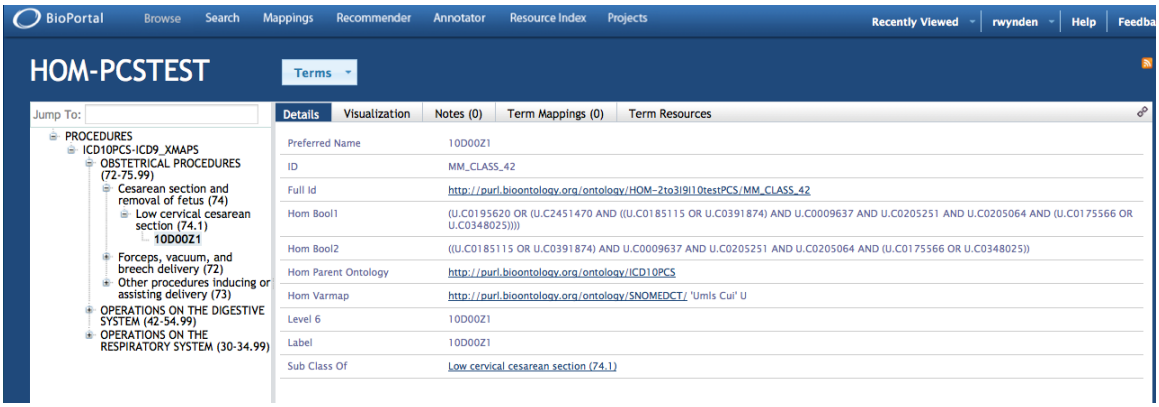


Figure 18. Example of a HOM_Bool and HOM_Varmap on the NCBO BioPortal.

HOM_ParentOntology

The HOM Method also provides a means of using inheritance within information models, biomedical terminologies and ontologies. The HOM_ParentOntology tag tells all HOM handler functions to traverse the parent ontology, via graph traversal [11], and process all of its HOM tags as well as the OWL information within which the HOM_ParentOntology tag is placed. The goal of HOM is to enable the radical efficiency of ontologists such that the existing population of medical terminology experts could feasibly describe the data transformation and analysis of data for all 5000 hospitals in the USA. Sharing OWL information for information models, terminologies and ontologies across multiple institutions, can facilitate that goal. For example, if two hospitals are both using the EPIC EMR platform, by using the HOM_ParentOntology tag, both institutions can share a common root definition for how data is extracted, loaded and transformed from EPIC. Similarly if information model instance data is first translated into biomedical terminologies and taxonomies and then subsequently mapped into ontologies then those ontology maps would be generic and could be used for any host EMR platform in a consistent manner.

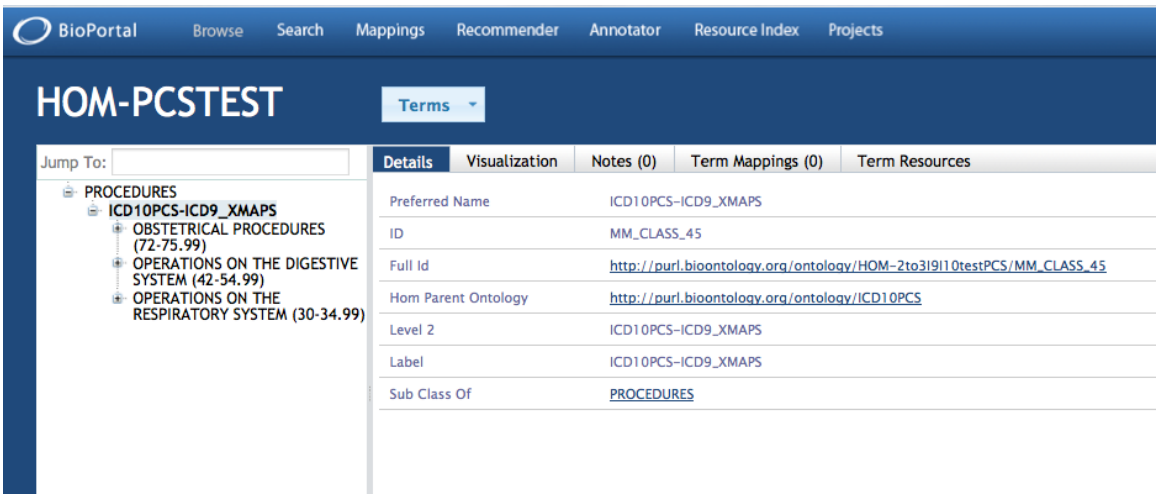


Figure 19. Example of HOM_ParentOntology on the NCBO Bioportal.

The HOM_Bool and HOM_ParentOntology tags can essentially make the HOM Method serve as an alternate form of knowledge-based system architecture [5]. In the HOM Method the knowledge-based system is distributed and uses OWL to capture rules. The HOM Method also includes mechanisms for semantic interoperability and data integration. Since all knowledge related to interoperability is contained within the terminology server environment, and not housed within the hospital, the resulting client warehouse systems do not require local hospital expertise to encode the rules.

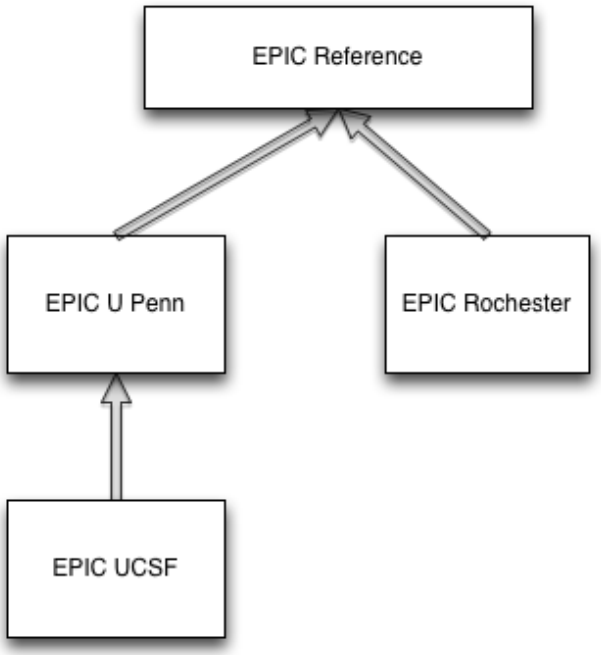


Figure 20. HOM allows terminologists to Inherit mapping content from previous ontology map sets. HOM maps can be over-ridden in derived ontologies to allow the efficient sharing of mapping content.

The HOM_ParentOntology tag describes inheritance of terminology server mappings. This allows terminologists to efficiently leverage prior art so that instance data mappings are re-used across multiple hospitals. The HOM Method describes a particular manner in which inheritance is implemented. In the HOM Method instance map inheritance must address 4 issues related to graph traversal on a terminology server: 1) instance map subset definition, 2) order of execution, 3) instance map overriding and 4) cyclic execution checks.

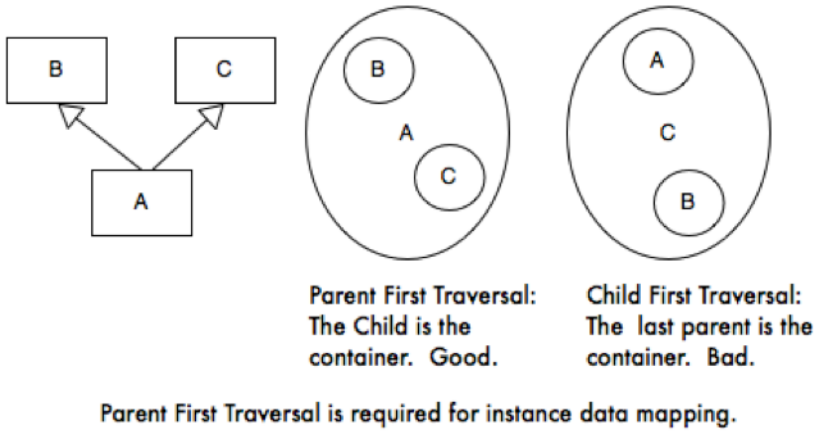


Figure 21. Instance Map Subsets: The entire parent must be traversed first before the child.

When inheriting instance map annotations from a parent ontology it is important to traverse parent content in a particular order so that maps are contained with the proper subsets. In the HOM Method any identified parent ontology must be traversed, and all of its HOM annotations must be processed prior the traversal of the child ontology.

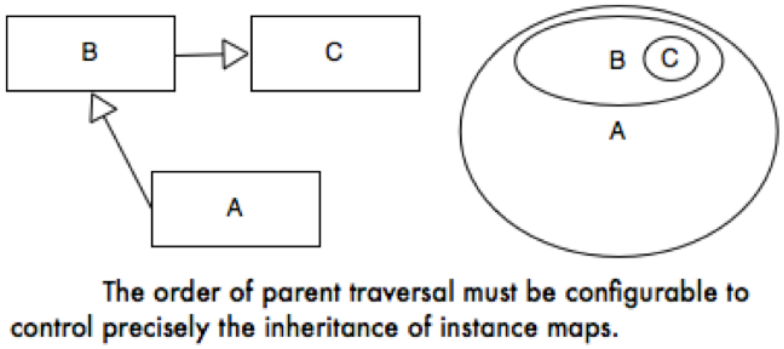
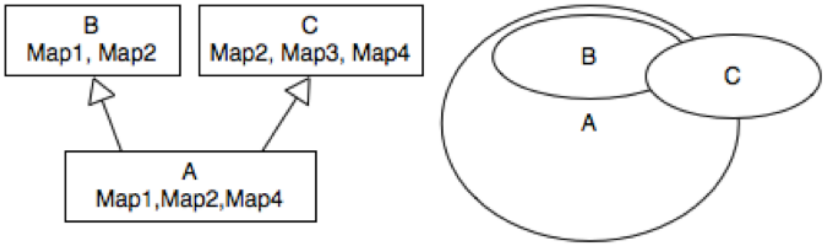


Figure 22. Order of execution: Allow terminologists to have precise control over inheritance.

In the HOM Method the terminologist must be free to assign parent

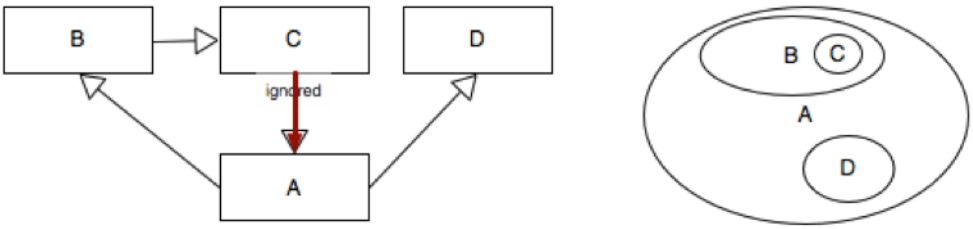
ontology annotations in a highly configurable manner. That way the order of parent traversal can be closely controlled.



To support the partial overriding of parent maps we need to cache maps before execution. Here B's Map 2 is overridden by C before execution of instance maps by A.

Figure 23. Instance Map Overriding: Support for partial overlap in function. Last parent defined wins.

If two parents both contain the same mapping then HOM defines inheritance such that the last definition of that mapping is the one that is executed on the clinical data. This rule is necessary to eliminate any possible ambiguity in the processing of the inherited content.



Cycles in the graph are ignored without halting graph traversal. Here parent D is still inherited even though the cycle back to A was skipped.

Figure 24. Cyclic checks: If the same URI is encountered then it is ignored.

The HOM Method also requires that inherited content check for cyclic

redundancy (loops). If the same ontology node is encountered it is simply ignored on the second pass. The rule is required to eliminate any possible ambiguity in the means by which content is inherited and as such is simply a convention used to eliminate any infinite loops that would otherwise be possible.

HOM Faceting Tags

The HOM Method also supplies one final HOM tag for a description of what are deemed useful search terms. The HOM_Facet tag is used to identify what target terminology/taxonomy terms or what ontology terms are useful when searching the data warehouse. This can be particularly useful when the target ontology is quite large and may confuse the researcher or QI staff that seeks to use the system.

HOM Search : Identify what sorts of data are to be used by search interfaces.

HOM_Facet IndexName [Search, ResultsList, Taxonomy:RelationName]

The HOM_Facet tag is used to mark terminologies and ontology terms as search interface terms. If a tag is applied then that tag and all of its children are included. This tag can be applied multiple times to the same ontology term and if this tag is applied to both a term and it's child then the child tag over-rides the parent. If no DisplayPropertyName is given it defaults to the PreferredName property.

Note: HOM_ParentOntology tags would separately be used to combine content. For example, if the "Ortho" ontology had the HOM_ParentOntology identified as "Demographics" then HOM_Facets would be pulled from both.

IndexName is the name of the search interface this facet it used for.

If the user can constrain the search by this term the "Search" parameter should be added. If the user should just see this term in the result set then the "ResultsList" parameter should be added. If "Taxonomy:RelationName" is given then the named relation (defaults to "isa") is used to traverse the ontology and generate a hierarchical taxonomy for use within the search interface.

Example: HOM_Facet "B020002 - CT Scan" Search
HOM_Facet "Ortho" Search
HOM_Facet "ICOM" ResultsList [DisplayNameProperty]

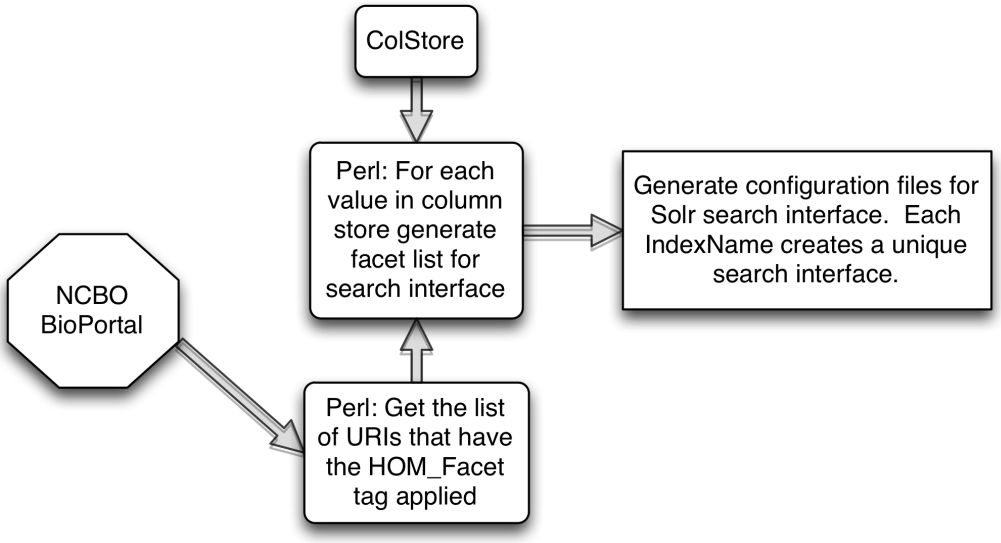


Figure 25. HOM_Facet tags to describe searching and browsing.

The HOM_Facet tag can be used to tag termologies/taxonomies and ontologies for use as search terms, result set entries or as hierarchical lists for browsing content. This tag is completely generic and can be used to describe any data warehouse search interface including i2b2, Apache Solr, RDF databases or data mining tools such as Business Objects, Pentaho and Cognos.

The usage of terminology server tags to describe the extraction of raw instance data from source biomedical databases, and the subsequent loading and transformation of that raw clinical data into biomedical terminologies and ontologies is a novel use of terminology server technology. This HOM Method has the potential to greatly increase the efficiency of data transformation when used with biomedical data sources and could provide an achievable means by which even small and short staffed community hospitals can analyze biomedical data for research, cost containment, population based decision support and quality improvement.

Data Loading

The HOM Method facilitates the loading of clinical data into a warehouse to

enable further analysis. Data can be loaded into any Fact Table schema (also referred to as Entity/Attribute/Value or EAV) but best performance is expected when the HOM Method is used with column-store based databases such as Sybase IQ, MonetDB, Vertica or InfiniDB. By enabling the translation of instance data after information has been loaded into a warehouse the HOM Method alleviates the need to translate clinical information statically and no longer requires the employment of IT development staff to translate clinical data during the warehouse loading process. Traditional warehouse data loading is called ETL (Extract Transform Load) processing whereas HOM Method uses ELT (Extract Load Transform) processing that is often referred to as the HOM UETL (Universal ETL method).

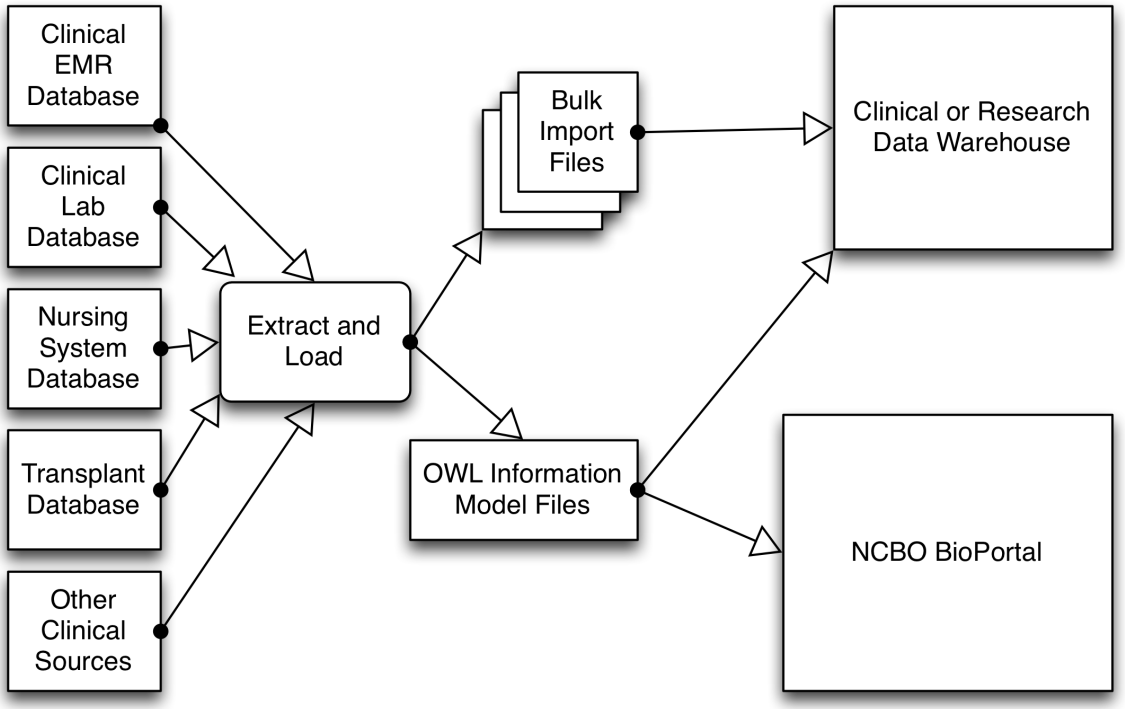


Figure 26. The HOM Method loading process: OWL (Web Ontology Language) files are generated that contain an Information Model describing the source schema on which instance maps run. URI's (Universal Resource Indicators) referring to elements within those information models are used to describe both BioPortal instance maps as well as the fact table IDs within each hospitals warehouse. This keeps the meaning of local hospital instance data in sync with the meaning of source data instance maps as described on BioPortal.

As previously described with the HOM tags, the HOM Method ELT process generates two sets of files. First it generates bulk import files for loading data into the warehouse. These bulk import files are a native database format supported by all database vendors. Bulk import files are the fastest possible means of importing data into any warehouse. Data that is not tagged with HOM filters is loaded into the warehouse unmodified and will be stored within the warehouse in the same format as it was read from the

data source. Later the HOM Method translates that information within the warehouse into multiple formats simultaneously resulting in a great deal of wasted logical disk space. But when the HOM Method is implemented on a column store (such as MonetDB) or vector database this results in little physical disk usage due to the natural column-based compression offered by column store databases. For example, the same UMLS CUI may be generated during translation for millions of input records but within a column store database that information is only represented on physical disk once resulting in little waste of physical disk and no impact on subsequent query performance.

This usage of a column store database as a means to compensate for the logical inefficiency of storing highly repetitive biomedical terminology is novel. Other warehouse solutions, such as the native i2b2 implementation, seek to assign meaning at time of query using predefined "modifiers" to define subsets within the data. But the HOM Method instead allows for sets within the warehouse be identified by URIs and the same fact can be stored many times and associated with many URIs in an ad-hoc fashion, thereby allowing the description of an unlimited number of ontology based

subsets. This is accomplished without wasting physical disk space and without any effect on query performance because HOM recommends the usage of column store databases (instead of row oriented databases) when deploying data warehouse technology with this method.

The second set of files generated in the HOM Method is the OWL Information Model files (also often called concept dimension files). These information model files encode the location of the data within the data source in OWL format. That data can later be referenced as a simple hierarchical list of parent child terms relating the name of the source, the name of the table and the name of the source column from which the data was read.

DataSourceName \ TableName \ ColumnName:

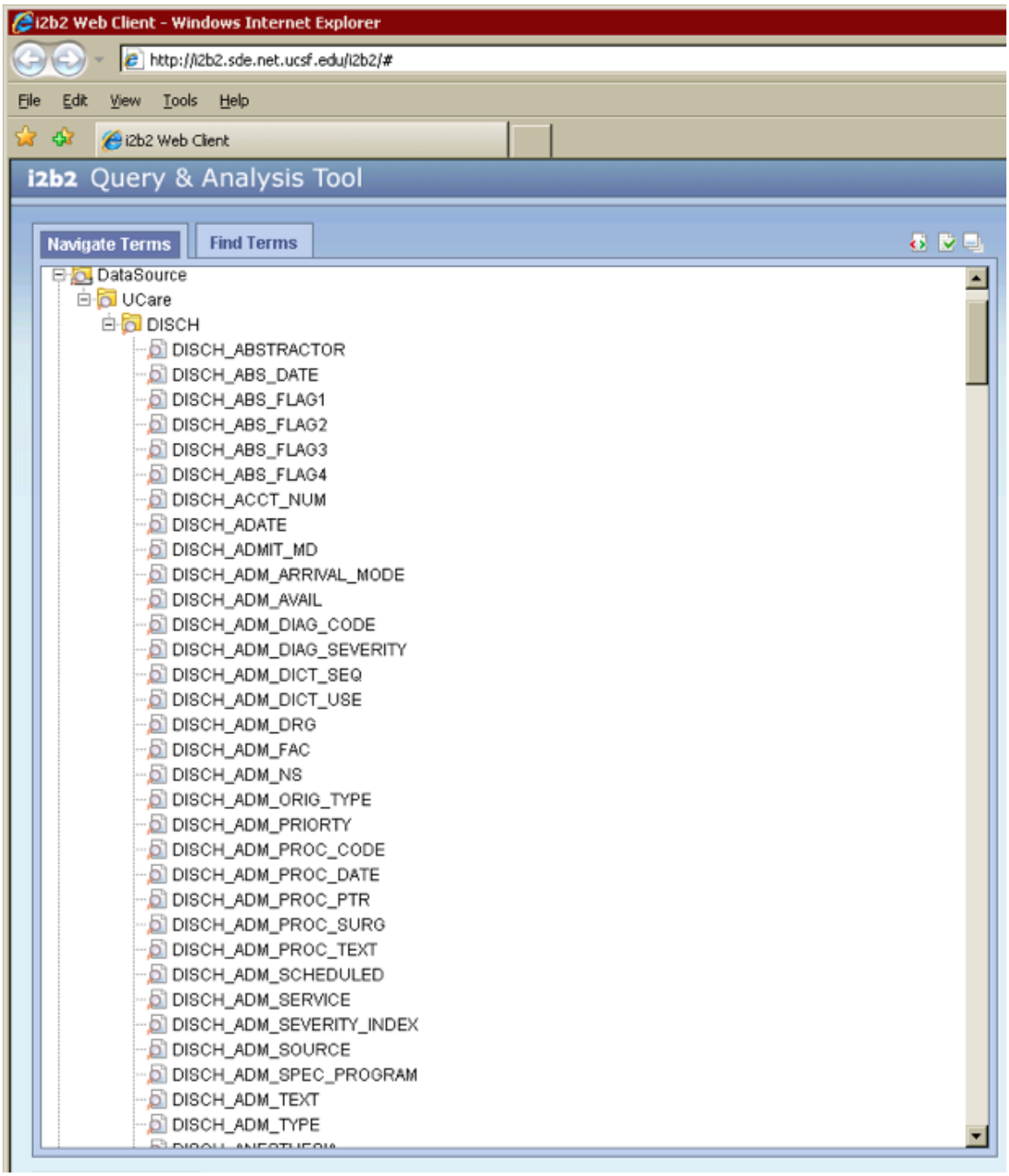


Figure 27. The discharge disposition data loaded into i2b2 in raw format for UCare.

Using this simple representation for the location from which the source data was loaded we can then construct an URI (universal resource indicator) for the terminology server (the NCBO BioPortal). The OWL

Information Model files are then loaded automatically over REST services into the NCBO BioPortal. That exact same URI is also then used to identify the type of data within the data warehouse. Centralized terminologist staff utilizes these BioPortal information models to define the instance data maps into biomedical terminologies and ontologies. These instance data maps run within the hospital data warehouse on the hospital instance data and are defined on the terminology server in a standard method.

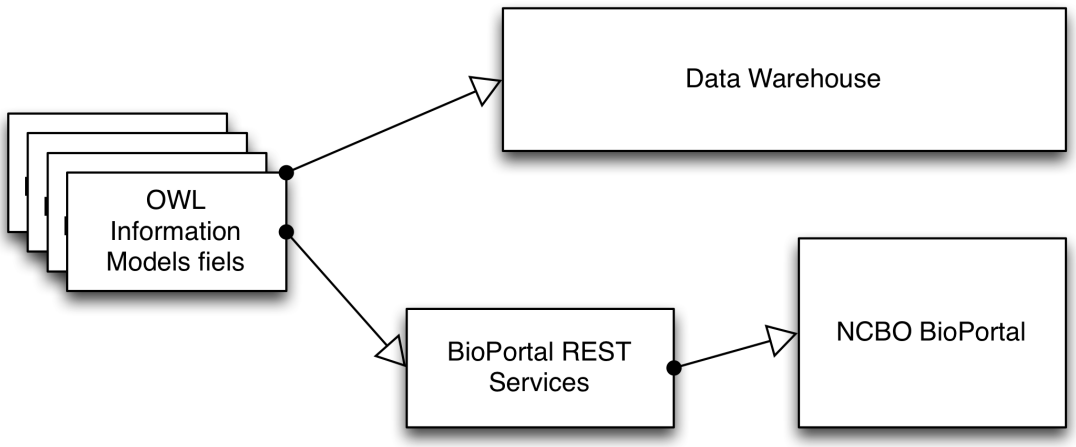


Figure 28. OWL Information Model files are used to load both the concept path (the fact ID) for any fact table based warehouse (such as i2b2) as well as the information models within BioPortal for those exact same data sources. Both the warehouse fact ID and the terminology server information models, terminologies and ontologies all reference the exact same purlz based URI's.

Once the information model files and raw source data have all been loaded both the data warehouse and the BioPortal reference the same concept identifiers for source data based on their common set of URI's. This method of using URI's, that are identified on a terminology server, as a

formal and computable definition of the meaning of warehoused clinical instance data is novel. Traditional EAV based data warehouse designs do nothing to automate, nor require the formal definition of, the meaning of the information that they contain. The HOM Method does enforce that definition and removes it to an external and Internet accessible terminology server in order to enable a centralized staff to define that meaning in a highly efficient and scalable manner.

Unstructured text handling during load

The HOM UETL component also contains an embedded copy of the NCBO Annotator service for annotating unstructured text. By using Annotator, clinical findings extracted from source clinical environments can be annotated with BioPortal medical terminologies such as SNOMED/CT. The annotator feature supports named entity recognition and negation. Annotator is not a fully featured NLP (natural language processing) environment but instead is packaged as an automated annotation component used internally by HOM and only during the data loading process. When HOM runs Annotator on incoming full-text (unstructured) data it first identifies a set of BioPortal URI's for portions of medical

terminologies stored on BioPortal. HOM selects multiple URI's to be annotated for topic areas of interest so that the same unstructured data can be interpreted within multiple contexts.

For example if HOM uses Annotator to select terms of interest in Cardiology, Orthopedic Surgery, and Pediatrics then annotations would be subsequently generated on the same unstructured text multiple times, once for each of those 3 domains. In this manner the HOM Method can select specific types of unstructured clinical findings and annotate those findings for usage within multiple domains of interest. The HOM Method uses the multivariate nature of HOM_Bool statements to choose the domain within which the text should be interpreted. For example, if the patient was admitted to the Orthopedics department then Annotator results may only be selected from the Orthopedics domain while ignoring all other possible interpretations.

The HOM Method when applied to Annotator chooses to interpret all text within all domains of possible interest, by calling Annotator multiple times as a result of multiple HOM_NCBOAnnotator tags each with specific

terminologies. Since annotator is called during the warehouse load process that has no effect on query performance. Since repeated data is compressed (as described above), that reinterpretation and regeneration of highly redundant biomedical terms does not result in any wasted physical disk space, nor does it result in slower query performance. The specific domain of interest is only chosen later, and in conjunction with other HOM_Bool maps to identify the likely domain of interest. It's not clear if this usage of NCBO Annotator is novel but I have been unable to find any other reference to it.

Instance Mapping

After the data is loaded HOM and the BioPortal can then be used to dynamically translate warehoused information by traversing maps defined on BioPortal. These maps translate information from source data information models into multiple standard medical terminologies and ontologies. For example, the local hospital discharge data stored within both GE UCare as well as within EPIC can be translated into the same HL7 Discharge Disposition format. Subsequent maps that utilize discharge disposition can then reference the standard HL7 Discharge format. After a

map has run the same data exists within the warehouse in both its raw un-translated form and in one or more translated standard medical terminologies. Additional mappings of the same source data can then be added at any time in the future without any need to reload the source data. It is expected that terminologist staff will define and implement new biomedical terminologies using previously loaded HOM data even many years following the data loading process.

These usages of ELT processing as a means to completely de-couple the exaction, and load of medical data from its subsequent transformation is novel. In the HOM Method, staff that has never even visited the hospital in question potentially defines EL and T processing, and transformation can be defined many years after the loading process has completed.

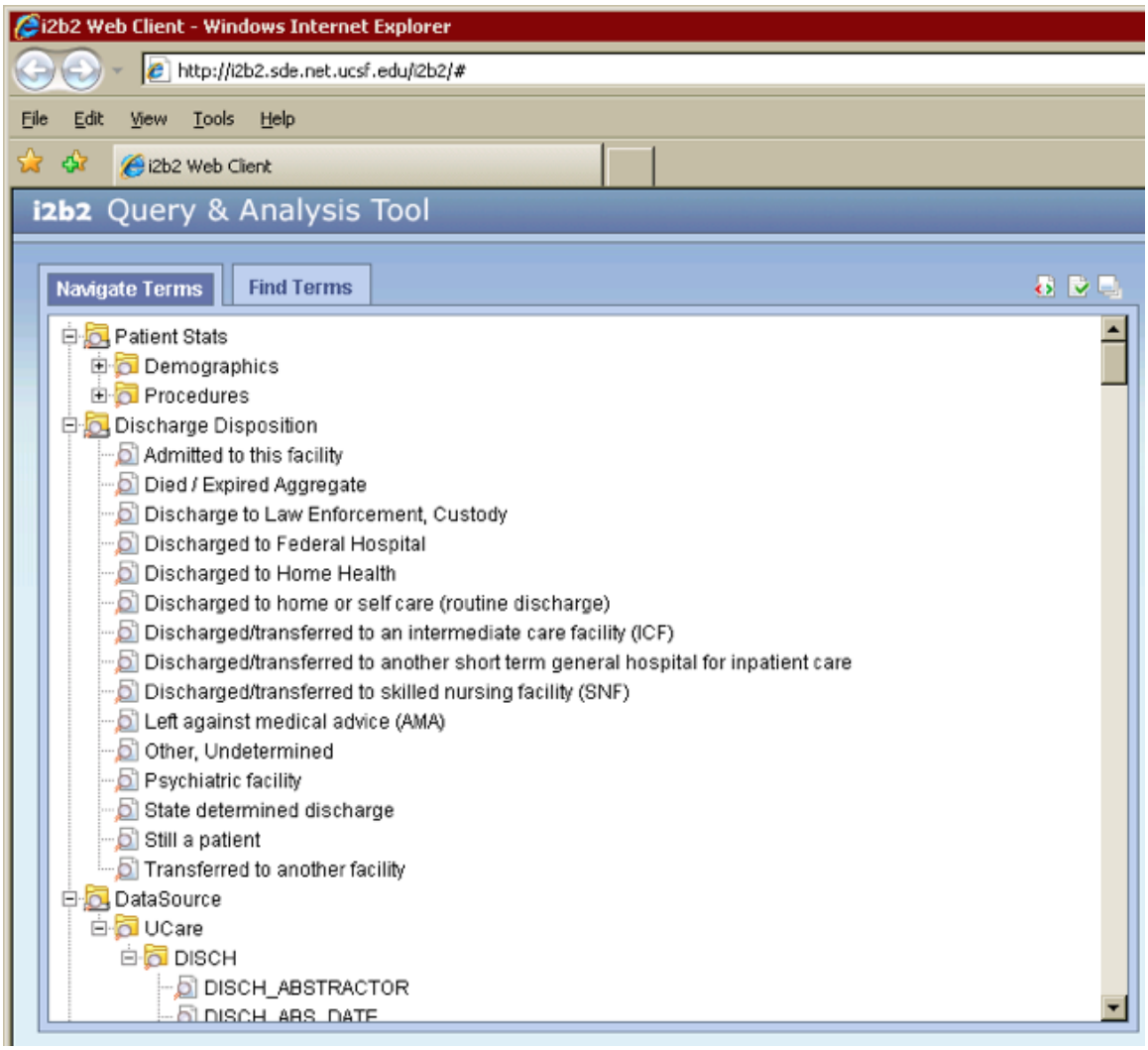


Figure 29. Discharge disposition data post-map in HL7 Discharge Disposition format.

The HOM Interpreter dynamically translates local clinical instance data by communicating with the BioPortal REST services API (application program interface). This translation into standard ontologies happens when requested by the researcher and after the data has already been loaded into the warehouse.

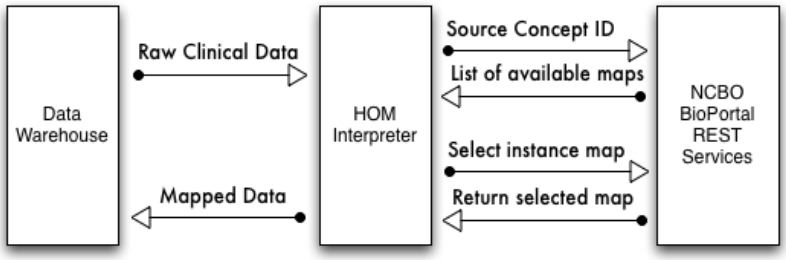


Figure 30. HOM Interpreter maps local instance data.

The instance maps stored on BioPortal can define three different classes of clinical instance data maps.

1) 1-to-1 maps
2) many-to-1 maps
3) many-to-many maps (also called “automatic maps”)

The HOM 1-to-1 maps will translate a single term within the value set of the source data system into a single term for the value set of the target medical terminology by using the BioPortal “Term Mapping” feature. The HOM many-to-1 maps will look for the presence of multiple value set terms from the source data and translate that information into a single target terminology term by evaluating a HOM_Bool statement.

Many-to-many maps utilize the results of multiple 1-to-1 and many-to-1 maps and may also execute function calls to R, Weka or the RxNav server

for drug normalization. Examples of these “auto maps” include the normalization of clinical lab data into bins of “Low”, “Low-Normal”, “Normal”, “High-Normal” and “High” following the automated generation of reference intervals.

This combination of extended terminology server features to include HOM Bool expressions utilizing a derivative of the SQLite Expression Syntax is novel. HOM has provided a standard means of describing instance data mapping both into and between medical terminologies and ontologies.

The above-mentioned HOM architecture has been implemented repeatedly at multiple institutions.

Experiments

The HOM Method has been evaluated as a semantic interoperability component on two CTSA projects. Those included the CICTR (Cross Institutional Clinical Translational Research) project and the CELDAC (Comparative Effectiveness Large Data Analytics Core) [48] grant.

CICTR Grant Tests

On the CICTR grant HOM was used for two separate semantic interoperability tests. Those included the LabNorm clinical lab data normalization test and the RxNorm drug dispensary normalization test.

In the RxNorm test the HOM Method was used to implement dispensary data normalization for the CICTR (Cross Institutional Clinical Translation Research) project at UCSF, UC Davis and UW [44]. In the LabNorm test HOM was used to provide clinical lab data normalization using data from UCSF and UW for the LabNorm project [43].

RxNorm Experiment Design (Scalability)

The RxNorm service was designed to run on the Health Ontology Mapper (HOM) platform and communicate in the background with both the NCBO BioPortal and the RxNav terminology server to translate IDR formulary data into the RxNorm standard automatically for any i2b2.org -based integrated data repository system. Once translated, the formulary records were then queried via the i2b2 workbench using RxNorm standard terms in addition to the original local terms. Additionally the data from multiple sites were

then connected over the SHRINE network for query. Although the locally encoded drug terms would not produce query results across multiple hospital systems, it was expected that the resulting RxNorm equivalent terms would be aggregated over SHRINE producing a single large federated database of dispensary data.

The RxNorm HOM map worked by iterating through the I2B2 Observation_fact table, and identifying medication records by the appropriate concept id and prefix. I2B2 has an existing link between the concept_cd and a local drug name and/or the NDC code. For each drug name/NDC code it found, the map invoked a search of RxNorm, via the RxNav middleware to find all related standardized terms for the same source drug. For each term it found, a new record was added to the observation-fact table indicating the association of a patient with the standardized drug name in addition to the original record associating the patient with the non-standardized drug name. The script then moved onto the next medication record in Observation fact. If the drug name has not been seen before, a fresh query of RxNorm via RxNav is performed, but if a new instance of a previously mapped drug name is seen, the cached version

of the map was used for the translation, thereby improving the efficiency of the map. (Caching of previous map results was added as a feature to the HOM engine). The net result of this process was a greatly enlarged observation_fact table that has multiple versions of the same medication record for each patient, reflecting the many different RxNorm terms that are analogous to the drug name from the original source data. RxNorm translated formulary records are easier to navigate and query than the raw formulary records. Also, the query of this RxNorm translated formulary offers far faster execution times than possible when attempting to translate the formulary records at point of query.

CICTR Participating Sites

The HOM RxNorm map [44] was run against all 3 of the participating SHRINE network sites on the CICTR grant, each site with an institution specific drug dictionary, with a total patient count of 1,169,322.

UW: 409,518
UCD: 679,632
UCSF: 80,172 : Total Patients: 1,169,322

A second query was run at the University of Pennsylvania and the University of Rochester to test the reliability of the mapped RxNorm terms

against the source formulary data.

In subsequent HOM RxNorm tests the formulary records for U. Penn and U. Rochester were also standardized to RxNorm with formulary record counts of 4,433,542 at U. Penn and 3,381,432 at U. Rochester to show that the HOM Method is reliable even with very large patient data sets.

Although this complex mapping task took several hours to complete the resulting queries in i2b2 all executed in less than 3 minutes.

The mapping results were successful and as a test of the system we demonstrated a cross institutional (grid-based) query via SHRINE for patients between 40-60 years old with Diabetes that were prescribed the drug Heparin, even though the word “heparin” was not part of the drug name stored within each local formulary. This query of over 1 million dispensary records across 3 separate hospital systems was completed in less then 3 minutes time.

LabNorm Experiment (Biostatistics)

Additionally on the CICTR a second experiment was performed on the use of the HOM Method for clinical lab data normalization.

Since the IDR contains the intersection of a large number of clinical data sources it is possible to select patient lab data that is dependent on the presence or absence of specific diagnosis (ICD-9) codes. As such the posteriori selection of patients can be restricted to those that are considered “normal” for the automated calculation of clinical lab reference intervals (RI). By leveraging this additional information we can automatically generate multiple reference intervals (for each ICD-9 code) within an inpatient hospital setting. Lab data was automatically loaded into the IDR via the Health Ontology Mapper platform in a raw and un-translated state. Once loaded a HOM instance map named Lab Norm provides a best-fit translation of the imported analyte values resulting in a set of automatically generated reference intervals.

The analyte values selected for line fitting were chosen at random from within the IDR and based on diagnosis (IDC-9) codes indicative of “normal” patients. Lab Norm then translated the lab data again into a normalized

space relative to the appropriate reference interval for each analyte measurement and the derived results were added back into the i2b2 IDR. The Lab Norm system ran in a completely automated fashion. Manual steps are required for quality control and system monitoring but the normalization of lab data imported into the IDR was handled as an automated background process.

The LabNorm HOM map [43] was executed on test data supplied as part of the CICTR grant and was shown to successfully predict reference intervals for clinical lab data using the i2b2 IDR.

CELDAC Test (Usability)

The HOM Method was used to implement the CELDAC (Comparative Effectiveness Large Dataset Analytics Core) grant [48] that provided a warehoused and mapped form of the State of California OSHPD database for the rapid analytics of data in public health. The OSHPD database is comprised of several smaller regional databases. The smaller databases differ slightly in structure and data quality making the aggregation of all OSHPD data for research purposes time consuming and difficult for

researchers. On the CELDAC grant a single large warehouse based on i2b2 was constructed to house the combined total of all OSHPD datasets.

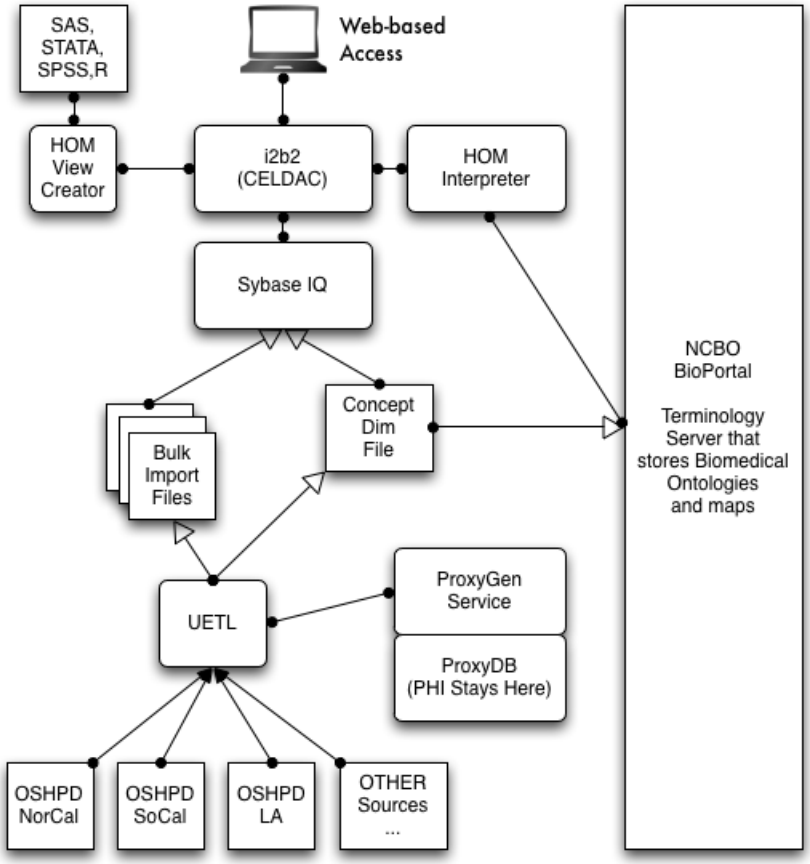


Figure 31. Complete high-level HOM Architecture that was used to implement the CELDAC grant for analysis of OSHPD data.

HOM was used to successfully aggregate all OSHPD data on the CELDAC grant and successfully complete the aims of the project. This system then was used to serve data requests for public health researchers at UCSF.

ICD-10 Code Generation (Accuracy)

This experiment provides a measure the effectiveness of HOM at improving the efficiency and utility of medical terminology data.

It was shown that the HOM Method can be used to generate ICD-10 detail codes based the translation of ICD-9 codes and unstructured clinical notes. [49]. In this test ICD-9 codes were used as input along with clinical discharge summary text. The discharge summaries were first translated into UMLS and then a multivariate HOM map was used to generate the valid ICD-10 detail code that closest matches the clinical event.

Currently in the USA medical diagnosis are encoded in ICD-9 format. Unlike Europe the USA also has many payers that reimburse medical providers based on these diagnosis codes. Soon, the USA will switch to the ICD-10 terminology for describing diagnosis. ICD-10 has over 10 times as many terms and will provide a record of patient diagnosis which is far more medically relevant.

Due to the nature of the "many payer" medical system in the USA the

transition from ICD-9 to ICD-10 could pose a public health challenge. Each year medical centers continuously negotiate and re-negotiate their reimbursement rates with payers relative to these codes. As the deadline for switching over to the ICD-10 standard approaches that negotiation process will need to be made on the new ICD-10 standard.

However, it is not currently possible for a provider to determine the distribution of ICD-10 codes that should be expected in the years ahead. Also, since there are many payers in the USA, each of which has unique state legal mandates, the remediation process followed by payers needs to follow a consistent algorithm based on the distribution of ICD-10 codes for which payments are to be made. But if the payers cannot predict the distribution ICD-10 codes used during the remediation process then the actuarial analysis generated by the payer would be inaccurate. This creates an environment that is highly unpredictable for both the providers and the payers as they negotiate their transition from ICD-9 to ICD-10.

HOM was used to create a common medical term “cross-walk” algorithm to predict that future ICD-10 distribution for any hospital that uses this ICD-10

code generation method.

Clinical data was first translated into standard biomedical terminologies and then using the context provided by the existing set of ICD-9 codes a cross-walk was constructed that determined the equivalent ICD-10 code that would have been expected had the same encounter occurred following the ICD-10 mandate.

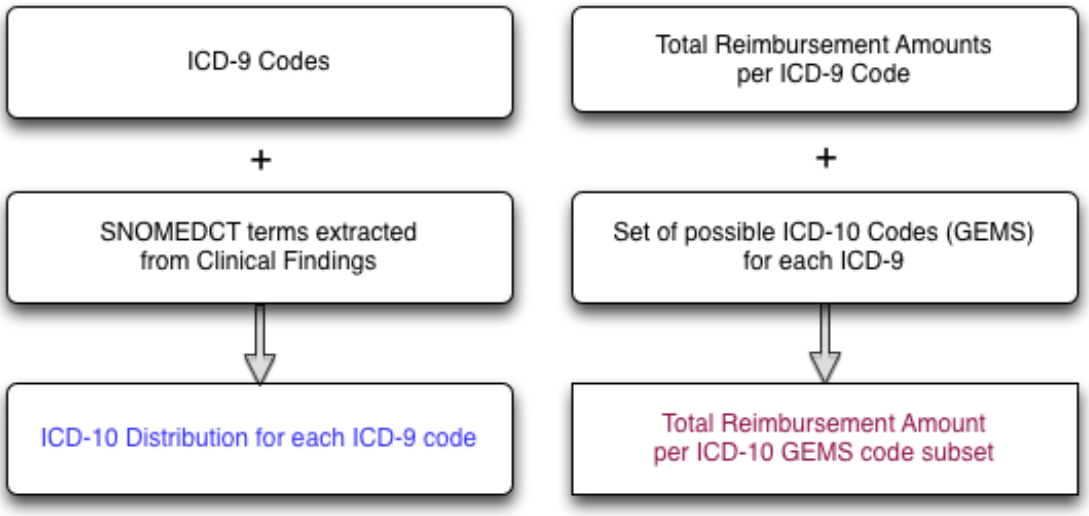


Figure 32. ICD-10 detail code generation: Using GEMS to calculate total reimbursement.

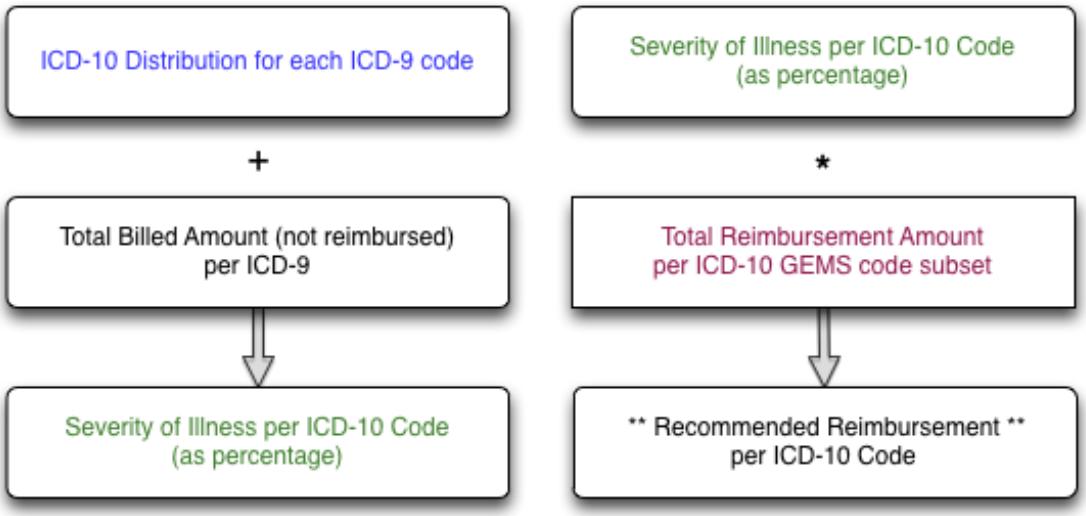


Figure 33. ICD-10 detail code generation: Predicting future ICD-10 detail code distributions based on historical ICD-9 data.

The goal of this test was to demonstrate automated ICD-10 detail code generation from clinical encounter data that has already been coded using ICD-9. We generated a distribution of ICD-10 detail codes (not GEMS groupings) for each ICD-9 based clinical encounter.

For this test we selected high complexity ICD-9 codes, each of which maps to 3 or more potential ICD-10 codes) within high cost categories. The selected ICD-9 codes were for Aortic Valve disorders.

Next HOM Bool annotations were generated for each clinical event type that maps into a specific ICD-10 detail code. The distribution and Severity

of Illness for these sets of ICD-10 codes were then generated, and to measure the accuracy of the results financial records were consulted to verify that the billed amounts matched the predicted reimbursements for the generated ICD-10 detail codes. The aggregated billed amounts for the predicted ICD-10 detail codes and the actual billed amounts in ICD-9 were then compared and shown to match.

In this test we verified the crosswalk using clinical and financial data from Stanford University. First a crosswalk was created that combined ICD-9 codes with SNOMEDCT terms that had been extracted from clinical findings text resulting in ICD-10 codes. This crosswalk was run on the entire test set of source ICD-9 codes. We then obtained the total reimbursement amounts for each ICD-9 code and used the CMS ICD-10 GEMS crosswalk to determine the total reimbursement amount per GEMS code subset. For this test it was assumed that the severity of illness closely tracks the total billed amount for each patient. We then combined the ICD-10 distribution generated above with the total billed amount (in ICD-9) to generate a Severity of Illness (SOI) per ICD-10 code as a percentage of 100. The SOI was combined with the total reimbursed amount for each GEMS set to

result in a recommended reimbursement for each ICD-10 detail code. These values were then used to verify the accuracy of the ICD-10 HOM map.

ICD-10 Code Generation Results

For the ICD-9 code 424.1 (Aortic Valve Disorders) we calculated ICD-10 detail codes for 9,098 patients from Stanford University with 701 patients identified as readmissions. The total reimbursement amount in ICD-9 (for the year 2010) was $120,251,480 or $13,217 per patient.

The ICD-9 codes were combined in a HOM crosswalk with SNOMEDCT codes extracted from clinical findings to produce ICD-10 detail codes. For this test those ICD-10 codes were:

1351 (Nonrheumatic Aortic Insufficiency) with 3,995 patients
1359 (Nonrheumatic Aortic Valve disorder Unspecified) with 0 patients
1352 (Nonrheumatic Aortic Stenosis with insufficiency) with 7 patients
1350 (Nonrheumatic Aortic Stenosis) with 3,924 patients
1358 (Nonrheumatic Aortic value disorders) with 1,873 patients

Total patients = 9,799 – 701 readmissions = 9,098 patients.

Having generated the ICD-10 codes and verified that the total number of patients matches, we then used accounting to verify the accuracy of the map.

ICD-10	SOI	Proposed Reimbursement	# Patients	Cost in ICD-10
1351	1.068	$14,116	3,995	$56,393,420
1359	0.7	$9,252	0	0
1352	0.96	$12,688	7	$88,816
1350	0.9	$11,835	3,924	$46,440,248
1358	0.7	$9,252	1,873	$17,328,996
		Totals:	9,799	$120,251,480

Figure 34. Verification of ICD-10 code generation using financial data.

The ICD-10 detail codes were then used to generate proposed reimbursement amounts and the total reimbursement amount calculated was shown to exactly match the reimbursed amount recorded in ICD-9.

Discussion

The HOM method drives forward several innovations that were required to enable the web-based description of biomedical data analysis.

1. HOM promotes an OWL based terminology server as the driver for ELT data transformation (Extract, Translate and Load). This approach

is not based on UML (Unified Modeling Language) but is instead a form of semantic web technology.

2. HOM describes a means by which terminology server annotations can determine the interpretation of HIPAA data requirements and control the generation of a HIPAA Limited Data Set (LDS) from clinical information.

3. HOM provides a means by which annotations on the terminology server are attached to auto-generated schema descriptions. This allows URI's to be used to describe source data originally collected in relational databases so that terminology server annotations may be attached to them.

4. HOM promotes the loading of relational data information models and the subsequent mapping of that data into biomedical terminologies and further into ontologies. The description of the UML based source data, and the subsequent mapping into higher forms is all provided on a single cohesive platform.

5. The HOM reference implementation used column store compression to allow ad-hoc URI based subsets within any data warehouse.

6. HOM uses URI's associated with fact ID's as its definition of "type".

7. HOM promotes the brute force annotation of text into all possible domains of interest and the subsequent mapping of that data in a multivariate context into target ontologies. For example, the annotation of text with UMLS, RxNorm and LOINC simultaneously and the subsequent mapping of those cui's along with clinical ICD-9 codes into ICD-10 detail codes.

8. HOM provides a SQLite based syntax for multivariate instance data mapping (HOM_Bool statements).

9. HOM defines ontology-based instance-mapping inheritance to enable the efficient reuse of instance data mapping content.

All content is managed centrally on a terminology server accessible via a web browser. Mapping content may be added to the server at any time, even years after the source data information model has been loaded in a just-in-time (on demand) fashion and thereby relieving experts from the requirement to define ontology requirements when data is first loaded.

These innovations were necessary to decouple ontology experts from the clinical sites they wish to service. This HOM method allows ontology experts to define the ELT based processing and analysis of clinical data without the need to physically visit the clinics in question. By enabling this web-based content management approach the analysis of medical data from small clinics and community hospitals is now an achievable goal.

Conclusion

Limited human resources exist to assist in the interpretation of clinical data. The HOM Method provides a possible means by which the existing and small population of ontologists in the USA, may drive the analysis of data for every hospital and outpatient clinic. This may be achieved by enabling the online collaboration of ontologists to generate a single shared

knowledge management resource that describes clinical instance data mapping and usage.

These methods enable the context specific interpretation of data, across multiple hospital systems without the need to transmit sensitive HIPAA regulated data. Data can be rapidly loaded and analyzed from any clinical source including claims data, structured and unstructured clinical notes and semi-structured clinical encounter data. HOM enables the generation of a common set of concept ID's with linkage to formally defined OWL based ontologies that are common to all data warehouses for consistent analysis and virtual aggregation over grid computing platforms.

These methods could be used to drive forward subsequent innovations in biomedical informatics, including the creation of instance data mapping content based on SparQL or other innovative RDF (Resource Description Framework) based syntax. This HOM Method enables virtual data aggregation across multiple clinical systems and can be used to further progress in quality improvement, health system financial forecasting,

decision support, computable clinical phenotype identification and biotech cohort selection.

References

1. Wynden R, et al. Ontology Mapping and Data Discovery for the Translational Investigator. AMIA CRI Summit 2010. [AMIA Distinguished Paper Award; http://bmir.stanford.edu/publications/view.php/ontology_mapping_and_data_discovery_for_the_translational_investigator]

2. Overview – i2b2 [cited 2009 October 31] – 1. Noy NF, Musen, MA. The PROMPT Suite: Interactive Tools For Ontology Merging And Mapping. International Journal of Human-Computer Studies 2003;59(6):983-1024.

3. Brinkley JF, Suciu D, Detwiler LT Gennari JH, Rosse C. A framework for using reference ontologies as a foundation for the semantic web. Proc. AMIA Symp. 2006; 96-100.

4. Gennari JH, Musen MA, Fergerson RW, Grosso WE, Crubézy M, Eriksson H, Noy NF, Tu SW. The evolution of Protégé: an environment for knowledge based systems development. International Journal of Human Computer Studies 2003; 58(1):89-123.

5. Advani A, Tu S, O'Connor M, Coleman R, Goldstein MK, Musen M. Integrating a modern knowledge-based system architecture with a Legacy VA database: The ATHENA and EON projects at Stanford. Proc. AMIA Symp. 1999; 653-7.

6. Serving the enterprise and beyond with informatics for integrating biology and the bedside (i2b2). Murphy SN, Weber G, Mendis M, Gainer V, Chueh HC, Churchill S, Kohane I., J Am Med Inform Assoc. 2010 Mar-Apr;17(2):124-30.PMID: 20190053

7. Guoqian Jiang, Christopher G. Chute: Comparing the Effects of Two Semantic Terminology Models on Classification of Clinical Notes: A Study of Heart Murmur Findings. KR-MED 2008

8. Philip V. Ogren, Guergana K. Savova, Christopher G. Chute: Constructing Evaluation Corpora for Automated Clinical Named Entity Recognition. LREC 2008

9. Ted Pedersen, Serguei V. S. Pakhomov, Siddharth Patwardhan, Christopher G. Chute: Measures of semantic similarity and relatedness in the biomedical domain. Journal of Biomedical Informatics 40(3): 288-299 (2007)

10. Natalya Fridman Noy, Nigam H. Shah, Patricia L. Whetzel, Benjamin Dai, Michael Dorf, Nicholas Griffith, Clement Jonquet, Daniel L. Rubin, Margaret-Anne D. Storey, Christopher G. Chute, Mark A. Musen: BioPortal: ontologies and integrated data resources at the click of a mouse. Nucleic Acids Research 37(Web-Server-Issue): 170-173 (2009)

11. Jyotishman Pathak, Guoqian Jiang, Sridhar O. Dwarkanath, James D. Buntrock, Christopher G. Chute: Adopting Graph Traversal Techniques for Context-Driven Value Sets Extraction from Biomedical Knowledge Sources. ICSC 2008: 460-467

12. Shah NH, Bhatia N, Jonquet C, Rubin D, Chiang AP, Musen MA: Comparison of concept recognizers for building the Open Biomedical Annotator. BMC Bioinformatics. 2009 Sep 17;10 Suppl 9:S14.

13. Jonquet C, Shah NH, Musen MA.: The open biomedical annotator., Summit on Translat Bioinforma. 2009 Mar 1;2009:56-60.

14. Canfield K, Silva M, and Petrucci K, *The standard data model approach to patient record transfer.* Procedures of the Annual Symposium on Computational Applications in Medical Care, 1994: p. 478-82.

15. Burdis C, Eaglestone B, & Procter P, *A unified model to support an information intensive health care environment.* Studies in Health Technology and Information, 1999. **68**: p. 171.

16. Barrows Jr. R. & Johnson S, *A data model that captures clinical reasoning about patient problems.* Procedures of the Annual Symposium on Computation Applications in Medical Care, 1995: p. 402.

17. Friedman C, et al., *The Canon Group's Effort: Working Toward a Merged Model.* Journal of the American Medical Informatics Association, 1995. **2**(1): p. 4.

18. Pollard D, & Hales J, *Evaluation of an object-based data model implemented over a proprietary, legacy data model.* Proc Annu Symp Comput Appl Med Care, 1995: p. 367.

19. Dore L, et al., *An object oriented computer-based patient record reference model.* Proc Annu Symp Comput Appl Med Care, 1995: p. 377.

20. Gouveia_Oliveira A & Lopes L, *Formal representation of a conceptual data model for the patient-based medical record.* Proc Annu Symp Comput Appl Med Care, 1993: p.466.

21. Rector, A.L., Thesauri and formal classifications: terminologies for people and machines, Methods Inf Med, 1998. 37[4-5]: p.501-9.

22. Bressan, S. and C. Goh, *Semantic Integration of Disparate Information Sources over the Internet Using Constraints.* 1997.

23. Sciore, E., M. Siegel, and A. Rosenthal, *Using Semantic Values to Facilitate Interoperability Among Heterogeneous Information Systems. ACM Transactions on Database Systems, 1994. 19(2):p.254.*

24. Goh, C., et al., Context Mediation: New Features and Formalisms for the Intelligent Integration of Information. 1997, Sloan Working Paper 3941.

25. Bressan, S., The COntext INterchange Mediator Prototype. ACM SIGMOD International Conference on Management of Data, 1997.

26. Chu, W., Merzbacher and Berkovich. The Design and Implementation

of CoBase, Proceedings of ACM SIGMOD. 1993

27. Chu, W., et al., CoBase: A Scalable and Extensible Cooperative Information System, Journal of Intelligent Information Systems, 1996

28. Garcia-Molina, H., et al., The TSIMMIS Approach to Mediation: Data Models and Languages. 1997.

29. Papakonstantinou, Y., et al., A Query Translation Scheme for Rapid Implementation of Wrapper, 1995.

30. Rector, A.L., et al., *Reconciling users' needs and formal requirements: issues in developing a reusable ontology for medicine.* IEEE Trans Inf Technol Biomed, 1998. p. 229-42.

31. Rogers, J.E., et al., *Validating clinical terminology structures: integration and crossvalidation of Read Thesaurus and GALEN.* Proc AMIA Symp, 1998: p. 845.

32. Solomon, W.D., et al., *Having our cake and eating it too: how the GALEN Intermediate Representation reconciles internal complexity with users' requirements for appropriateness and simplicity.* Proc AMIA Symp, 2000: p. 819.

33. Masarie, F.E., et al., *An Interlingua for Electronic Interchange of Medical Information: Using Frames to Map between Clinical Vocabularies.* Computers and Biomedical Research, 1991.

34. Grimson, W., et al., *Federated healthcare record server--the Synapses paradigm.* Int J Med Inf, 1998. **52**(1-3): p. 3-27.

35. Heimbigner, D. and D. McLeod, *A Federated Architecture for Information Management Systems.* ACM Trans Office Info Systems, 1985.

36. Hurlen, P. and K. Skifjeld, *Design and functional specification of the Synapses federated healthcare record server. Synapses Consortium.* Stud Health Technol Inform, 1997.

37. Mostardi, T. and C. Siciliano. *An overview of WIND (Wide Interoperable Networked Databases)* in *Twenty-Seventh Hawaii International Conference on System Sciences*. 1994.

38. Sheth, A. and J. Larson, *Federated Database Systems for Managing Distributed Heterogeneous and Autonomous DataBases.* ACM Computing Surveys, 1990. **22**(3):

39. Gligor, V. and G. Luckengaugh, *Interconnecting Heterogeneous Database Management Systems.* IEEE Computer, 1984: p. 33.

40. Wiederhold, G., et al. *KSYS: An Architecture for Integrating Databases and Knowledge Bases.* in *Integration of Information Systems: Bridging Heterogeneous Databases*. 1989, IEEE Press.

41. *ISO/IEC 19757-3*. ISO/IEC . 1 June 2006. p. vi.

42. Pathak, J., Wang, J, Kashyap, S, Basford, M., Li, R., Masys, D Chute, C. "Mapping Clinical Phenotype Data Elements to Standardized Metadata Repositories and Controlled Terminologies: The eMERGE Network Experience." J Am Med Inform Assoc 2011; 18:376-386.

43. Rob Wynden, Donna L. Hudson (2010) Lab Norm: Automated Clinical Lab Data Normalization, *SEDE 2010 (International Conference on Software Engineering & Data Engineering)*, (ISCA) International Society for Computers and Their Applications, San Francisco, ISCA, Available from: http://www.escholarship.org/uc/item/62r2m5j2?display=all

44. Wynden R, Anderson N, Casale M, Lakshminarayanan P, Anderson K, Prosser J, Errecart L, Livshits A, Thimman T, Weiner M, Using RxNorm for cross-institutional formulary data normalization within a distributed grid-computing environment. AMIA Annu Symp Proc. 2011 ;2011:1559-63. Epub 2011 Oct 22. [http://www.ncbi.nlm.nih.gov/pubmed/22195221]

45. Rob Wynden, Michael Kamerick, MyResearch: Using Security to Speed the Adoption of the IDR, AMIA Proceedings 2011. [AMIA Distinguished Paper Award;

http://proceedings.amia.org/16pa5q/16pa5q/1].

46. DC. Nyulas, M.J. O'Connor, S.W. Tu. "DataMaster - a Plug-in for Importing Schemas and Data from Relational Databases into Protégé". 10th International Protégé Conference, Budapest, Hungary, 2007.

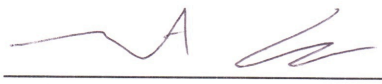
47. J. F. Sequeda, C. Cunningham, R. Depena, D. P. Miranker, Ultrawrap: Using SQL Views for RDB2RDF, 8th International Semantic Web Conference

48. Coffman Janet M., MA, MPP, PhD, Comparative Effectiveness Large Data Analysis Core (CELDAC) [http://ctsi.ucsf.edu/our-work/comparative-effectiveness-large-dataset-analysis-core-celdac]

49. AMIA-0825-A2012. An automated transition to ICD-10 encoding, Wynden R.; Grause H.; Mobed K.; Rekapalli H.; Lakshminarayanan P; Weiner M.

Publishing Agreement

I hereby grant permission to the Graduate Division of the University of California, San Francisco to release copies of my thesis, dissertation, or manuscript to the Campus Library to provide access and preservation, in whole or in part, in perpetuity.



Author Signature

06/12/2013
Date