2018-05-06

# KNowledge Acquisition and Representation Methodology (KNARM) and Its Applications

Hande Küçük McGinty
*University of Miami*, handemcginty@gmail.com

UNIVERSITY OF MIAMI

# KNOWLEDGE ACQUISITION AND REPRESENTATION METHODOLOGY (KNARM) AND ITS APPLICATIONS

By

Hande Küçük McGinty

A DISSERTATION

Submitted to the Faculty
of the University of Miami
in partial fulfillment of the requirements for
the degree of Doctor of Philosophy

Coral Gables, Florida

May 2018

UNIVERSITY OF MIAMI

A dissertation submitted in partial fulfillment of
the requirements for the degree of
Doctor of Philosophy

# KNOWLEDGE ACQUISITION AND REPRESENTATION METHODOLOGY (KNARM) AND ITS APPLICATIONS

Hande Küçük McGinty

Approved:

| | |
|---|---|
| Ubbo Visser, Ph.D.<br>Associate Professor of Computer Science | Hüseyin Koçak, Ph.D.<br>Professor of Computer Science |
| Geoff Sutcliffe, Ph.D.<br>Professor of Computer Science | Stefan Wuchty, Ph.D.<br>Associate Professor of Computer Science |
| Stephan Schürer, Ph.D.<br>Associate Professor of Molecular and Cellular Pharmacology | Guillermo Prado, Ph.D.<br>Dean of the Graduate School |

KÜÇÜK MCGINTY, HANDE (Ph.D., Electrical and Computer Engineering)

KNowledge Acquisition and Representation Methodology (KNARM) and Its Applications (May 2018)

Abstract of a dissertation at the University of Miami.

Dissertation supervised by Professor Ubbo Visser.
No. of pages in text. (156)

Technological advancements in many fields have led to huge increases in data production, including data volume, diversity, and the speed at which new data is becoming available. In accordance with this, there is a lack of conformity in the ways data is interpreted. In-depth analyses making use of various data types and data sources, and extracting knowledge has become one of the many challenges with this big data. This is especially the case in life-sciences where simplification and flattening of diverse data types often leads to incorrect predictions.

Effective applications of big data approaches in the life sciences require better, knowledge-based, semantic models that are suitable as a framework for big data integration, while avoiding overly extreme simplification, such as reducing various biological data types to the gene level. A major challenge in developing such semantic knowledge models, or ontologies, is the knowledge acquisition bottleneck. Automated methods are still very limited and significant human expertise is required.

In this research, we describe a methodology to systematize this knowledge acquisition and representation challenge, termed KNowledge Acquisition and Representation Methodology (KNARM). We also present how KNARM was applied on three ontologies: BioAssay Ontology (BAO), LINCS FramEwork Ontology (LIFE) ,and Drug Target Ontology (DTO) built for three different projects: BioAssay Ontology, Library of Integrated Network-Based Cellular Signatures (LINCS), and Illuminating the Druggable Genome (IDG), and how they work together in complex queries.

*To My Family.*

# Acknowledgements

First and foremost, I am most grateful to my advisor, Dr.Ubbo Visser who has supported me throughout this research with his extensive knowledge, encouragement, and patience, giving me confidence to develop different aspects of my research. Without him, this thesis could not have been completed or written. In addition to my advisor, I also would like to offer my sincere thanks to Dr. Hüseyin Koçak, Dr. Geoff Sutcliffe, Dr. Stefan Wuchty, and Dr. Stephan Schürer for their encouragement and insightful comments. I would like to acknowledge the Computer Science Department, Center for Computational Science (CCS) and National Institute for Health (NIH grants U54CA189205 (Illuminating the Druggable Genome Knowledge Management Center, IDG-KMC), U24TR002278 (Illuminating the Druggable Genome Resource Dissemination and Outreach Center, IDG-RDOC), U54HL127624 (BD2K LINCS Data Coordination and Integration Center, DCIC), and U01LM012630-02 (BD2K, Enhancing the efficiency and effectiveness of digital curation for biomedical big data)) for their financial support during my studies and toward the conferences I attended.

I am blessed to have such wonderful friends in and outside of the Computer Science Department, who supported, helped, and encouraged me even during the most difficult times of this period. I thank them all for everything, especially to Hema Raju, Basar Koç, Prajwal Devkota, and Seminda Abetruwan for listening to me in the office and their help in various parts of my research. I would like to thank the UM Writing Center for their help with the editing of this dissertation.

My studies would not have been completed without the everlasting love, constant support, and guidance of my parents Mahide and Yalçin Küçük, my brother Yigit Küçük, my parents-in-law, my husband Jacob McGinty, and my wonderful son William Sarp McGinty, who is the reason why I continued to finish this journey.

Hande Küçük McGinty

*University of Miami*

*May 2018*

# Table of Contents

# List of Figures

# List of Tables

# CHAPTER 1

# Introduction

*Big Data* has become one of the most popular subjects in business and research. Many research universities, such as University of Michigan [6], Stanford University [7], University of Virginia [8], among others are creating multidisciplinary centers and offering various degrees related with *Data Science*. There is no doubt that *big data* is bringing many opportunities. However, many challenges related with various aspects of *data science* are also outlined in various studies [9–13]. For this research, we define *big data* in *life-sciences* as data *high in volume* (terabytes or larger), *too complex* (interconnected with over 25 highly accessed databases [1] and over 600 ontologies [14] that contain various types of data - from gene sequencing to cell imaging ), and *too dynamic* (growing exponentially [1, 15, 16]) for conventional data tools to store, manage, and analyze. Figure 1.1 shows the interconnected nature of the different life-sciences-data resources. One should note that the resources shown in the figure are only a partial list of tools and data resources available for life-sciences data (i.e. tools found in the European Bioinformatics Institute (EBI) repositories). As we take more tools and resources into account, we see these connections become a *hair-ball* graph very quickly. Figure 1.2 shows the exponential increase in the GenBank data over the years and how the data creation is still increasing.

In the era of *big data*, extracting and representing knowledge hidden in large amounts of scientific data has become a daunting task [1, 17]. This is only one of the challenges of big data. Big data challenges include dealing with increasing volume, securing the data, and creating the infrastructure that allows analysis, in addition to extracting knowledge from available data [10–12, 17]. Life-sciences data is not

only increasing in volume, but also fitting more into the description of big data as described above. In accordance with this, challenges specific to life-sciences data, in addition the general challenges mentioned above, arise. One of these challenges is: currently available complex life-sciences data is not being efficiently translated into a format that is *unambiguously* readable and understandable by machines and humans. Other difficult problems are: how to organize, how to standardize, and how to analyze the life-sciences data without flattening it (because, flattening diverse life-sciences datasets could lead to incorrect predictions). Provided good and feasible solutions to these problems, the vast amounts of data could reveal new knowledge and discovery.

As the life-sciences data grows, the need to build intelligent systems that will store, organize, and help scientists analyze the data is growing as well [1]. Ideally, in such systems, the computer and researchers will have an unambiguous understanding of what the data means. Furthermore, the computer system will allow the life-scientists to connect scattered pieces of information and help them acquire new knowledge, i.e. *inference of knowledge* that they didn't possess when building the system.

Building such systems can be accomplished by using semantic web technologies. In the past decade, many research efforts aimed at building such systems. As a result a significant number of ontologies have been built related with life-sciences (the number of ontologies on Bioportal in November 2016 was 529-, in November 2017 was 665-, in February 2018 was 690.). Although there exist some studies about different techniques and tools about building such technologies, there is still a lack of widely-accepted methodologies, best-practices rules, and tools that could help build effective technologies that address the challenges mentioned above.

The life-sciences do have a profound interest in building ontologies, although most of them are not widely used. Recently, commercial communities, such as drug companies, have developed a keen interest in them as well. The interest in ontologies is mostly for annotation purposes. The companies and researchers use the life-sciences related ontologies to annotate their assays, scientific papers, or even databases.

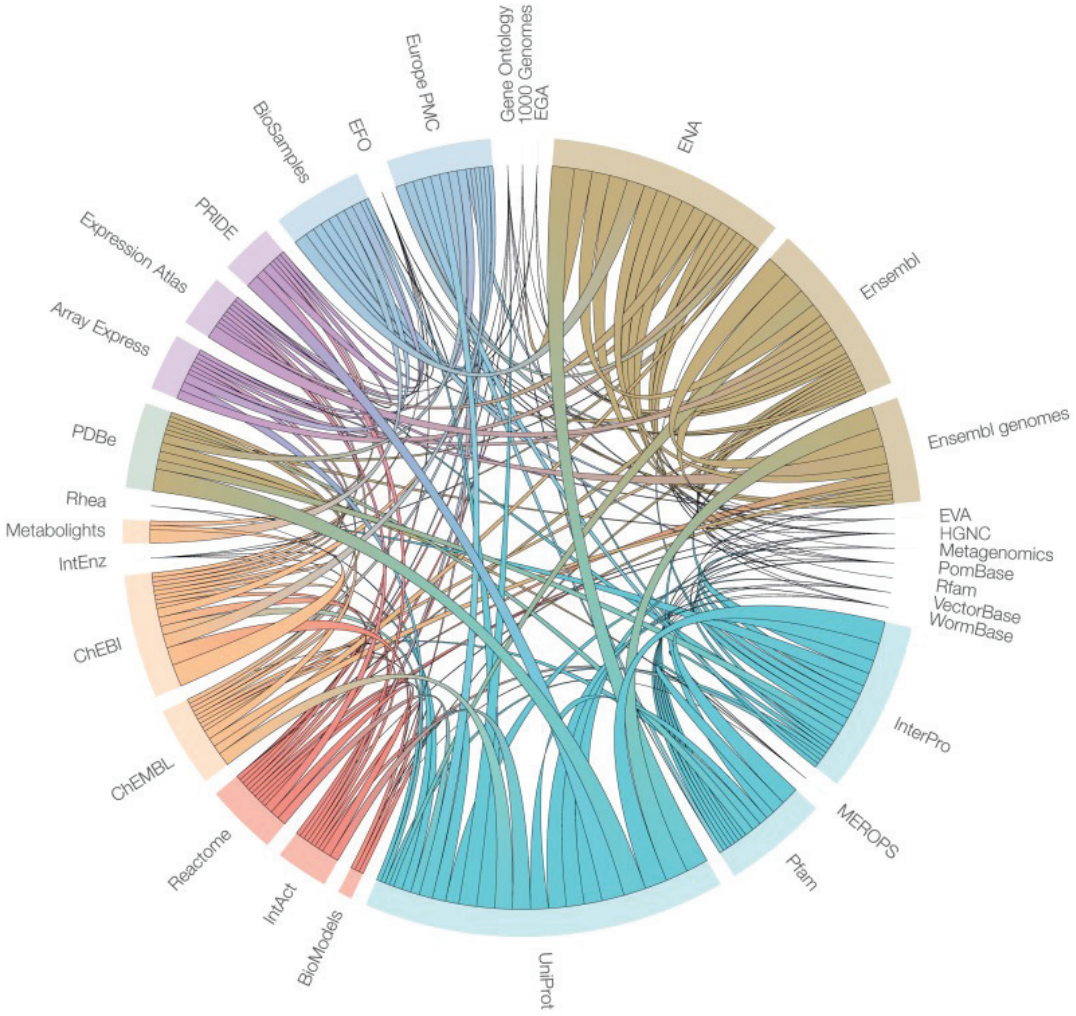



Figure 1.1: Data and connections among pieces of various data types in the European Bioinformatics Institute (EBI) [1] This figure shows the inter-connectivity of the different databases and different datasets. The figure is limited to the databases, ontologies, and other services provided by the EBI

In accordance with this, most of the ontologies built focus on creating controlled and/or standardized vocabularies (i.e. taxonomies), not necessarily the intelligent systems mentioned above. Thus, the majority of the methodologies for ontology building focus on creating taxonomies. Some existing tools and methods focus on using computers and databases alone for creating such taxonomies. While these taxonomies are very useful for certain types of text annotation (i.e. tagging human readable text with machine readable vocabulary) purposes, they fail to capture the depths of the life-sciences related knowledge, and fail to utilize the computational reasoning capabilities.

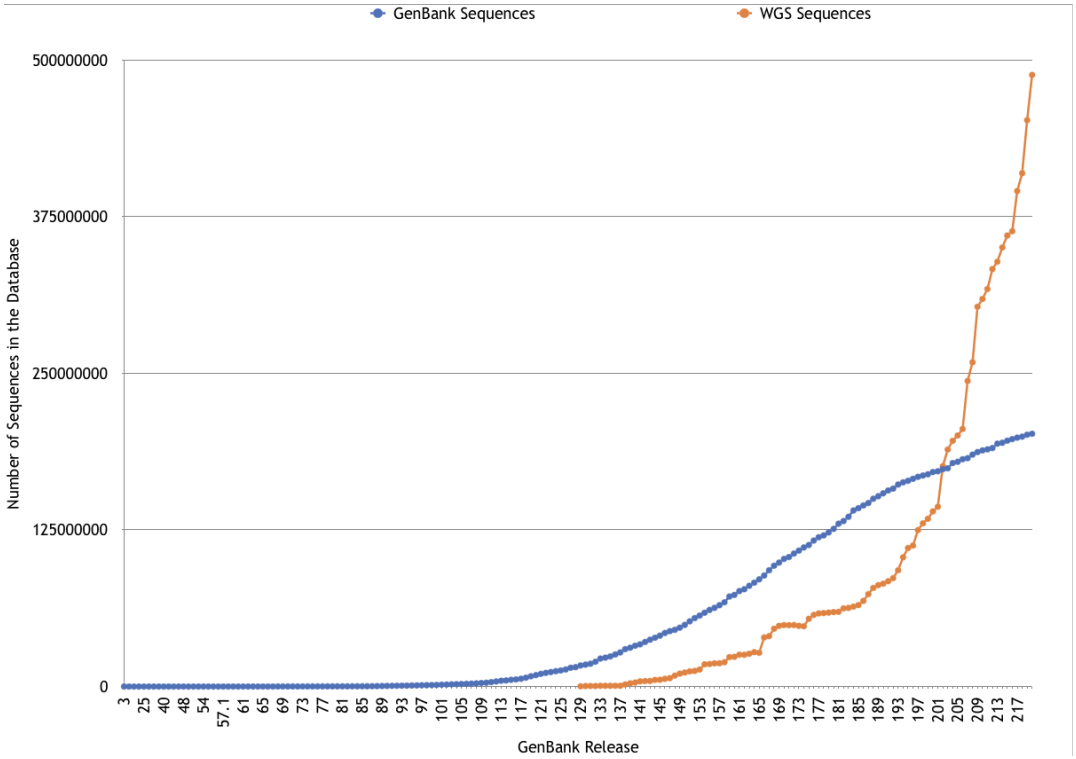



Figure 1.2: GenBank growth statistics (last updated July 2017) that show that there is an exponential increase in the data created

The majority of the ontologies available focus on hierarchical relationships among individual types of bio-molecules or life-sciences related concepts. For example, only proteins are linked together in Protein Ontology (PRO) [18], only diseases are linked together in Disease Ontology (DOID) [19], and only cell lines are hierarchically described in Cell Line Ontology (CLO) [20]. Some ontologies use a limited number of relationships (e.g. 'is a' and 'has a', and 'part of') to formally describe the data. Therefore, only limited aspects of the concepts can be formally represented. This then allows limited inferences about the data.

As mentioned above, the ontologies that provide only controlled vocabulary, or provide limited formal descriptions of data may be useful for annotating textual information such as assay descriptions and/or scientific papers. As useful as they are, they cause problems for the scientists at times. For example, the use of the same concept under different names results in ambiguity for the users. This repetition in the efforts not only causes ambiguity, but also costs time and effort for the users. The users spend time trying to identify equivalent concepts and create equivalency

relationships. If they fail to do so, then their search and analysis may not be comprehensive. Furthermore, many users might not know how to connect the existing ontologies as a part or module to their own system. Either way, having unlinked related data is a cost to the user. In addition to this, ontologies similar to the ones listed above, fail to aid complex computational inferences and/or machine learning applications because of their limited set of axioms.

In addition to the tools and methodologies that allow creation of taxonomies, there are also tools which aim to combine data in databases in RDF form. These tools make use of well known life-sciences databases in an attempt to merge scattered life-sciences data. However, they fail to integrate insights that can be provided by the domain experts in combining the available data. Moreover, the possible connections can only be made for a number of well known databases. Thus, these tools cannot be used with newly formed databases or semantic web applications.

In this study a new methodology, termed KNowledge Acquisition and Representation Methodology (KNARM) is proposed in order to help aid challenges summarized above and achieve better acquisition and representation of knowledge while avoiding over simplification. We propose this systematic, methodological approach utilizing description logic and semantic models that addresses the knowledge acquisition bottleneck. This methodology is created and used for this research project by combining available methods for Database Management Systems (DBMS), Object Oriented Programming (OOP), and Knowledge Acquisition Methods. It was designed, implemented, and used based on our needs and challenges related to our ongoing projects, namely the BioAssay Ontology (BAO), Library of Integrated Network-Based Cellular Signatures (LINCS) project, and Illuminating the Druggable Genome (IDG) projects.

KNARM is a hybrid methodology that combines human and machine capabilities for extracting knowledge and representing it in an ontology. It is designed to handle both new and existing knowledge/data and allows building ontologies with high expressivity. The knowledge representation uses axioms in a Systematically Deepening Modeling (SDM) approach for defining concepts in formal logic.

As mentioned above, KNARM, was created with three projects in mind: the BioAssay Ontology (BAO), Library of Integrated Network-Based Cellular Signatures (LINCS) project, and Illuminating the Druggable Genome (IDG) projects. All of the projects are nation-wide projects with data creation centers outside of the University of Miami, and are funded by National Institute of Health (NIH). Data from these projects is not only big in size, but also varies in types of data, from cell phenotype images to gene mutations to disease associations. Although the projects' aims are different, they are related (such as providing new and/or (more) effective therapeutic solutions to existing diseases) as well as some of the data involved with the projects, such as genes, proteins, and diseases, among other pieces of data. However, acquiring knowledge out of the available data has proven to be a difficult task, even for a few projects from cooperative sources. As we explored more to see what technologies and methodologies already exist, we observed that the need for a systematic methodology for knowledge acquisition and representation by building semantic web tools has been pointed out in the literature several times for more than two decades.

This dissertation aims to describe details of KNARM, and showcase how KNARM was used to design, implement, and update two major ontologies BioAssay Ontology (BAO) (designed and implemented for NIH funded BioAssay Ontology Project) and Drug Target Ontology (DTO) (designed and implemented for NIH funded Illuminating the Druggable Genome (IDG) project) as well as small application ontology LINCS FramEwork Ontology (LIFE) (designed and implemented for NIH funded LINCS project). This introduction is followed by a review of related research. We then describe details of KNARM's steps. After the detailed description of KNARM, details of its application over BAO, LIFE, and DTO are explained (with documents provided in the appendices). As results, we show use cases that utilize the three ontologies (BAO, LIFE, DTO) with their external ontologies (GO, DOID, BRENDA to name a few). We further exemplify how they work together in queries. We provide *proof of concept* results on how formal descriptions of life-sciences data may lead to new discoveries, and new leads on drug research and discovery by using the inference capabilities of semantic web technologies. We also describe the systematic patterns and ontology architecture we use to build the ontologies in order

to reuse parts of one another and work concurrently to help aid drug discovery. This dissertation ends with a discussion of how to improve current results and what future ideas could be implemented that could address challenges of *big data* and its knowledge acquisition bottleneck.

# CHAPTER 2

# Related Research

For this research, we reviewed existing knowledge acquisition methods, general ontology building methodologies, and reviews that combine them together. We focused more on specialized knowledge acquisition and ontology building methods and tools used for handling the life-sciences data. However, we also briefly review widely accepted methodologies such as CommonKADS [21]. The reviews are listed chronologically.

## 2.1 Reviews

The reviews can be viewed in two groups: The first group has relatively older papers whose authors define and describe the basics of ontologies and how to create one. The second group of papers are newer and they focus on the life-sciences related ontologies in addition to the issues related with them.

Stevens *et* al. [22] gives an overview of ontologies and how they can be used for bioinformatics applications. This paper can be viewed as a crash course in how to build ontologies written more than a decade ago. It aims to better help scientists with life-sciences background who want to make use of ontologies for their research and applications. The paper does not provide a novel approach or a new ontology, but a collection of existing approaches and applications. Ontology as a concept is introduced along with the justification for why they are useful in representing life-sciences data to be used. The authors give a brief introduction about how ontologies can be created. They also explain ontology related concepts such as

"*defined concepts* ", "*primitive concepts*", and different types of relationships, as well as axioms. Furthermore, they define different methodologies for building ontologies, such as the V-model inspired methodology. The authors emphasize how biology is rich in taxonomies. Therefore, ontologies can be used for different applications related with biology. Consistent with their approach, they focus on ontologies that aim to create controlled vocabularies more than ontologies that aim to generate logical descriptions of biological concepts and processes.They continue with stating that there exists a number of taxonomy-based ontologies. The paper also provides us a survey of early bio-ontologies at the time, which are about 15 years old now. They focus on RiboWeb [23], EcoCyc [24], and Gene Ontology [25]. Today, a collection of 568 (last access: July 2017) bio-ontologies can be found on BioPortal [26].

Being an early review, this study focuses on justification of using ontologies mostly as taxonomies. As useful as these taxonomies are, they don't provide the full capabilities of an ontology with axioms and mappings of multiple ontologies focusing on similar datasets.

Next, Wache *et* al. [27] summarized existing approaches in ontological integration of heterogeneous information sources in their review. The authors summarize how the use of ontologies effect their architecture, how the ontology representations can vary, how to approach ontology mapping challenges, and what are the current development methodologies. The review contains the different languages and tools used for ontology building, most of which are currently outdated. The tools included KRAFT [28], Ontobroker [29], SIMS [30], and SHOE [31].

KRAFT [28] is a tool that allows ontology building based on existing database schemas. As mentioned previously, this is a valuable automization, however conversion of database schemas lead to oversimplification. Furthermore, It focuses on vocabulary generation and doesn't allow integration of domain expert knowledge directly. Ontobroker [29] focuses on generating ontologies using a meta-data description provided by the users. It is mainly focused on generating vocabulary rather than knowledge based on domain experts' knowledge and axioms. SIMS [30] is a semantic platform that is designed to help databases communicate one another. SHOE [31] provides a set of simple HMTL extensions which allows the world wide web authors

to annotate their content. This review includes comparison of approaches for ontology evolution and determine that only the SHOE system accomplishes proper automated approach for helping update the ontology resources.

The authors summarize that the current systems use ontologies that aim to integrate their sources by using a similar structure. The common languages for the ontologies are based on description logics and the systems make use of subsumption reasoning for mapping of the ontologies. The building of the ontologies is performed using specialized editors, such as OntoEdit [32] and SHOE's Knowledge Annotator.

This review concludes that there is a striking lack of sophisticated methodologies for the development and use of ontologies. They suggest that such methodology should be independent of the languages they reviewed that may be used to build ontologies. They claim that a good methodology should also cover the evaluation and verification of the decisions made with respect to the language and the structure of the ontology. This conclusion that the authors' reached is still valid today. There is still a lack of sophisticated methodologies for development and use of ontologies. Furthermore, the problem got bigger for life-sciences as the number of available ontologies increased over the years. The problem also extended to include the question: how could we better share, reuse, and evolve these ontologies faster, better, and in a more automated way in order to provide better analysis opportunities.

Another one of the early pointers to the conceptualization of biological data was a Nature article [33]. Blagosklonny and Pardee define "conceptual biology" as the information in databases that are related to the life-sciences. They refer to the large number of databases with enormous amounts of biological data waiting to be decoded. The authors point out how all the biological systems are interconnected and the separation of the different biological systems is artificial. They suggest how semantic conceptualization of the data in the scattered databases could help connect data that is seemingly disconnected. They also point out how biology is a hypothesis-driven discipline. They claim that currently hypotheses are based on labor-based studies. However, one could benefit from the computational data available for the hypotheses. Additionally, the computer systems might suggest or infer new knowledge. As addressed by Barnes, following the Blasgosklonny and Pardee

paper, this process has a lot in common with drug discovery and related sciences by using the existing biomedical data [34] and has been very useful for drug-discovery related research.

As pointed out in this study, scattered life-sciences data keeps challenging research and analysis efforts. Furthermore, duplication of data in different resources with different identifiers brings about mapping and alignment problems involving the data. Although some practice guidelines exist for some communities us as OBO foundry [35] [36], there are no best practices approaches set.

Following that, in 2003, Corcho *et* al. [37] reviewed methodologies for ontology building and reviewed the existing methods such as Cyc [38], TOVE (TOronto Virtual Enterprise) [39], and finally METHONTOLOGY with the later generated by the same group [40]. In their survey, they focus on methodologies that are used for *building ontologies*. The authors review the above mentioned methods and the tools that they use. They then point out the common points and differences of the methods by creating a table.

Cyc has three phases. The first phase is to manually extract knowledge from resources. The second and third phases involve acquiring new knowledge using natural language or machine learning tools. The difference between the second phase and the third phase is that the second phase is aided by tools, but requires labor by humans, while the third phase the acquisition is mainly performed by tools.

TOVE applies a methodology inspired by the development of knowledge-based systems using first order logic. They take possible use cases as a starting point to identify the scope of the ontologies and the concepts of the ontology. Though it is a robust method which takes advantage of the classical logic, it is not a fit method for the era of big data.

METHONTOLOGY describes the ontology building process in detail. It has three main steps: The identification of the ontology development process, a life cycle based on evolving prototypes, and particular techniques to carry out each activity. The ontology development process involves identifying tasks that should be performed when building ontologies (scheduling, control, quality assurance, specification, knowledge acquisition, conceptualization, integration, formalization, im-

plementation, evaluation, maintenance, documentation and configuration management). The life cycle step decides on the stages through which the ontology passes during its lifetime, as well as the inter-dependencies with the life cycle of other ontologies. The methodology also specifies the techniques used in each activity, the output products that each activity and how they have to be evaluated.

Their evaluation aligns with the evaluations of Wache *et* al. [27]. They also emphasize the lack of methodologies for ontology building and evolution. However, this work is more focused on a few tools and includes only the ontology building process, rather than focusing on all the tasks related with ontologies such as ontology evolution, ontology re-engineering, etc.

In the review, the authors go over the steps for each methodology briefly. The main common point of the above listed methodologies is that they are domain independent and not designed for collaborative development. They all follow a similar fashion as a software engineering methodology would follow. They suggest that the approaches from different groups should be combined into a single methodology. They also talk about the different languages, SHOE, RDF, RDFS, and OWL [41] and their different expressiveness and reasoning capabilities. As pointed out by many before them, they mention the importance of choosing a language for your ontology. It is now widely accepted that choosing a language and expressiveness level to build an ontology is one of the most important steps. They also write about ontology building tools of the time, such as Protégé [42] and Onto Edit [32], and they claim that they are necessary tools for the trade. Although this review makes good points about methodology building and re-purposing existing software methodologies, the tools and methods mentioned are now outdated.

A more genomics-data-focused review was performed by Louie et *al.* [2]. They call the mosaic of life-sciences as 'genomic medicine' and they include many different disciplines - such as biology, chemistry, medicine, marine biology - in the mosaic. Today we call the same data as life-sciences data. The authors also point out that due to the dynamic nature of the data (rapidly growing and changing), it is hard to encompass the entire scope of the data. Today, this observation still holds especially with the introduction of new types of life-sciences data available, such as RNA se-

quencing data. The review attempts to discuss how the data integration related to biomedical informatics should be performed, what the challenges and future opportunities were, as well as what tools would bring the most out of the different kinds of biomedical data.

Their main focus points were the representation of data suitable for computational inference (knowledge representation) and linking heterogeneous data sets (data integration) for the life-sciences data. They point out that existing solutions for data related problems can also be applied to life-sciences data. They claim that by identifying the existing data integration techniques, and where your data lies on an imaginary plane defined by two axes, you may find a proper approach to better integrate life-sciences data (As shown in Figure X).

Their imaginary plane is defined by two orthogonal axes where the biomedical data and the metadata resides, and the representation of the data and data models (Figure 2.1). The two axes are: (1) integration architecture, i.e. where should your data live: databases, data warehouses, peer data management systems, etc. and (2) data and knowledge representation, i.e. relational database schema, semi-structured data, ontology, etc. The review points out the pros and cons for each different architecture and knowledge representation method.

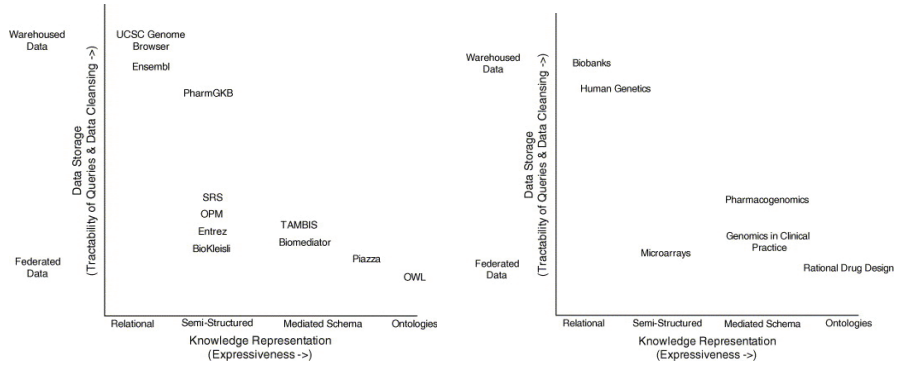



Figure 2.1: Left: Different Tools and Their position on the axes, Right: Data Storage and Knowledge representation options for different problems. By identifying the existing data integration techniques, and where your data lies on an imaginary plane defined by two axes, you may find a proper approach to better integrate life-sciences data. The two axes are: (1) integration architecture, i.e. where should your data live: databases, data warehouses, peer data management systems, etc. and (2) data and knowledge representation, i.e. relational database schema, semi-structured data, ontology, etc. [2].

For the first axis, the authors review the databases, database federations, and Peer Data Management Systems (PDMS). They argue that keeping your data in

smaller databases can help you query faster, but it may cause your data to go stale after a while. Database federations, such as BioMediator [43], might have longer turn around time for queries; however they have access to more current data. Finally the authors mention the pros and cons of Peer Data Management Systems (PDMS). The PDMS doesn't have a complex schema like a database federation would have, and it provides a more flexible architecture. However, it is still mostly experimental and may provide slow queries.

For the second axis, the authors review relational schemas, semi-structured data, and ontologies. For the relational schemas, they review the traditional database systems. They point out that the traditional databases need precise relationships among entities, however currently, the relationships among biological concepts are not always precise. In addition to their observation, we can say that database systems only represent the given information and lack the open-world assumption, i.e. there may be more information inferred from the current data. Since currently, most of the biological processes and experiments are treated as a "black-box"[1]. Therefore, having a close-world assumption fairly limits our ability to model the exact nature of the processes and experiments. Despite this, database systems have been the most common and familiar mediums for storing the life-sciences' data. Semi-structured data representation languages such as XML and RDF (in contrast to databases) release us from trying to find rigid relationships among entities. However, one of the most limiting aspects of XML is lacking the many to many relationships. However, for life-sciences data, the ability to create many-to-many relationships is crucial in modeling concepts such as pathways. For example, the PharmGKB uses XML for its efforts to create a pharmacology related knowledge base [44] Finally Louie et *al.* describe ontologies as "specification of conceptualization". They refer to the ontologies built in OWL and they state that ontologies represent knowledge in a computer readable format. They mention how this allows us to utilize computers in various ways such as complex queries and knowledge inferences. They also state how

[1]Because of the design of the experiments and technological shortcomings, many of the biological and chemical processes happening in the cell and in the body have various unknowns. Therefore, we keep records of the perturbations and the final changes in systems, however the details of how the changes occurred is usually unknown, i.e. a black box.

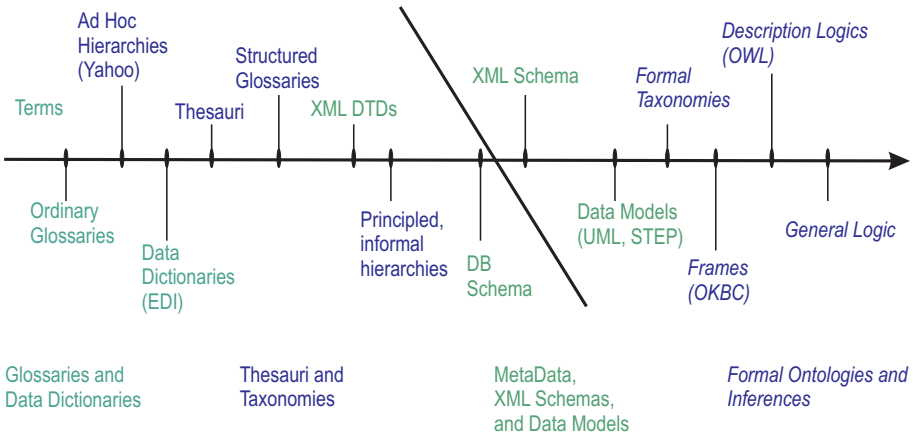



Figure 2.2: Kinds of ontologies [ [3] p. 10, changed]. This figure shows the different languages and different kinds of ontologies that can be created. As seen in the figure, Description Logic (DL) is the most expressive language before general logic and formal taxonomies are not as expressive as ontologies built using DL.

different ontologies can be combined together which would allow a larger amount of inferences and discoveries. In figures they also show that ontologies built using OWL are the most expressive representation of knowledge (cf. Figure 2.2 ).

The authors finish their review with suggestions about what architectures should be used for different types of data. They also address different types of open questions and emerging research. They further state some of the current issues, such as not having standards for data collection and representation which is a major problem stated by many before them.

A very comprehensive review of Knowledge acquisition (KA) methods in biomedicine is performed by Payne et *al.* [45]. In this paper the authors review different KA methods from different perspectives, for example KA according to education, and cognitive basis of KA and finally scenarios about how that existing KA methods can be used in biomedical projects.

They define conceptual knowledge as the atomic pieces of information and the relationships among them. They then focus on the conceptual knowledge acquisition (CKA) in biomedicine as it applies to life-sciences. They acknowledge that knowledge acquisition (KA) and conceptual knowledge acquisition (CKA) are parts of a larger domain called knowledge engineering (KE), beyond the scope of these research projects. There are four basic steps of knowledge engineering:

1. Acquisition of knowledge (KA)

2. Representation of knowledge in a computable form

3. Implementation or refinement of applications that use the knowledge representation created in the previous steps

4. Verification and validation of the tools and knowledge representation

They further elaborate on how the cognitive science literature describes two types of knowledge: procedural and declarative knowledge. Procedural knowledge is a process-oriented understanding of a given problem domain. Declarative knowledge is largely synonymous with conceptual knowledge mentioned above, i.e. a combination of atomic units of information and the meaningful relationships between those units. However, declarative knowledge consists solely of "facts" without any explicit reference to the relationships that may exist between those "facts".

Payne et *al.* review the education literature which defines conceptual knowledge as a combination of atomic units of information and the meaningful relationships between those units. The education literature also describes two other types of knowledge, procedural and strategic. Procedural knowledge is definition stays as a process-oriented understanding of a given problem domain in this context as well. Strategic knowledge is used to operationalize conceptual knowledge into procedural knowledge.

Finally, the authors mention that another significant literature for KA can be found in the computer science (CS) literature, especially regarding artificial intelligence. They argue that the CS literature is more focused on procedural knowledge, including those used in a large number of intelligent agents and decision support systems. They also point out that artificial intelligence literature is extremely sparse with respect to KA methods intended to elicit conceptual knowledge.

The authors also look deeper into how cognitive science, psychology, and programming languages affect knowledge acquisition methods and taxonomy creations. They provide KA techniques from the existing literature and advise on how to best use the different techniques. The methods reviewed in their paper can be summarized as follows:

**Informal and Structured Interviewing:** Interviews conducted either individually or in groups can provide investigators with insights into the knowledge used by domain experts.

**Observations:** Observations generally focus on the evaluation of expert performance, and the implicit knowledge used by those experts by observing them perform tasks.

**Categorical Sorting:** A number of categorical, or card sorting techniques have been developed, including Q-sorts, hierarchical sorts, all-in-one sorts and repeated single criterion sorts. The different sorting methods can be applied to the concepts and how they should be sorted in the taxonomies created.

**Repertory Grid Analysis:** Repertory grid analysis is a method based on the Personal Construct Theory (PCT). PCT argues that humans make sense of the "information world" through the creation and use of categories. Repertory grid analysis involves the construction of a non-symmetric matrix, where each row represents a construct which corresponds to a distinction of interest, and each column represents an element (e.g. unit of information or knowledge) under consideration.

**Formal Concept Analysis:** Formal concept analysis (FCA) has often been applied to the tasks of developing and merging ontologies. FCA is almost exclusively used for eliciting the relationships between units of information or knowledge. FCA can be automated using different algorithms. When FCA is performed using automated methods, large-scale KA studies are feasible. However, FCA techniques are limited to the discovery of relationships between conceptual entities, i.e. data already available in databases, rather than the entities themselves. Therefore, other KA techniques must often be applied prior to FCA to determine a corpus of entities and attributes.

**Protocol and Discourse Analysis:** The techniques of protocol and discourse analysis are very closely related and similar to observation. Both techniques elicit knowledge from individuals while they are engaged in problem-solving or reasoning tasks.

**Sub-Language Analysis:** Sub-language analysis is a technique for discovering units of information or knowledge, and the relationships between them within

existing knowledge sources, including published literature or corpora of narrative text.

**Laddering:** Laddering techniques involve the creation of tree structures that hierarchically organize domain-specific units of information or knowledge. Laddering is another example of a technique that can be used to determine both units of information or knowledge and the relationships between those unit.

**Group Techniques:** Several group techniques for multi-subject KA studies have been reported, including brainstorming, nominal group studies, DELPHI studies, consensus decision making and computer-aided group sessions. All of these techniques focus on the elicitation of consensus-based knowledge. While consensus-based knowledge is arguably superior to the knowledge elicited from a single expert, conducting multi-subject KA studies can be difficult due to the need to recruit appropriate experts and logistical challenges involved in assembling the experts.

The authors also elaborate on verification and validation step of the KE. They define verification as the evaluation of whether a knowledge-based system meets the requirements of end-users established prior to design and implementation. Validation is defined as the evaluation of whether that system meets the realized (i.e. "real-world") requirements of the end-users after design and implementation. They point out that during verification, results are compared to initial design requirements, whereas during validation the results are compared to the requirements for the system that are realized after its implementation. They acknowledge that validation and verification work in parallel, and that verification would address the internal validity of the knowledge collection, while validation would address the external validity of the knowledge collection.

The authors provide a great review for identifying relevant concepts and their relationships. However, they do not provide any methods that will allow the users to model and create a knowledge base using the concepts identified with the help of knowledge formalization, such as using axioms.

CommonKADS [21] has been the widely accepted and used structured knowledge engineering methodology since it was introduced over thirty years ago. CommonKADS aims to provide solutions for all aspects of knowledge management,

knowledge analysis, and system development. It provides several different solutions for problems related with knowledge engineering. However, because of the breath of its focus, it fails to give a set of steps that can easily be followed to build practical knowledge bases from scratch.

CommonKADS does provide different approaches for knowledge acquisition. Although most of the principles introduced still hold today, CommonKADS approach builds around how to perform *interviews* with the domain experts in order to understand the data. Additionally, knowledge acquisition and representation advice is mainly about understanding whether some data is a concept, a relationship, or an attribute. It does provide the methods to identify the different concepts lie in different hierarchies and different philosophies for representing data can be used for this purpose. Therefore, like all of the above mentioned methods, CommonKADS is also about how to create taxonomies, rather than creating formal representations and models of data using axioms.

We can say that the scope of CommonKADS is larger than what we are presenting with this research. Furthermore, CommonKADS aims to help a broader and larger audience with all aspects of Knowledge Engineering. KNARM (KNowledge Acquisition and Representation Method) that we are introducing in this study, on the other hand, currently focuses on life-sciences related knowledge acquisition and representation.

We should also note that CommonKADS is almost thirty years old. Today, with the advances in computational speed and terrabytes of data created, we acquire, formalize, and represent data for various new purposes easier. Current projects are more interested in efficiency in acquiring and representing the knowledge. With increased inference capabilities, we aim to better utilize computers for knowledge acquisition and representation. Moreover, as opposed to CommonKADS' interview heavy approach, we are trying to identify and generate ways to automate most of the formalization process so that we can update our knowledge bases frequently due to the rapid changes in knowledge.

All the methods reviewed in this section lack techniques needed to review and formally represent existing knowledge related with life-sciences data. These meth-

ods aim at creating controlled vocabularies and taxonomies for life-sciences related terminology. However, they fail to capture the knowledge using formal logic. Furthermore, they do not offer well-defined set of steps that are useful to create a new knowledge base or ontology from scratch. Most importantly, these methods do not offer solutions for the fast evolving data and data structures. New methods and fast evolving ontologies are needed to represent the new and growing data created by life-scientists. Table 2.1 on page 21 lists all the different methodologies reviewed in this study.

## 2.2 Tools

In the previous section, we have reviewed the methods and reviews of the existing methods related with semantic web applications for managing the evolving life-sciences data. This section focuses on existing tools that aim to utilize semantic web technologies for life-sciences data.

One of the first approaches to conceptualizing biology was by Gottgtroy and colleagues [46]. They created Neucom and a primitive ontology that deals with biomedical data. Although the ideas they had are exciting, both Neucom and the ontology they created are primitive and fail to generate practical and useful application examples.

Their main idea is to use existing databases to create gene-disease maps [46, 47]. They also summarized how the ontologies could be engineered to aid ontology learning for the evolving domain of biology. They propose to create ontologies that would infer new knowledge and feed it back to the ontologies. Their approach also involves using existing databases and database schemas to build ontologies in an automated way. Integration of different databases and database schemas is a desired outcome. This very problem has been handled by various different groups, e.g SEMEDA, Bio2RDF (reviewed below), in different ways in the past two decades. However, they fail to notice that integrating existing databases is not the only problem with

| Projects | Knowledge Unit Elicitation | | | Knowledge Relationship Elicit. | | | Combined Elicit. | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Interview | Structured Interview | Observation | Categorical Sorting | Reporotory Grid Analysis | Formal Concept Analysis | Protocol Analysis | Disclosure Analysis | Sub-language Analysis | Laddering | Grouping Techniques |
| OBO | x | x | x | ✓ | x | ✓ | ✓ | ✓ | ✓ | x | ✓ |
| SEEK | x | x | x | ✓ | x | ✓ | x | x | x | x | x |
| CommonKADS | ✓ | ✓ | x | ✓ | x | ✓ | x | ✓ | ✓ | ✓ | |
| Bio2RDF | x | x | x | ✓ | ✓ | x | x | x | x | x | x |
| METHONTOLOGY | ✓ | ✓ | x | ✓ | x | ✓ | ✓ | x | x | x | |
| PharmGKB | x | x | x | ✓ | x | x | ✓ | | x | x | ✓ |

Table 2.1: Usage of Knowledge Acquisition Techniques and Methodologies Reviewed

conceptualizing the biological data. Another problem is conceptualizing new data coming from new experiments.

Previously mentioned Protégé [48] was also introduced in the context of designing and implementing the Gene Ontology (GO) [25]. GO has been one of the most widely used biological ontologies. Like all ontologies, it grows and needs to be maintained. Therefore knowledge acquisition is essential for maintaining GO. However, in their paper about the Knowledge Acquisition of GO [49] the authors only specify what new concepts and relationships were introduced. They do not provide details on how they performed the extraction of human and literature knowledge nor how they translate the knowledge into logical axioms. They only point out that they have to translate the knowledge under the right tree of the taxonomy. This task was covered by CommonKADS very elegantly. While being widely used and very useful, GO does not provide complex relationships to define the concepts it is representing. For the purposes of GO, the two relationships ("is a" and "part of") that it uses are enough to represent their set of vocabulary. There is no doubt that the GO taxonomy is very useful. However, we should note again that it does not deal with complex formalization of the data. GO is one of the groups that advocate using fewer relationships in ontologies. Some other OBO foundry ontologies also agree with them. Their aim is to simplify the structure of the ontologies and their underlying graphs. However, the life-sciences data is too complex to represent using two relationships. Furthermore, in cases of inconsistencies and misrepresentations, having very few relationships would cause inference and reasoning leading to incorrect knowledge and other problems, such as not representing essential pieces of information.

Another tool called SEMEDA for database integration using semantic web was introduced by [50]. SEMEDA is designed as a three-tier system which aims to allow users transfer the relationships and attributes that they have for their databases into an ontology. It consists of a relational database backend (Oracle 8i or newer) to store ontologies, database metadata and semantic database definitions. Java Server Pages are used in the middle tier to dynamically generate the HTML frontend. Authors mention that SEMEDA"s architecture allows handling of large controlled vocab-

ularies and ontologies with a virtually unlimited number of concepts. SEMEDA retrieves information from different databases and then uses its own custom ontology for database integration. This created ontology is a small top level ontology, which defines databases at the schema level. SEMEDA uses basic Semantic Web practices in order to transfer the data and the structure of a database to an ontology. Like all of the tools available today, SEMEDA is focusing on integrating different databases with the help of alignments for concepts and attributes from the different resources it uses. However, it doesn't offer a solution for handling plain textual data and/or new data that does not exist in databases. It also doesn't provide a workflow for aligning the attributes and concepts from different resources.

One of the highly cited works is the Open Biomedical Ontologies (OBO) Foundry paper [51] in 2007. The OBO foundry aims to overcome the disconnected nature among the existing biomedical ontologies. As the authors pointed out, the biomedical ontologies increased in size and changed shape dramatically between 2003 and 2007 and they are still increasing in size and changing in shape since that time. The OBO foundry aims to align the efforts for ontology building and connect the different ontologies to allow information exchange. The OBO foundry designed an ontology building language, OBO, for the purpose of easily integrable ontologies. However, the OBO language that has been built to incorporate the different ontologies is not as expressive as some of the other ontology building languages, such as OWL2DL [41]. Therefore OBO is almost obsolete within 10 years. Most of the OBO foundry ontologies mainly contain "is a" and "part of" relationships, and are not as expressive as some other ontologies. For example DL expressivity for the Disease Ontology (DOID) [19] is AL while the expressivity for BAO 2.0 is SOIQ(D). Although it is known and agreed up on that biology is a taxonomy rich discipline, we also think that building taxonomies alone fails to capture the complexity of the life-sciences.

OBO foundry and OBO foundry ontologies focus on finding standardized vocabulary for life-sciences rather than formal definitions of the concepts. As mentioned before, while the standardized vocabulary is important for several purposes, we think that semantic web technologies could provide more insight to life-sciences. In this

paper the authors review the ontologies in their early versions and explain that they are going under revision about a decade ago. Today, most of the ontologies mentioned in this paper have a version in OWL. However, there are no major changes in the expressivity levels of the ontologies, i.e. they are still standardized vocabularies.

The OBO foundry authors concluded by stating their long-term goal as generating a system that allows the data generated through biomedical research form a single, consistent, cumulatively expanding and algorithmically tractable whole. They acknowledge that their goal, may affect the flexibility and advancement of the sciences. We can further argue that the rigidly structured and centralized efforts do not take the dynamic nature of the life-sciences into account. Today, many different biological assays and life-sciences related tools and molecules have been added to the literature compared to ten years ago. With the advances in technology, we can predict that the life-sciences data will keep growing and evolving. Therefore, tools, methodologies and standards that can handle the dynamic nature of the data are crucial.

The need to integrate scattered ontologies has also observed by the Bio2RDF team [52] in 2008. Like the OBO foundry, it is an open source effort that aims to create on demand knowledge base (KB) views. However, while OBO foundry focuses on creating ontologies from scratch and creating standardized vocabulary for life-sciences, Bio2RDF focuses on integrating existing database resources to create different knowledge views in RDF. Bio2RDF is written in JSP that perform translations to RDF. It further utilizes the Sesame open source triplestore and OWL technologies. Documents from highly cited databases such as KEGG, PDB, and several NCBI databases can be made available in RDF file using the Bio2RDF tools.

The Bio2RDF team also recognizes the existance of tools such as SEMEDA that aim to integrate bioinformatics related data. However, they also point out that SEMEDA fails to make use of it. Furthermore, they claim that OWL is becoming the standard language for life-sciences ontologies. A decade after this publication, we can confirm this claim by looking at over 500 ontologies at the Bioportal. Consequently, the authors list the lessons learned from sparse projects which they performed aim to integrate partial life-sciences related data. The main lesson is that

utilizing the semantic web approach for life-sciences data integration is currently the most effective approach. Furthermore, integration efforts of different data sets for several projects aim to answer specific questions for that particular project. Therefore the integrated data is not comprehensive, i.e. not all data from all databases is integrated. Last, but not least, Bio2RDF team points out that using and encouraging to use open-source tools is crucial for more comprehensive integration and analysis purposes.

The authors describe the Bio2RDF applications by consolidating several examples of databases that they integrated by using their RDF versions. They conclude by stating that Bio2RDF is still in progress, like all ontology related tools. They also stated, in the future they planed to apply algorithms to see how the underlying graphs of the databases might be related. However, this is an NP-complete problem, and is still work in progress.

In summary, the Bio2RDF effort focuses on bringing together the diverse databases and ontologies and creating a triple store using the different resources. They are also concerned with the scalability, performance, and re-producibility of the results. These are concerns that the computational biology community shares and agrees upon. Like Bio2RDF, the data in life-sciences as well as the applications related with them, are dynamic, changing everyday. Bio2RDF is not concerned with creating new concepts and/or ontologies. It also does not provide solutions for integrating new information, i.e. new data available outside of publicly available databases, into the ontologies.

One of the most recent attempts to standardize the handling of biomedical data was the SEEK platform [53].The SEEK is a web-based resource for sharing and exchanging Systems Biology Data and models that are used by the JERM (Just Enough Results Model) ontology. The SEEK platform takes the different data standardizations, such as the MIBBI (Minimum Information for Biological and Biomedical Investigations) guidelines. Wolstencroft et *al.* [53] point out that it is becoming more important to collect and annotate data in a more standardized way. This is a widely accepted argument at this time and many attempts have been made to standardize the large amounts of systems biology data. They also point out the

diverse nature of the data and the need to correctly describe and link the different types of data by using mathematical models. However, most of the data in the SEEK platform"s *metadata*" are not useful for formally describe how the assays are performed. Important information about the assays and the assay participants are collected by the SEEK platform. However, the data and JERM ontology fails to model the diverse types of assays that deal with different types of data, by using axioms, in order to model the ontology using description logic. The SEEK platform collects the assay description in text, but do not offer solutions on how to handle the textual assay descriptions into semantic knowledge.

The latest research related with better ontology building practices includes a set of steps generated by He et *al.* called XOD (The eXtensible Ontology Development principles) [36]. XOD focuses on ontology creation tools implemented by He Lab and tools provided by the OBO foundry collaborators. There are four principles of XOD are:

1. **Ontology Term Reuse :** This step proposes reusing terms from "reliable" ontologies from a registered OBO foundry ontology. The proposed version of term reuse heavily relies on He Lab's tools such as OntoFox [54].

2. **Ontology Semantic Alignment :** The second principle involves aligning imported ontology terms and newly added terms with the same or compatible semantics. They propose to achieve such alignment by reusing the same object properties for axioms involving imported classes. Thus, they propose to import axioms of a concept.

3. **Ontology Design Pattern- Based Ontology Development :** This step relies on having design patterns and OBO tools for a recurrent ontology design problem. In case of a new design pattern, one can add it using OBO tools. The process follows three steps: (1) entering new terms / annotations to a form on the OBO tools, (2) converting the form to an Excel sheet, (3) converting excel sheet to OWL files.

4. **Community Extensiblity :** XOD recommends that a broad community of users and developers join the ontology building process. While this might lead to widely accepted term generation, XOD agrees that this process and this step could be the biggest bottleneck in the ontology building process.

XOD methodology focuses on building vocabularies and reusing existing ontologies using its principles. It relies heavily on OBO foundry tools and communities and doesn't offer any solutions to extracting knowledge out of the vast amounts of data. Although the principles address important questions about ontology building, the solutions provided are limited to the OBO foundry community. Furthermore, they fail to address questions that involve long textual data and integration of domain experts' knowledge with existing tools. Finally, they don't offer a solution or a workflow for building an ontology from scratch.

Table 2.2 lists above mentioned tools and Table 2.3 summarizes the methods and tools reviewed in this study.

Above mentioned studies and all the current efforts focus on how to integrate the existing databases. Moreover, many of these current projects lack a concrete plan for evolution of their systems, although they all agree that the available biological data is multiplying and evolving in alarming levels. Last but not least, all these studies listed in the table ignore the RDF integration of new experimental data that usually comes in textual form.

## 2.3 Review of Related Life-Sciences Studies

In the previous sections we described the methods and tools that aim to utilize semantic web technologies for storage, interpretation, and representation of life-sciences data. In this section, we briefly mention relevant life-sciences studies that create(d) data used in or related to our current research studies.

Biological components such as genes, proteins, metabolites and other molecules work together in harmony within cells to promote growth and development of living beings. Understanding how these interconnected components of biological pathways

| Projects | Knowledge Unit Elicitation | | | Knowledge Relationship Elicit. | | | Combined Elicit. | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Interview | Structured Interview | Observation | Categorical Sorting | Reporotory Grid Analysis | Formal Concept Analysis | Protocol Analysis | Disclosure Analysis | Sub-language Analysis | Laddering | Grouping Techniques | |
| RiboWeb | x | x | x | ✓ | x | x | x | x | x | x | x | |
| EcoCyc | x | x | x | ✓ | x | x | x | x | x | x | x | |
| OntoBroker | ✓ | | | ✓ | x | x | ✓ | x | | | | |
| BioMediator | x | x | x | ✓ | x | x | x | x | x | x | | |
| SEMIDA | ✓ | | | | | ✓ | ✓ | ✓ | | | | |
| SHOE | ✓ | | | | | ✓ | ✓ | ✓ | | | | |
| OntoEdit | ✓ | | | | | ✓ | x | x | | | | |

Table 2.2: Usage of Knowledge Acquisition Techniques and Projects Reviewed

and networks is one of the biggest challenges of this era. Furthermore, a great amount of research is being performed to understand how these components work under various perturbations, genetic and/or environmental stressors, which may cause disease or changes in systems. Last but not least, how the changes in molecular and system levels effect phenotypes of the living organisms. This effort to understand the underlying natures of interactions and changes in systems and phenotypes is then used to develop new and/or better therapies to return perturbed systems to their normal state.

Today, most pharmaceutical companies use High-Throughput Screening (HTS) the primary engine driving lead discovery [55]. However, with the increased ability in combinatorial techniques and advances in molecular biology better targets for therapeutic intervention can be provided. BAO is aimed at helping with that goal and is primarily designed to formally describe HTS assays.

High-throughput screening (HTS) has evolved into an industrialized process and HTS of small molecules is one of the most important strategies to identify novel entry points for drug discovery projects.

Until about half a decade ago, HTS and ultra-high throughput screening (uHTS) has been primarily part of the pharmaceutical industry. In 2003, National Institute of Health (NIH) started to make HTS and uHTS capabilities accessible to public sector research via the Molecular Libraries Initiative to advance translational research and specifically the Molecular Libraries Program (MLP) [56]. MLP projects aim to use various assays to develop compounds effective at alternating biological processes and/or disease states. The program has established publicly funded screening centers along with a common screening library (the MLSMR, Molecular Libraries Small Molecule Repository) and data repository, PubChem [57].

Since 2004, the MLPCN centers have deposited over two thousand HTS assays testing the effects of several hundred thousand compounds. More recently a European effort, EU Openscreen [58], to establish small molecule screening capabilities is being developed. Several other publicly accessible resources of screening data exist, for example ChEMBL [59], a database that contains structure-activity relationship (SAR) data curated from the medicinal chemistry literature [60], the Psychoac-

tive Drug Screening Program (PDSP), which generates data from screening novel psychoactive compounds for pharmacological activity [61], or Collaborative Drug Discovery (CDD), a private company enabling drug discovery research collaborations [62].

The rate of assay submission to PubChem and other repositories show that there is a vast amount of textual data being produced in addition to the data being added to the various databases for proteins and genes.

In addition to the MLP program, many other NIH funded projects started creating different assay types. One of the largest efforts is the NIH LINCS and BD2K projects that create various assays that are interrelated.

The underlying premise of the LINCS program is that disrupting any one of the many steps of a given biological process will cause related changes in the molecular and cellular characteristics, behavior, and/or function of the cell – also known as the cellular phenotype. A cellular phenotype is, in turn, intended to reflect signatures derived for comparable assays of clinical states. Observing how and when a cell's phenotype is altered by specific stressors can provide clues about the underlying mechanisms involved in perturbation and ultimately disease.

To achieve this goal, the Library of Integrated Network-based Cellular Signatures (LINCS) program is developing a "library" of molecular signatures, based on gene expression and other cellular changes that describe the response different types of cells elicit when exposed to various perturbing agents, including small bioactive molecules. High-throughput screening approaches are used to interrogate the cells and mathematical approaches are used to describe the molecular changes and patterns of response. LINCS data are openly available as a community resource for researchers to address a broad range of basic research questions and to facilitate the identification of biological targets for new disease therapies. The LINCS program was implemented in two parts. The pilot phase took place from 2010-2013 and focused on the following activities:

- Large-scale production of perturbation-induced molecular and cellular signatures.

- Creation of a database, common data standards, and a public user interface for accessing the data.
- Computational tool development and integrative data analyses.
- Development of new cost-effective, molecular and cellular phenotypic assays. Integration of existing datasets into LINCS.

The current phase of the program began in 2014 and builds on what was learned from the pilot. This phase of the program consists primarily of LINCS Data and Signature Generation Centers. The Centers carry out the following activities:

- Generating public datasets of cellular signatures collected in response to treatment with perturbing agents.
- Developing tools to optimize the accessibility and utility of their data.
- Organizing outreach activities with the broader research community so they can make use of LINCS data and tools.

The current phase of the LINCS program also works in synergy with the NIH Big Data to Knowledge (BD2K) program through a BD2K-LINCS-Perturbation Data Coordination and Integration Center (DCIC) at the Icahn School of Medicine at Mount Sinai. Read a description of the DCIC project on the NIH Common Fund's BD2K Funded Research page.

The LINCS project, in contrast to traditional screening, generates extensive signatures of cellular responses consisting of thousands of results for any perturbation (such as small molecule drugs) to enable the development of better system-level disease models. Examples of LINCS screening results and assays include Landmark gene expression signatures (L1000), Kinome-wide binding affinities (KINOMEscan), phenotypic profiling across 1,000 cell lines, and many others, covering "omics" and HTS data. LINCS results are currently available via participating centers and can be queried and explored via the LINCS Information FramEwork (LIFE) developed by our group [63].

In addition to the assay data being generated with the MLP, LINCS, and BD2K projects, the efforts to provide better therapeutics on molecular levels is also addressed as a challenge by the NIH.

The NIH funded IDG (Illuminating Druggable Genome) initiative is one of such efforts. The IDG project is to evaluate, organize and prioritize the potential disease-linked drug targets based on available data, knowledge and algorithms, for four protein families: G-protein -coupled receptors (GPCR), nuclear receptor (NR), ion channels (IC) and kinases.

The IDG drug targets are categorized as four super families with respect to the depth of investigation from a clinical, biological and chemical standpoint [64]:

1. **Tclin** (i.e. clinical) are targets for which a molecule in advanced stages of development, or an approved drug, exists, and is known to bind to that target with high potency.

2. **Tchem** (i.e. early stage) are proteins for which no approved drug or molecule in clinical trials is known to bind with high potency, but which can be specifically manipulated with small molecules in vitro.

3. **Tbio** are targets that do not have known drug or small molecule activities that satisfy the Tchem activity thresholds, but were the targets annotated with a Gene Ontology Molecular Function or Biological Process with an Experimental Evidence code, or targets with confirmed OMIM phenotype(s) [43].

4. **Tdark** (i.e. no prior information) refers to proteins that have been described at the sequence level, do not satisfy Tclin/Tchem/Tbio criteria, and meet two of the following three conditions: a fractional PubMed publications count [44] below 5, three or more NCBI Gene RIF annotations [45], or 50 or more commercial antibodies, counted from data made available by the Antibodypedia database [46].

5. **Tclin** are targets for which a molecule in advanced stages (Phase I clinical trials and beyond) of development, or an approved drug, exists, and is known to bind to that target with high potency;

6. **Tchem** (i.e. early stage) are proteins for which no approved drug or molecule in clinical trials is known to bind with high potency, but which can be specifically manipulated with small molecules in vitro; typically, the small molecule will have been developed in the context of some interesting target related biology;

7. **Tbio** are targets do not have known drug or small molecule activities that satisfy the activity thresholds detailed below AND satisfy one or more of the following criteria: target is above the cutoff criteria for Tdark, or target is annotated with a Gene Ontology Molecular Function or Biological Process leaf term(s) with an Experimental Evidence code, or target has confirmed OMIM phenotype(s);

8. **Tdark** (i.e. no prior information) refers to proteins that have been described at the sequence level and no further studies have been disclosed. To gain more in -sights on those drug targets, it is necessary to link the proteins to their genomic data, structure data, publicly available small molecule data, as well as the gene expression data in cell lines and tissues.

With the help of these new classifications, IDG aims to shed a light on the poorly understood proteins of the four important gene families (i.e. kinases, GPCRs, hormone receptors, and ion channels) and foster basic research by accumulating genomic data to inform our knowledge of the proteome. In this way, help the pharmaceutical industry with the ability to design novel therapeutics to increase human health.

Despite being publicly available, current data repositories for assays and biological molecules suffer from structural, syntactic, and semantic inconsistencies, complicating data integration, interpretation and analysis. As one of the largest and first repositories of public drug screening data, PubChem, has been essential to illustrate the need for clear metadata standards to describe drug and chemical probe discovery assays and screening results [65]. To address these prevailing issues; we have previously developed the first version of the BioAssay Ontology (BAO) [66]. This first version was developed iteratively based on domain expertise and available

assay data, primarily from the MLP, which we annotated using evolving versions of BAO.

Since the first release of BAO, we have engaged with several more groups in public research projects such as the LINCS, BD2K, and IDG projects, and in pharmaceutical companies and the biomedical ontology community. We aligned the organization of BAO with existing efforts as much as possible, most importantly at the Novartis Institutes of BioMedical Research, and we have significantly extended the terminology and axioms in BAO to cover a broader range of assays and related concepts. Engaging in several other projects also brought the need to develop other ontologies such as LIFE and DTO as well as the need to update the ontologies frequently because of the frequent updates of input data.

In order to handle the challenges that come with creating multiple related ontologies and evolve them frequently we developed a methodology, KNARM. We are currently using the methodology for building LINDO (LIncs meta-Data Ontology).

| Projects | Knowledge Unit Elicitation | | | Knowledge Relationship Elicit. | | | Combined Elicit. | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | *Interview* | *Structured Interview* | Observation | Categorical Sorting | Reporotory Grid Analysis | Formal Concept Analysis | Protocol Analysis | Disclosure Analysis | Sub-language Analysis | Laddering | Grouping Techniques |
| RiboWeb | x | x | x | ✓ | x | x | x | x | x | x | x |
| EcoCyc | x | x | x | ✓ | x | x | x | x | x | x | x |
| OntoBroker | ✓ | | | ✓ | x | x | ✓ | x | | | |
| BioMediator | x | x | x | ✓ | x | x | x | x | x | x | |
| SEMIDA | ✓ | | | | | ✓ | ✓ | ✓ | | | |
| SHOE | ✓ | | | | | ✓ | ✓ | ✓ | | | |
| OntoEdit | ✓ | | | | | ✓ | x | x | | | |
| OBO | x | x | x | ✓ | x | ✓ | ✓ | ✓ | ✓ | x | ✓ |
| SEEK | x | x | x | ✓ | x | ✓ | x | x | x | x | x |
| CommonKADS | ✓ | ✓ | x | ✓ | x | ✓ | x | ✓ | ✓ | ✓ | |
| Bio2RDF | x | x | x | ✓ | ✓ | x | x | x | x | x | x |
| METHONTOLOGY | ✓ | ✓ | x | ✓ | x | ✓ | ✓ | x | x | x | |
| PharmGKB | x | x | x | ✓ | x | x | ✓ | | x | x | ✓ |

Table 2.3: Usage of Knowledge Acquisition Techniques and Projects Reviewed. This table shows a summary of different tools and projects reviewed and what common methods and techniques they use.

# CHAPTER 3

# Approach

In this chapter we describe KNowledge Acquisition and Representation Methodology (KNARM) and how KNARM was applied to build the ontologies in this study: BioAssay Ontology (BAO) 2.0, LINCS Information FramEwork (LIFE) ontology, and Drug Target Ontology (DTO). The ontologies built were designed to share data among themselves in order to avoid duplicating existing work. They also import data from existing ontologies without disturbing their own integrity. Furthermore they are designed to accommodate minor updates and the fluidity of the data coming from the LINCS and IDG projects that are currently in progress.

## 3.1 KNowledge Acqusition Methodology (KNARM)

Effective applications of big data approaches in the life sciences require better, knowledge-based, semantic models that are suitable as a framework for big data integration, while avoiding overly extreme simplification, such as reducing various biological data types to the gene level. A huge hurdle in developing such semantic knowledge models, or ontologies, is the knowledge acquisition bottleneck. Manual methodologies require too much time, are prone to human errors, and fail to help the need when there is large amounts of data production. Automated methods are still very limited and significant human expertise is required.

To systematize this knowledge acquisition and representation challenge, we created a new methodology called KNowledge Acquisition and Representation Methodology (KNARM) is created and used for this research project by combining methods for Database Management Systems, Object Oriented Programming, and Knowledge Acquisition Methods. KNARM is a hybrid method for both acquiring new and existing knowledge and building ontologies with high expressivity. The methodology aims to represent the knowledge acquired from textual data, data available in databases, and ontologies. The Knowledge Representation uses axioms in a systematic, structured, deepening-layering approach for defining concepts in formal logic. We showcase ontologies built using KNARM, explain the details about how it helps better formalize the data with the help of domain experts' insights and use computers' reasoning capabilities to infer new knowledge.

With KNARM, we provide a set of steps that allow the acquisition of the knowledge out of the raw data. We start with the analysis of textual data. This is followed by the acquisition of knowledge out of the existing resources, such as databases and ontologies that is related to the textual data. We then suggest using description languages and formal logic to represent the processed data as well as the existing databases and ontologies. During this process, we suggest two check points for validation of the knowledge acquired and represented in a formal way. It is an agile methodology allowing updates after each iteration, and semi-automated so that each iteration can contain minimum amount of errors and the iterations can be performed fast.

The methodology may be generalized for any big data, but currently it is described for handling biomedical big data. Here is the summary of the steps performed:

1. Sub-language Analysis
2. In-House Unstructured Interview
3. Sub-language Recycling
4. Meta-Data Creation and Knowledge Modeling

5. Structured Interview
6. Knowledge Acquisition (KA) Validation
7. Database Formation
8. Semi-Automated Ontology Building
9. Ontology Validation

### 3.1.1 Sub-language Analysis

Sub-language analysis is a technique for discovering units of information or knowledge, and the relationships between them within existing knowledge sources, including published literature or corpora of narrative text [45]. As the first step of formalization of the data we recommend starting with the existing literature and/or reports for the data. While reading the text data, it is desired to try creating use cases and taking notes aiming to identify patterns and the units of information, concepts and facts in data, that have a recurring pattern. A unit of information is a concept, relationship or data property contained in the data in hand. A use case is a list of actions, event steps that users might follow, questions that can be asked by users, and/or scenarios that users may find themselves in. Example use cases are as follows:

- Search for proteins are in the same kinase branch as target X where there were validated chemical hits from external or internal sources.
- One has an assay X, find the other assays that have the same design but different targets
- Which assay technologies have been used against my kinase? Which cell lines?

After identifying units of information, patterns, and listing some possible use cases the ontology engineers can introduce the domain experts to their preliminary analysis, or continue to work with them towards the next steps of the methodology.

### 3.1.2 In-House Unstructured Interview

After identification of the key concepts and units of information during Sub-language analysis, we perform an interview with the domain experts that are closest to us, who work in the same team. This step can be a performed separately after the sub-language analysis or in a hybrid fashion with the previous step. The unstructured interview is aimed at understanding the data and their purposes better with the help of the domain experts. It can be performed in a more directed fashion by using the previously identified knowledge units or could be treated as a separate process. Together with the previous step, this step also help identify the knowledge units and key concepts of the data.

### 3.1.3 Sub-language Recycling

Following the identification of knowledge units through the textual data of the assays, literature, and unstructured interview with the domain experts, we perform a search on the existing ontologies and databases. The aim of the search on the databases and ontologies is to ascertain the already formalized knowledge units that are identified. We perform and encourage reuse of existing -relevant, and well-maintained- ontologies, aligning them with our ontologies, and using cross-references (annotated as Xref in the ontology) to the various databases that contain the same knowledge units and concepts that we determined to formalize. By recycling the sub-language, not only we save time and effort, but also reuse widely accepted conceptualization of knowledge. In this way, we also aim to help life-scientists by sparing them the painful data alignment practices, and by helping them avoid redundant and/or irrelevant data available in different data resources.

### 3.1.4 Meta-Data Creation and Knowledge Modeling

In this step, we combine the knowledge units and essential concepts identified with those recycled from the existing databases and ontologies to create the meta-data describing the domain of the data to be modeled. The metadata creation can

be a cumbersome task that could be performed in different levels by defining subsets of metadata on various details of the data. For example, with our systematically deepening approach of formalization (i.e. *Systematically-Deepening-Modeling* approach (SDM)), we started with the metadata for proteins and genes, followed by metadata for diseases, tissues and small molecules. The SDM approach allows us to focus on one aspect at a time and extract more detailed (i.e. deeper) metadata, which later allows creating more complex axioms (i.e. modeling of concepts).

In combination with the metadata creation comes a very important step in knowledge acquisition and representation: knowledge modeling. Here, we define knowledge modeling as using axioms to define concepts and aim to help infer new knowledge based on existing data using this axiomatic modeling of concepts. While modeling, we focus on one aspect at a time and create more complex axioms as going deeper into the knowledge. The detailed metadata extracted is utilized on different levels to create axioms that can be modeled without overwhelming the reasoners and other semantic web technologies by creating *nested axioms*. By dividing the knowledge into detail levels and representing different detail levels of the knowledge in different ontologies, we allow reuse of concepts and axioms easily as well (also see modular architecture in Semi-Automated Ontology Building section). This step can be performed within the team first and then can be discussed with the collaborators and other scientists. Alternatively, a bigger initiative can be set up to agree on the metadata, axioms, and knowledge models (examples include OBO Foundry ontologies [35])

### 3.1.5 Structured Interview

Structured Interview consists of close ended questions that are aimed at the domain experts. For our purposes we use metadata created for the knowledge obtained so far in order to perform an interview with the collaborators who are involved in the data creation as well as the scientists who are not involved in the data creation. The aim of the structured interview is to identify any important points that might have missed by the knowledge engineers and the domain experts so far.

In this step, the metadata identified for the data is presented by the knowledge engineers. The data could be dissected based on the metadata identified and the dissected information could also be presented to the collaborators.

### 3.1.6 Knowledge Acquisition Validation

This step could be considered the first feedback. In this step, the sub-language identified and recycled, the metadata, and the data dissected based on the metadata is presented to the domain experts by the knowledge engineer. It could also be presented to a small group of users based on the use cases. The aim in this step is to identify any knowledge that is missed or misinterpreted. If such knowledge exists, we recommend starting from the first step and reiterating the steps listed above.

### 3.1.7 Database Formation

After validating the knowledge acquired is correct and consistent, we start building the backbone for the representation of the knowledge. The first step is to create a database to collect the data in a schema that will facilitate the knowledge engineering. Typically, this will be a relational database. The domain experts may prefer to use different means of handling and editing their data, such as a set of flat files, but we recommend using a database as the main data feed to the ontology that will be created as the final product. The details of the database are designed based on the acquired metadata and data types collected and their relations (see Figure 4.15 for an example database schema). Ideally, the databases should contain the metadata as well as the knowledge units and the key concepts identified in the knowledge acquisition steps. Information that the database may not hold directly includes specific relationships or axioms involving the different knowledge units and key concepts that are identified during the knowledge acquisition. We placed the relationships among the pieces of data in the next step during the ontology building process.

### 3.1.8 Semi-Automated Ontology Building

After placing the data dissected based on the metadata as well as the metadata into the database, we convert the data to a more meaningful format that allows inference of new knowledge that is not explicit in the flat representation in the database. This is achieved using semantic web technologies, mainly an ontology. Building an ontology is particularly relevant for representing complex knowledge involving hierarchies of concepts (i.e. *classes* in ontology) and many specific relationships (i.e. *object properties* in ontology) among concepts and their data properties (i.e. *data properties* in ontology). In this way, flat data obtained can be used to create axioms that represent current knowledge. With the help of DL reasoners, inference of new knowledge and performing complex queries for analysis and exploration becomes possible and easily operable.

We follow the modular architecture that we presented [67–69] while building the ontology. The modular architecture allows easier management and sharing of ontology files, standardized vocabularies and axiomatic representations of knowledge. Modular architecture also allows us to create inter-operable pieces of knowledge that we can easily share, manipulate, and assemble into diverse knowledge environments.

Modularization and ontology development can be performed manually. However, especially while building DTO, we improved our approach by adding automation. We created all vocabulary files and some of the axioms using the database back end and a Java application, OntoJog [68], adding a layer into the modularization and separating the axioms that are automatically created by a software that we implemented and the axioms that are manually added to the ontology.

In order the create the modular architecture first, we determine the abstract horizon between TBox and ABox. TBox, terminological component, contains modules, which define the conceptualization without dependencies. ABox is the assertion component of the ontology, where instances of concepts defined in the TBox are added.

Vocabularies and modules in the core TBox are self-contained and well-defined with respect to the domain and they contain concepts, relations, and individuals. In

this research, *self-contained* means that there is no outside term or relationship in the files; *well-defined* means the terms, relationships, and individuals are generated unambiguously.We can have n of these vocabularies and/or module files in the TBox.

Second, after the n files of modules and/or vocabularies are defined, the modules with axioms that can be generated automatically are created. These new modules created have interdependent axioms. At this level one could create any number of gluing modules, which import other modules without dependencies or with dependencies.For the ontology's core file, these modules need to be self-contained.

Third level contains axioms created manually, however the axioms generated are independent and self-contained. The manual modules are an optional level and they inherit the axioms created automatically.

Forth, at this level we can design modules that import modules from our domain of discourse, and also from third party ontologies. Third party ontologies could be large, therefore a suitable module extraction method (e.g., OWL API) can be used to extract only part of those ontologies (vide supra). We would model this in the ontology-complete level. We can have one ontology-complete file or multiple files, each may be modeled for a different purpose, e.g., tailored for various research groups. Once these ontologies are imported, the alignment takes place. The alignments are defined for concepts and relations using equivalence or subsumption DL constructs. The alignment depends on the domain experts and/or cross-references made in the ontologies.

Fifth, release the TBox based on the modules created from the third phase. Depending on the end-users, the modules are combined without loss of generality. With this methodology we make sure that we only send out physical files that contain our (and the absolute necessary) knowledge.

Sixth, at this level, the necessary modules ABoxes are created. ABoxes can be loaded to a triple store or to a distributed file system (Hadoop DFS [70]) in a way that one could achieve pseudo-parallel reasoning.

At the seventh level, using modules, we define *views on the knowledge base*. These are files that contain imports (both direct and indirect) from various TBoxes

and ABoxes modules for the end-user. It can be seen as a view, using database terminology.

Our modular architecture for the ontologies improved over the span of this research. The modular architecture described for BAO [4] was relying on manual axioms and manual vocabulary additions. Because of the difference in data increase and rapid update requirements as well as the automated steps integrated during implementation, we added a new layer in which we only generate modules that are built using the automated process. We then add the modules that are manually created with the help of a domain expert (see Figure 3.2 for the current modular architecture of DTO).

### 3.1.9 Ontology Validation and Evoluation

The final step in the proposed workflow is the ontology validation. The domain experts as well as the knowledge engineer performs different tests in order to find out if the information in the ontology is accurate. In addition, different reasoners can be run on the ontology to check its consistency. Additional software can be implemented to test the different aspects of the ontology (for example java programs that compare the database with the ontology classes, object properties, data properties, etc.) Finally, queries for the different use cases can be run to check if the ontology implementation answers questions it was meant to answer. If there are any inconsistencies or inaccuracies in the ontology, the knowledge engineer and the domain expert should try to go back to the ontology building step. If the inconsistencies are fundamental, we recommend starting from the first step and retracing the steps that lead to the inconsistent knowledge. Domain experts and ontology engineers can also choose to go back to the *Metadata Creation and Knowledge Modeling* or *Sub-language Recycling* step.

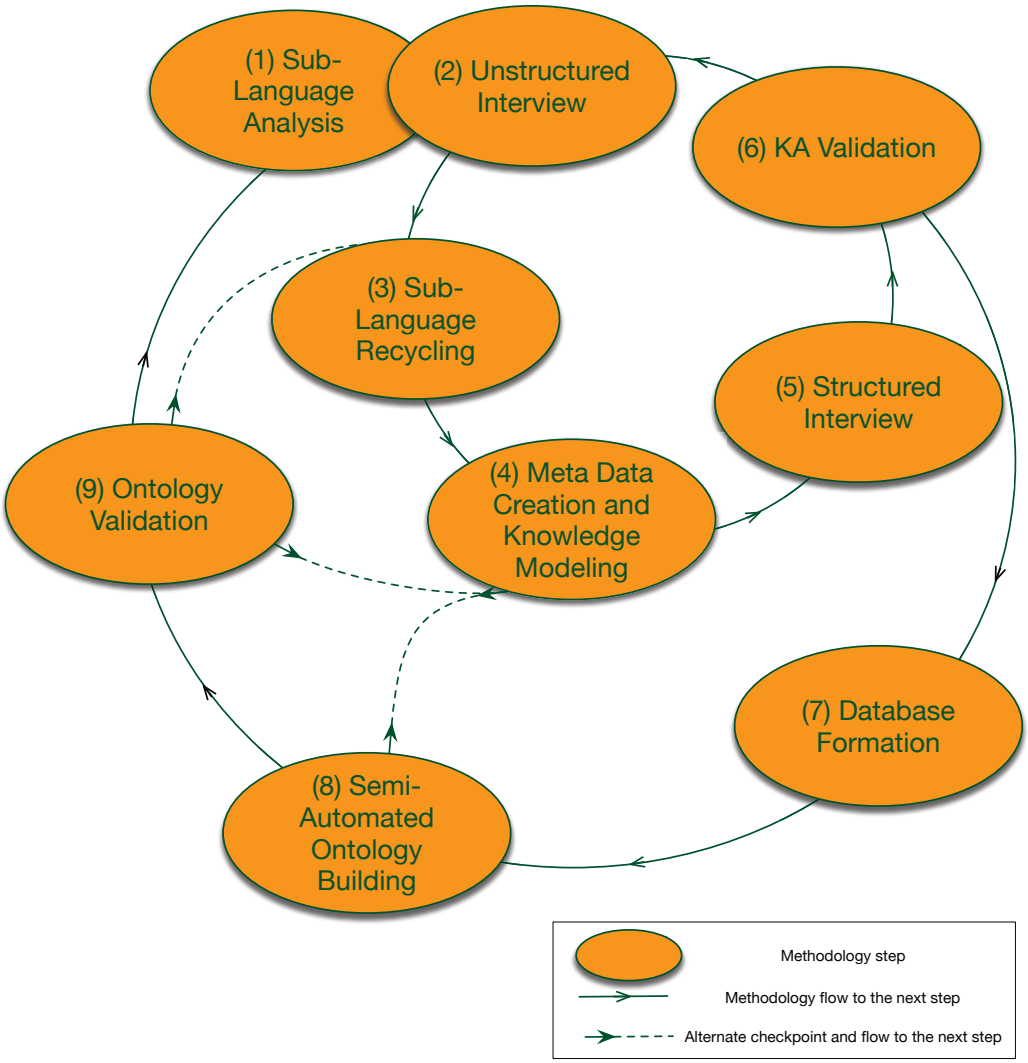



Figure 3.1: The steps of KNowledge Acquisition and Representation Methodology (KNARM). The figure aims to emphasize the continuous development cycle and agile nature of the methodology. The inner cycle can be repeated as many times as required by the domain experts and ontology engineers so that the knowledge can be captured and modeled without flattening the data while making sure that it's accurate. One should note that the inner cycle is more manual, relying more traditional KA methods. The outer cycle is composed of more automated steps, aiding faster building of ontologies.

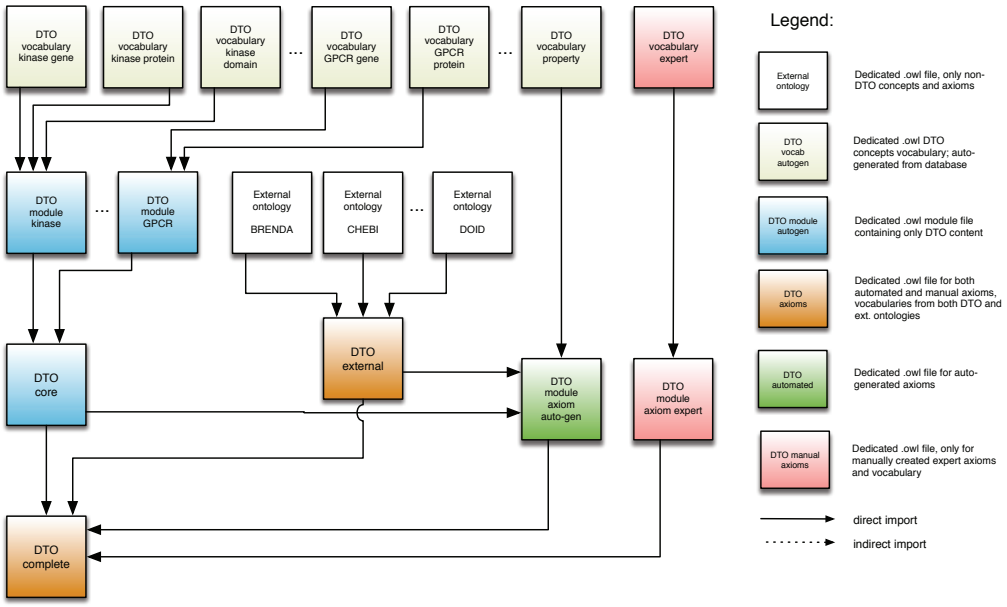



Figure 3.2: Modular Architecture for the Drug Target Ontology. As described our modular architecture for the ontologies improved over the span of this research. The modular architecture described for BAO [4] was relying on manual axioms and manual vocabulary additions. Because of the difference in data increase and rapid update requirements as well as the automated steps integrated during implementation, we added a new layer in which we only generate modules that are built using the automated process. We then add the modules that are manually created with the help of a domain expert.

# CHAPTER 4

# Methods and Applications of KNARM

In this chapter, how KNARM (KNowledge Acquisition and Representation Methodology) was formed, matured dynamically and how KNARM is used for three ontologies - the BioAssay Ontology (BAO), LIncs FramEmework Ontology (LIFE), and Drug Target Ontology (DTO)- is described.

KNARM started to form based on our need to build better and concordant ontologies in a systematic way, more efficiently, and harmoniously. The following subsections give the details of how each step of KNARM was performed while building BAO, LIFE, and DTO along with other design and implementation details for the three ontologies.

## 4.1 LINCS Information FramEwork (LIFE) and The BioAssay Ontology (BAO) 2.0:

### 4.1.1 Sub-language Analysis and Unstructured Interview for BAO and LIFE

As described above the first step, sub-language analysis, is focused on discovering and defining units of information, i.e. concepts and relationships. This is the first step towards identifying meta data information.

BAO [5] was designed and implemented to axiomize knowledge about bioassays. The first version of BAO was designed and implemented with PubChem assays in

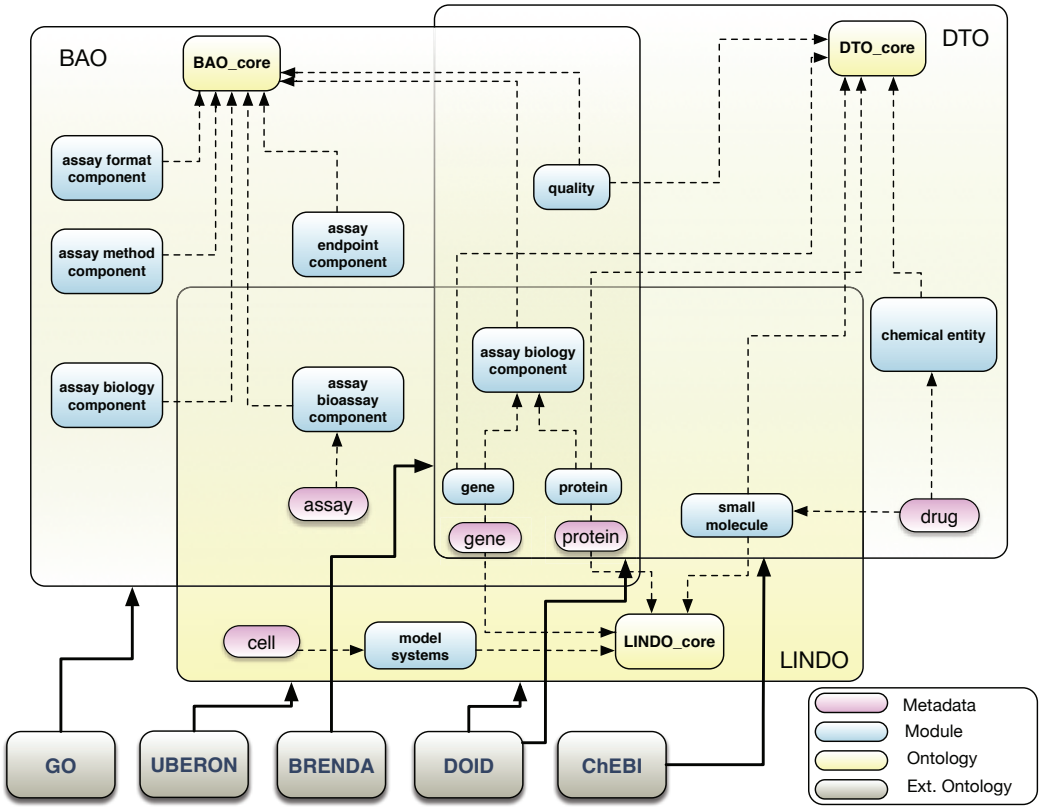



Figure 4.1: Overview of Modeling of BAO, LIFE, and DTO using KNARM. This figure shows how we built concordant ontologies using KNARM as a consistent methodology and our modular architecture which allowed us to reuse and align parts from different ontologies. This conceptual description shows that relationship among some core concepts in the ontologies.

mind, therefore the modeling of the assays and the concepts created in BAO were a reflection of this dataset (see Figure 4.1 for the Overview of Modeling in BAO 1.0)

With the introduction of the The **L**ibrary of **I**ntegrated **N**etwork-Based **C** ellular **S**ignatures (LINCS) project, we worked on integrating the new LINCS assays into the existing BAO. In this step, we reviewed textual descriptions of LINCS assays with domain experts. Starting with the initial concepts identified in BAO's first version, we tried to identify key concepts from the LINCS assays and the bio-entities used in the assays that will allow us to perform key queries. We quickly realized that, BAO's structure and modeling was not designed to handle the changes. Thus, we decided to take a more systematic approach for modeling of the LINCS assays in BAO.

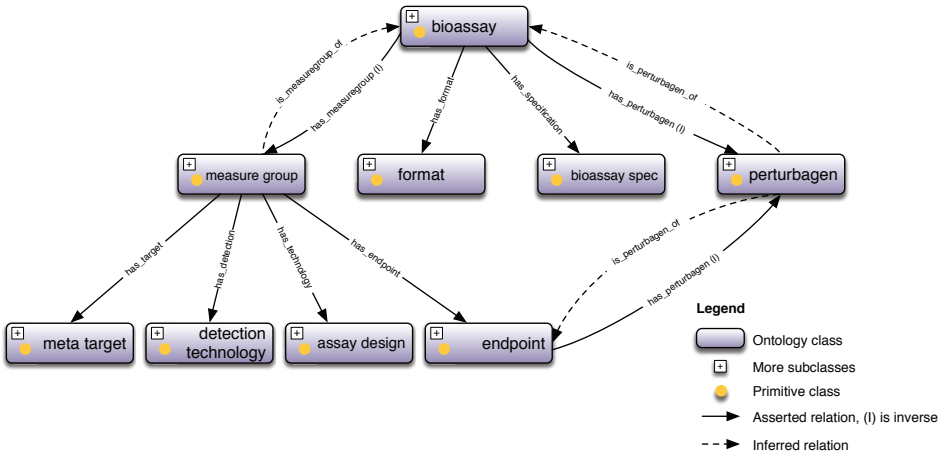



Figure 4.2: Overview of Modeling of BAO assays in BAO1.0 [5]. This modeling of the bioassay concept was based on the bioassays submitted to PubChem. PubChem requires users to enter certain fields of information before they can upload their textual descriptions of assays.

The **L**ibrary of **I**ntegrated **N**etwork-Based **C**ellular **S**ignatures (LINCS) project aims to use computational tools to integrate this diverse information into a comprehensive view of normal and disease states that can be applied for the development of new biomarkers and therapeutics (See Figure 4.3 for the different datasets). [71]

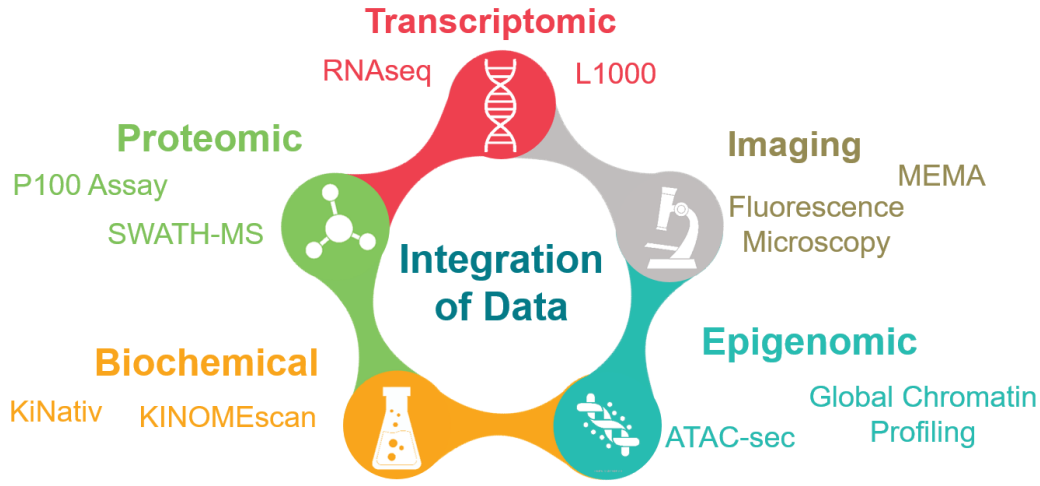



Figure 4.3: Diversity of LINCS assays

The diverse datasets in this project are created by running various assays with different types of molecules such as proteins and genes. Each assay uses these biological molecules in different *roles* in order understand how these molecules react

under varied circumstances and perturbagens. The assays that are reviewed during the Sublanguage Analysis and Unstructured Interview are as follows (also see Figure 4.5):

- **KinomeSCAN** KinomeSCAN assay measures a group proteins after a group of perturbagens were introduced to the biochemical model system. [72]
- **KiNativ** KiNativ assay observes proteins in the presence of small molecule perturbagens in Lysates. [73]
- **L1000 Assay** The L1000 platform pairs ligation-mediated amplification (LMA) with a Luminex-bead based detection system to allow the quantitation of 1000 mRNA transcripts per well. The L1000 assay is an extension of a previously reported method for expression profiling based on Luminex bead technology [74] to create a 1000-plex profiling solution. By using this platform, certain genes are over-expressed and under- expressed, and based on the expression levels signatures are created. [75]
- **2-3 Color Apoptosis** The 2-Color and the 3-Color Apoptosis assays use different markers to illuminate different cell lines and observe which cells are going through apoptosis in the presence of small molecule perturbagens. [76]
- **Cell Cycle State Assays** These assays are called *Proliferation/Mitosis Assay* and *Mitosis/Apoptosis Assay* by the data creator, Harvard Medical School. The two assays use different markers to identify the cell-cycle states and apoptosis. [76]
- **Cell Growth Assay** In this assay cell growth and apoptosis is observed after using perturbagens on different cells. Cell nuclei is stained and cell division is measured and reported. [76]
- **Cue Signal Response (CSR) Assay** The CSR Assay measures the cytokine secretion and phosphorilation levels in different cell lines after introducing small molecule and biological molecule perturbagens to the model system. [76]

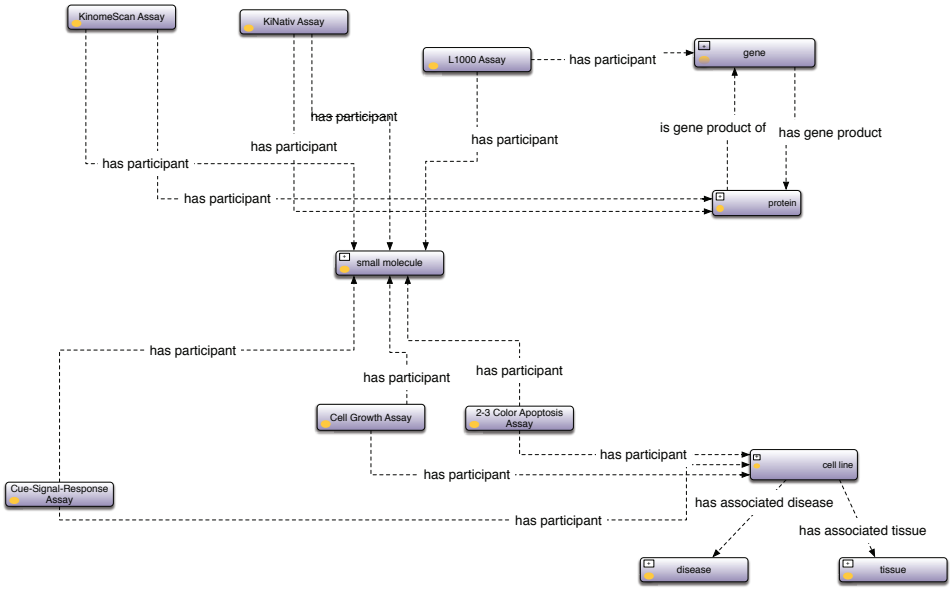



Figure 4.4: LINCS assays currently described in BAO. This figure shows how the assays connect via perturbagens and other *participants* in the modeling.

As briefly described above, these assays aim to measure different biological processes and/or molecular functions. Furthermore, the assays use different biomolecules, such as proteins, genes, as well as their mutated variations, small molecules, cell lines, and various other participants.

An unambiguous description the assays is essential for the LINCS project in order to help both the researchers who are participating in the LINCS project and those who are using the LINCS findings to aid projects outside. In order to provide an unambiguous, formal description, we generated a new set of meta-data for the second version of BAO.

### 4.1.2 Sub-language Recycling for BAO and LIFE

As described previously, this step involves searching and discovering units of information that already exists in other ontologies and databases. We aim to adopt as many concepts as possible from existing ontologies, given the logical (or in the cases of taxonomies textual) descriptions of the concepts align with our needs. This also helps avoid duplication of efforts and reuse of previously established vocabulary for the terms. We also aim to accomplish community suppor and cross-reference

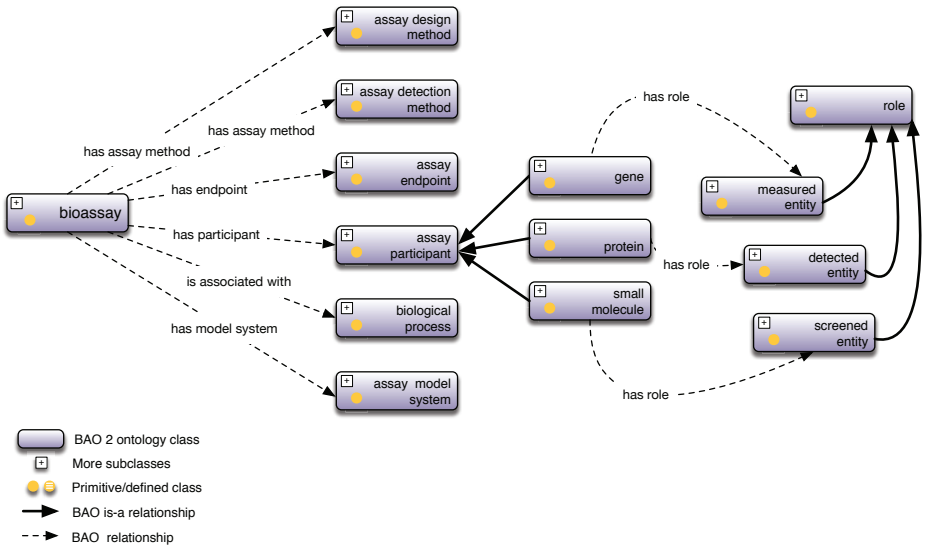



Figure 4.5: Basic Conceptual Modeling in BAO 2.0 and beyond. This figure shows how the current modeling of assays differ from the modeling in BAO1.0. Systematically Deepening Modeling (SDM) adds a layer to the basic concepts such as genes and proteins by giving them different *roles* in different assays. While this modeling is very comprehensive and accurate in terms of philosophical view of concepts, sometimes this deepening modeling causes problems with computation of inferences.

and/or map existing efforts. Using Bioportal [14] or contacting with ontology groups (such as OBO foundry) we search for assay related terms. So far we have used the following ontologies:

1. Biological process and molecular function terms were extracted from the Gene Ontology (GO) [25]
2. A number of relationships are extracted from Relationship Ontology (RO) [35]
3. All the organism names are extracted from NCBI Taxonomy Ontology (NCBITaxon) [77]
4. Most of the Cell Lines are extracted from Cell Line Ontology [20]
5. Diseases from the Disease Ontology(DOID) [19]
6. Units from Unit Ontology (UO) [78]

7. Chemical entities and roles from ChEBI [79]

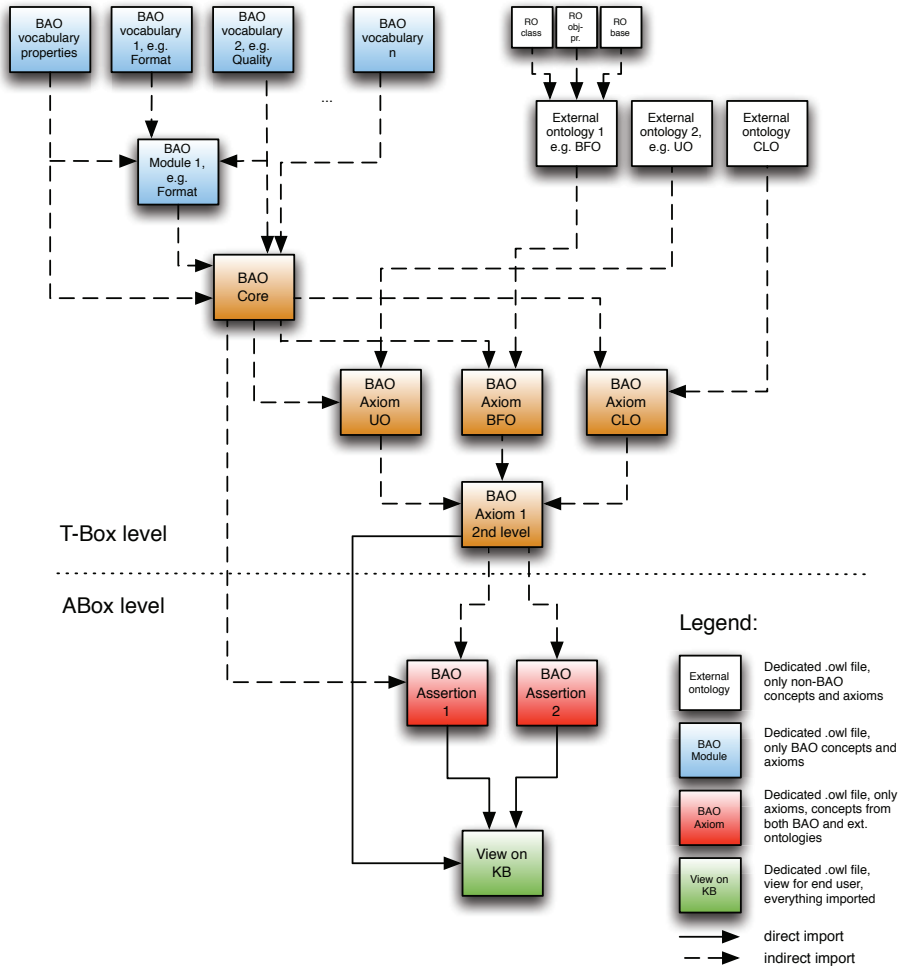



Figure 4.6: BAO Modularization. This modularization assumes that all axioms are added manually.

We also used well-established databases such as UniProt and ENTREZ for information related with bio-molecules such as proteins and genes. For example all the protein names are extracted from UniProt and cross references to UniProt and ENTREZ IDs are contained in the ontologies.

### 4.1.3 Meta-Data Creation snd Knowledge Modeling for BAO and LIFE

Based on the Sub-language Analysis, the In House Unstructured Interview and Sub-language Recycling, the next step in formalizing the assays is creating a set of meta-data.

The meta-data creation step is a combination of analyzing the standards already existing,i.e. widely used data integration standards such as Minimum Information for Biological and Biomedical Investigations (MIBBI) Standards, and understanding the patterns of the data in hand.

For the LINCS assays, we were able to identify patterns among the assays and created a sheet with meta-data (see Figures 13 and 14 for details of metadata and modeling and an example modeling of an assay based on metadata defined). This meta-data then served as the modeling pattern for the assays. In this way, a uniform modeling was identified. We axiomize the reoccurring components for all of the assays as follows:

*Assay Participants:* In the LINCS project, even though all assays are related with and complementary of each other, each assay deals with a diverse group of molecular entities. The molecular entities that take place in the different assays are defined as a 'participant' of the assay.

*Model System:* Inspired by the *model organisms*, we started using the term *Model System* to identify the assay mediums that are used while performing the assays. This term is a generalized version of previously used BAO term 'bioassay format'.

*Perturbagen:* One of the most important part of the assays is the perturbing agent. While many assays used small molecules as perturbagens, we also had assays that use hormones, or other biological molecules as perturbagens in order to detect different cellular responses.

*Assay Design Method:* The *Assay Design Method* describes how a biological or physical process screened / investigated in the model system is translated into a detectable signal.

*Biological Process or Molecular Function:* According to the Gene Ontology (GO) [25] a biological process is any process specifically pertinent to the functioning of integrated living units: cells, tissues, organs, and organisms. A process is a collection of molecular events with a defined beginning and end. In addition, a molecular function is defined as elemental activities, such as catalysis or binding, describing the actions of a gene product at the molecular level. A given gene product may exhibit one or more molecular functions.

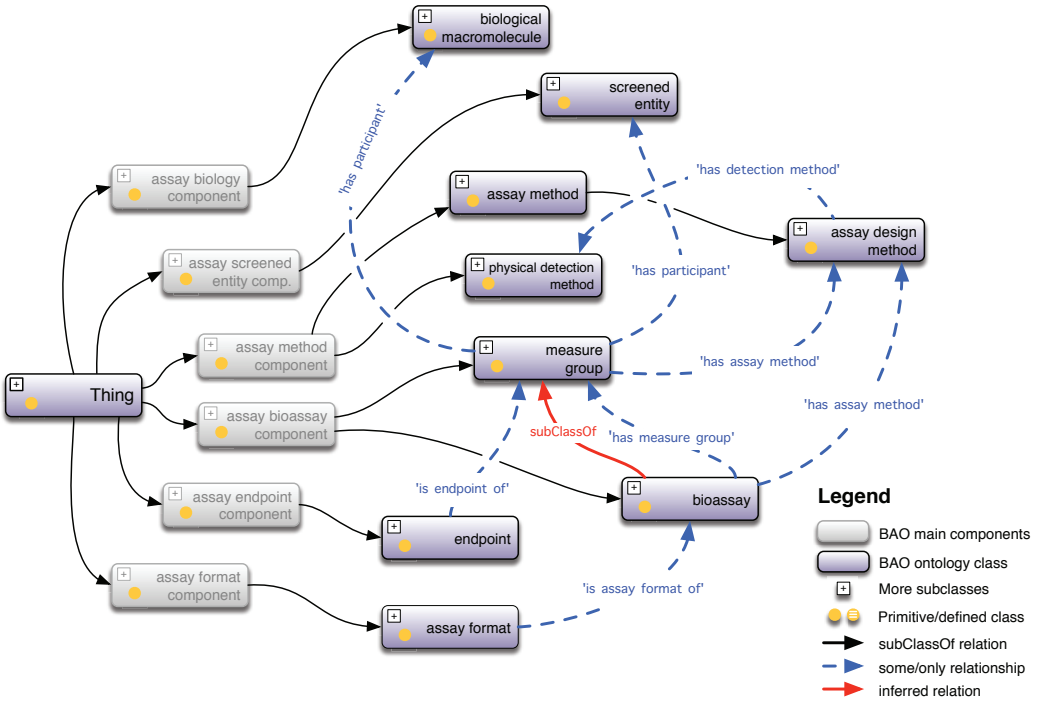



Figure 4.7: Main Classes and Basic Object Properties for Modeling *bioassay* concept in BAO 2.0. As seen in the figure even for a single concept, the relationships and modeling becomes a highly connected graph.

At its core, the assay aims to measure responses for a biological process or a molecular function. Since the LINCS assays are complementing each other in various ways, we decided that it is crucial to capture which biological processes and/or molecular functions the assays are involved with. In this way, we could

easily query and reason assays that are involved with the same biological processes and/or molecular functions. Since GO is widely used by many different databases such as UniProt [80], Reactome [81], and ChEBI [79], we chose to extract terms from the GO, and use the GO terms in our logical axioms.

*Measured Entity:* In the assays that we modeled, the same molecular entities are used in different *roles*. For example, one protein could be the end product in one assay, while it is a byproduct in another. In order to model in a clear and concise fashion, we decided to logically axiomize the roles of the participants in the assays, as opposed to having multiple upper classes to the same entity. The concept Measured Entity is a product of such need. It is modeled as a role. The entity that has the measured entity role is the output of a biological reaction or process that is quantized either directly (by the presence of a tag or probe) or indirectly in a coupled reaction.

*Assay Detection Method:* Assay Detection Method refers to the physical method or technology that generates a readout for the effect caused by a perturbagen in the assay. The assay detection method could be an instrument or a combination of instruments, tags, and/or dyes.

*Detected Entity:* This concept is being recorded because of the need to differentiate between what is measured in the assay and what captured by the detection method. Detected Entity is the immediate entity that is detected by using the detection method. In some cases, detected entity can be the measured entity, however, in other cases detected entity acts as a bridge for what we are aiming to measure with our assay.

*Endpoint:* An endpoint, alternatively called *result*, is a quantitive or qualitative representation of a perturbation (change from a defined reference state of the model system) that is measured by the bioassay.

The creation of meta-data further allowed us to find a solution for difficult modeling problems such as the modeling of assay endpoint vs. measure group.We had

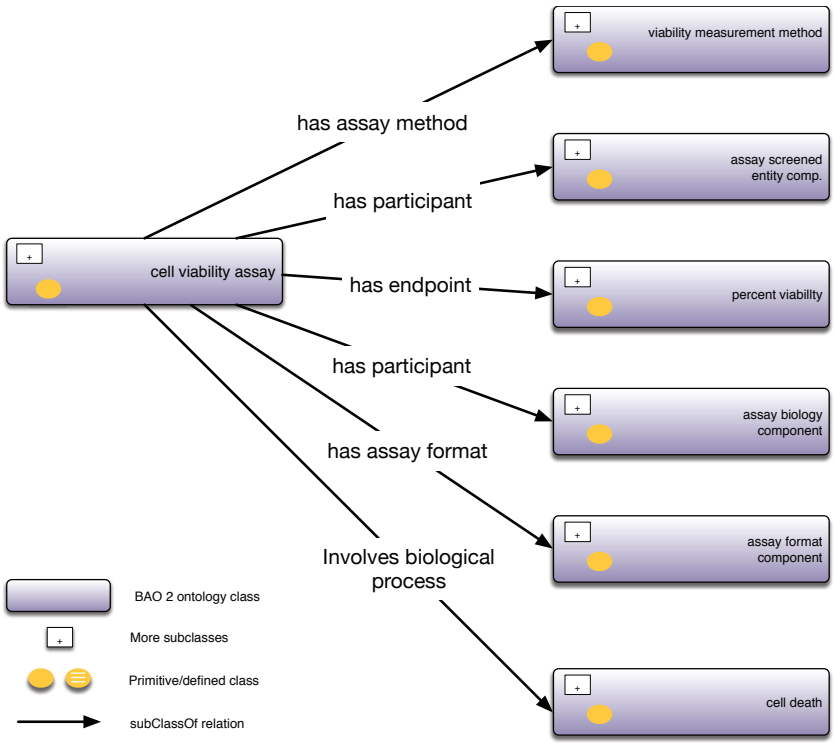



Figure 4.8: An example modeling and metadata of Cell Viability Assay in BAO.

previously introduced the concept *measure group* to link multiple endpoints to the same bioassay [82] (See Figure 4.9). We have now generalized this model so that *measure group* can be derived from one or more measure groups. This allows the formal and iterative construction of more complex assays and endpoints that are derived from multiple measurements.

### 4.1.4 Structured Interview for BAO and LIFE

Based on the meta-data created, I have interviewed the researchers at the LINCS data creation centers and outside of the group, mainly the data creation group at Harvard Medical School.

This step is to confirm that the interpretation of the text data is correct and accurate. Additionally, this step can be used in combination with other methods in order to decide on a concept's proper name. With this step, the aim is to finalize names and types of concepts used in the meta-data. Furthermore, it is to make sure

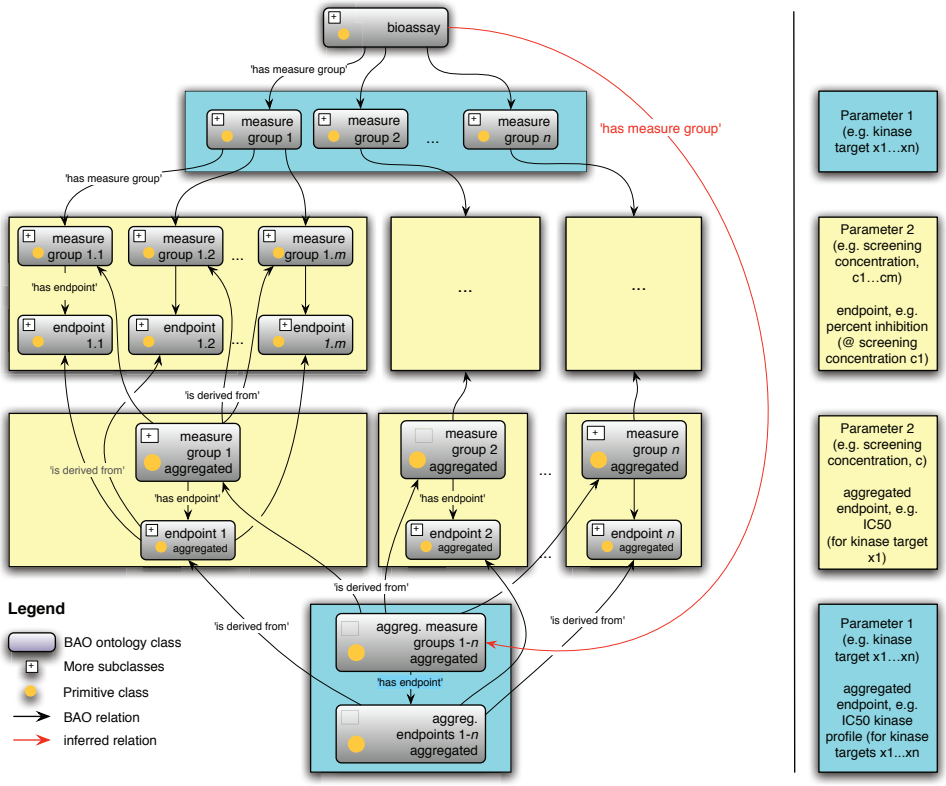



Figure 4.9: Modeling of Measure Goup Concept in BAO.

that the ontology engineer is on the same page as the domain experts before starting to write the axioms into the ontology. Therefore, this step can be combined with the next step, i.e. Knowledge Acquisition Validation.

### 4.1.5 Knowledge Acquisition Validation (KA Validation) for BAO and LIFE

In this case after the metadata creation and after the various interviews and reviews of the data the forms I have designed were filled. Before the axiomization of the assays, the forms were shared with the research scientists inside and outside of the SchÃ$\frac{1}{4}$hrer Lab in order to make sure that the information contained was valid. Corrections if necessary were made on the excel sheet provided and sent back to me. Please see appendices for documents created working with the Harvard Medical School for the Structured Interview and KA Validation steps.

### 4.1.6 Database Formation for BAO and LIFE

This step of KNARM provides a basis for the semi-automated ontology building. Initially for the second version of BAO, a database was not created. This was because previously BAO concepts and axioms have been created manually and there wasn't a big demand in adding various new concepts at once. A new database is being built for generating BAO vocabulary in an automated fashion by the SchÃ$\frac{1}{4}$hrer Lab as a result of the full application of this methodology (i.e. KNARM), and as the need for regular updates for the BAO project increased.

For the LINCS data a database was created by the software group at the CSS for keeping the LINCS data and providing the back-end for the LIFEwrx web-based software (see Figure 4.10). However, the database was used to extract the data as Excel files for semi-automated ontology building.

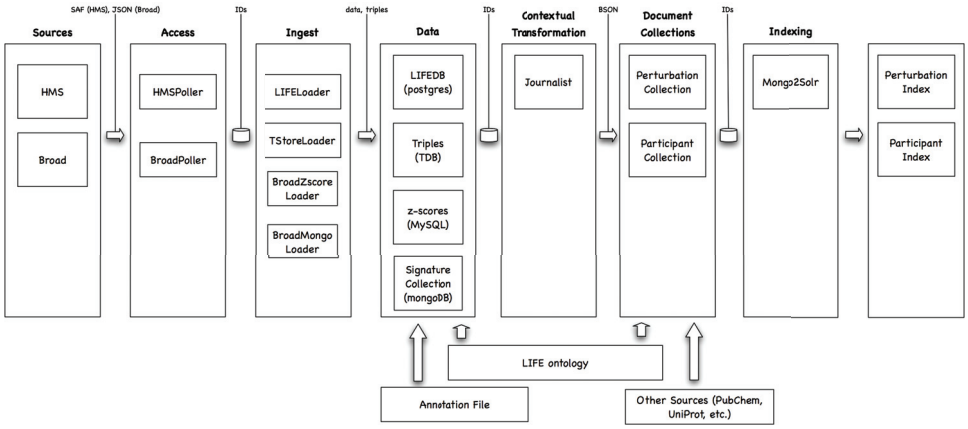



Figure 4.10: Use of Database and the LIFE ontology for the LIFEwrx software (Courtesy of Schührer Lab)

### 4.1.7 Semi-Automated Ontology Building for BAO and LIFE

As mentioned above, most of the data for the LIFE and BAO ontologies were collected and formed via excel sheets. Therefore, I used Java and OWL API to process the excel files and convert them to RDF in order to build parts of the ontology in an automated fashion.The modular architecture formed for BAO was adopted for LINCS, but no formal changes were made to the architecture.

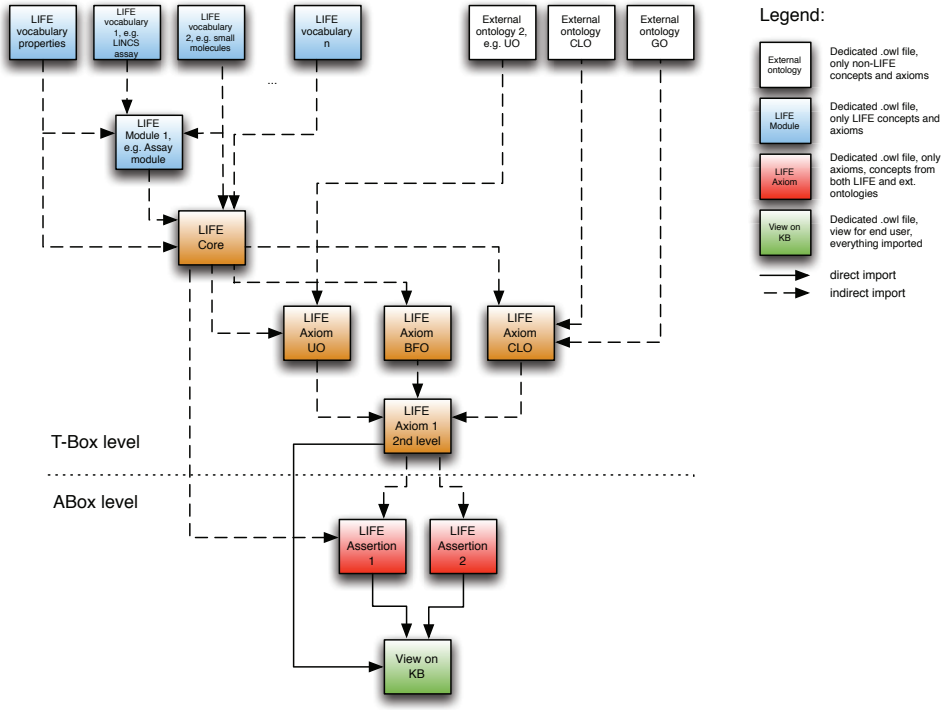



Figure 4.11: LIFE Modularization following the same principles followed for BAO modularization

### 4.1.8 Ontology Validation for BAO and LIFE

LIFE ontology was not published to the community, so the only validation it had was through the various reasoners. BAO, on the other hand, is a widely used ontology with lots of applications, such as BioAssay Express (BAE) of Collaborative Drug Discovery (CDD), pharmaceutical companies (Astra-Zeneca, Roche, etc.), and government facilities, such as Environment Protection Agency (EPA), and government funded projects, such as BARD, LINCS, BD2K, using it as their primary ontology for annotating their bioassays. The need to update BAO for the needs of different users urged us to identify a systematic and consistent routine for updating. An initial updating routine was implemented via a joint effort between CDD and SchÃ¼hrer Lab and a new NIH grant is awarded for further implementation of tools for efficient semi-automated ontology validation and updating process.

The key steps for ontology validation and update using the BAE tool are as follows:

1. Create BAE Absence Report (generated at: `http://www.bioassayexpress.com/BioAssayExpress/diagnostics/absence.jsp`)

2. Absence reported is exported to Excel, and then reviewed by a content (domain) expert to QC, filter for unique new terms, which are then exported to a new-term template to further clarify by adding required fields–definitions, parent BAO class, relevant references/ hyperlinks. This List of Requested Terms is then shared with University of Miami (BAO team).

3. A survey of content experts could be used to decide on the final labels for terms. (Currently this step is not performed) The links to surveys are in a document and they live in SurveysAndResults folder in Google Drive

4. The List of Requested Terms from Step 2 above is divided and transformed into appropriate separate .csv files by a University of Miami BAO domain expert along with the ontology engineer, inputting the new BAO ID to be assigned and the appropriate BAO parent class ID, using a predefined template.

5. Attention must be paid to terms that already exist in external ontologies (but need to be added to BAO) and terms for which BAO needs to coordinate with external ontologies (e.g.DOID, CLO) to request external IDs.

6. Output files are created in .owl format by the ontology engineer at UM

7. Output files are merged with the appropriate vocabulary files by the ontology engineer in UM and merged into BAO complete for initial check-up and QC run by the Java Programs and Pipeline Pilot Scripts. Manual check is also performed using Protege.

8. After final corrections (changes may be needed, iterating back to step 4 and performing steps 4,5,6 again or manually editing the .owl files), final bao_complete.owl is created (as per KNARM).

9. After the finalized BAO_complete is created, all files are updated on GitHub and the BioAssay homepage for BioPortal to collect the new version by the ontology engineer at UM.

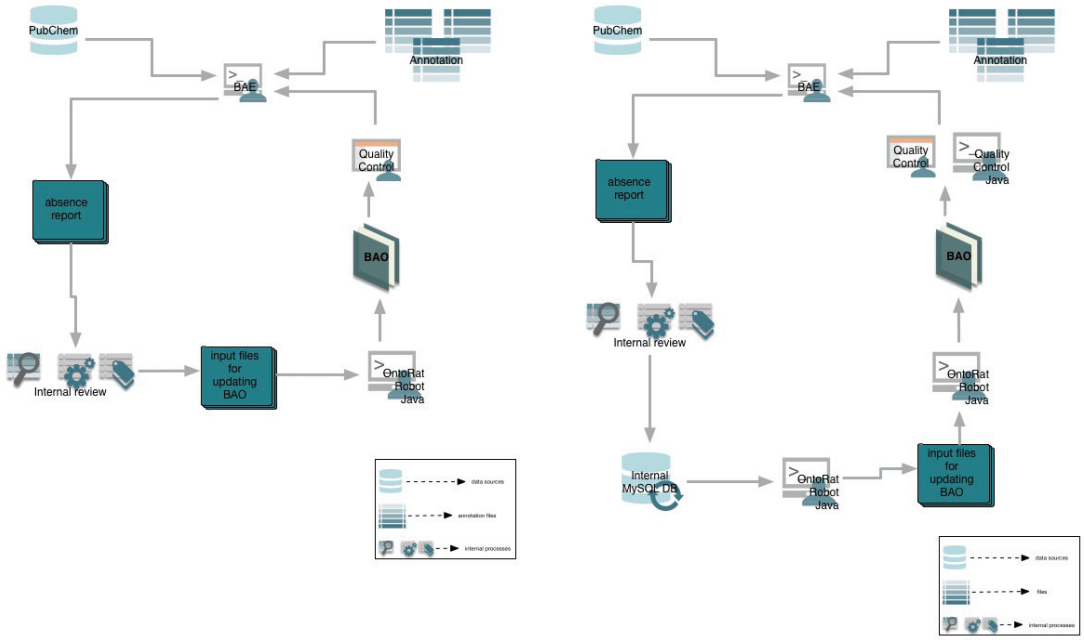



Figure 4.12: Left: Current workflow for evolving BAO, Right: Ideal Workflow for Evolution of BAO

## 4.2 Drug Target Ontology (DTO):

As a part of the IDG project, we designed and implemented the **D**rug **T**arget **O**ntology (DTO), advancing the ontology architecture that we used for the BAO and LIFE ontologies. The goal of the IDG project is to improve our understanding of the proteins that belong to the four most commonly drug-targeted protein families properties and yet are not annotated with as many details as the commonly used drug-targets. In its pilot phase the program aims to create a data resource center, the Knowledge Management Center (KMC) that will catalog known information about the four protein families and obtain additional information about their function(s). Ultimately, the KMC aims to have methods that allow the life-scientists to identify and prioritize the poorly annotated proteins for further study [64]. The major difference between the previously defined LINCS and the IDG projects' data is that IDG focuses on biomolecules and their tissue and disease relationships while LINCS main focus on the different assays that they create and run. The IDG data is mainly composed of drug-targeted protein families: G-protein coupled receptors (GPCR), nuclear receptors, ion channels, and protein kinases. Therefore, we fo-

cused more on modeling the biological and chemical molecules, changes that they have been through such as modifications, mutations, etc., rather than the assays they were used in.

KNARM finalized before the implementation of DTO and all the steps have been followed while building the Drug Target Ontology.

### 4.2.1 Sub-language Analysis and In-House Unstructured Interview for DTO

Different communities have been using the term 'drug target' ambiguously with no formal generally accepted definition. The DTO is aimed at developing a formal semantic model for drug targets including various related information such as protein, gene, protein tissue localization, disease associations, and many other types of information. The initial interviews and sub-language analysis steps involved determining the different classifications of the drug targets and the properties of them. Recently the IDG project defined *drug target* as a native (gene product) protein or protein complex that physically interacts with a therapeutic drug (with some binding affinity) and where this physical interaction is (at least partially) the cause of a (detectable) clinical effect. DTO defined a DTO specific term *drug target role* to be used in axioms related with the proteins listed in DTO. The text definition of *drug target role* is a role that is beared in a material entity, such as native (gene product) protein, protein complex, microorganism, DNA , etc., that physically interacts with a therapeutic or prophylactic drug (with some binding affinity) and where this physical interaction is (at least partially) the cause of a (detectable) clinical effect.

At the current phase, DTO focuses on protein targets.

The IDG drug targets are categorized as four super families with respect to the depth of investigation from a clinical, biological and chemical standpoint:

1. Tclin are targets for which a molecule in advanced stages of development, or an approved drug, exists, and is known to bind to that target with high potency

2. Tchem are proteins for which no approved drug or molecule in clinical trials is known to bind with high potency, but which can be specifically manipulated with small molecules in vitro

3. Tbio are targets that do not have known drug or small molecule activities that satisfy the Tchem activity thresholds, but were the targets annotated with a Gene Ontology Molecular Function or Biological Process with an Experimental Evidence code, or targets with confirmed OMIM phenotype(s)

4. Tdark refers to proteins that have been described at the sequence level and no further studies have been disclosed

DTO proteins have been classified into various categories based on their structural (sequence/domains) or functional similarity. A high-level summary of the classifications for Kinases, Ion Channels, GPCRs, and Nuclear Receptors.

Most of the 578 kinases covered in the current version of DTO are protein kinases. These 514 PKs are categorized in 10 groups that are further subcategorized in 131 families and 82 subfamilies.

The 62 non-protein kinases are categorized in 5 groups depending upon the substrate that are phosphorylated by these proteins. These 5 groups are further sub-categorized in 25 families and 7 subfamilies. There are two kinases that have not been categorized yet in any of the above types or groups.

The 334 Ion channel proteins (out of 342 covered in the current version of DTO) are categorized in 46 families, 111 subfamilies, and 107 sub-subfamilies. Similarly, the 827 GPCRs covered in the current version of DTO are categorized in 6 classes, 61 families and 14 subfamilies. The additional information whether any receptor has a known endogenous ligand or is currently `orphan` is mapped with the individual proteins. Finally, the 48 nuclear hormone receptors are categorized in 19 NR families.

Following my reviews of the free-form text about the data in hand, the domain experts in the group provided help with answering my questions. At times, the reviews of the free-form text was performed together with the domain experts. This process is defined as the unstructured interview, because there are no predefined set of questions asked to the domain expert. The questions are asked in a conversation-

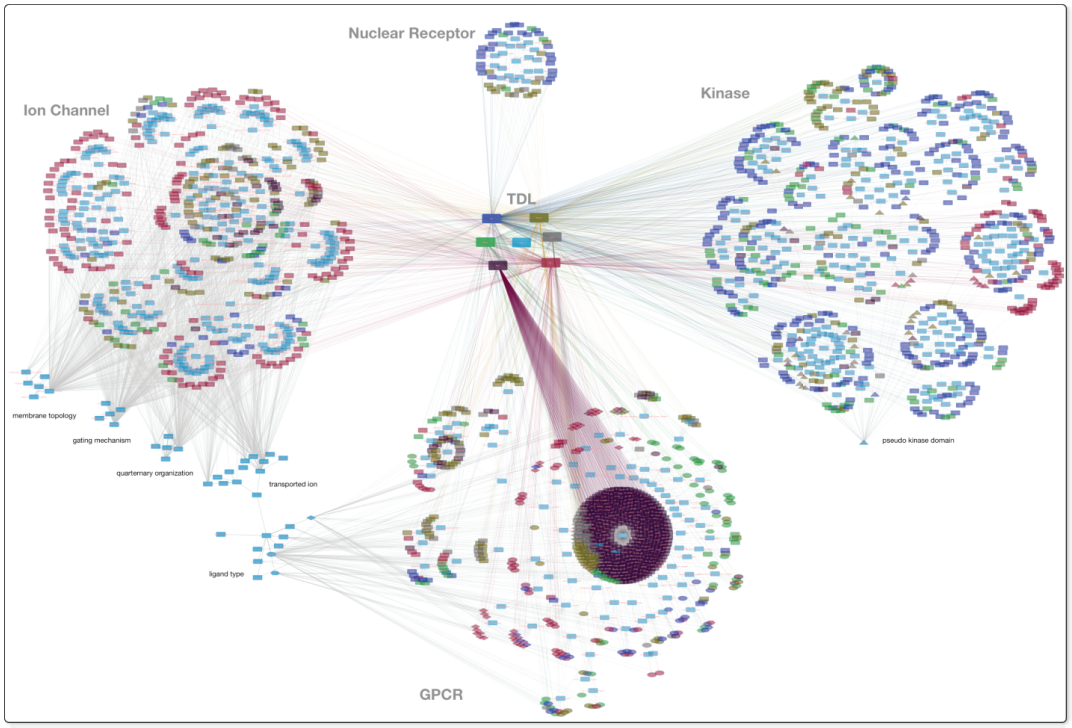



Figure 4.13: Protein Classes in DTO

like environment in order to understand the assays better and identify a pattern among the various kinds of bio-molecules and their uses as well as their structure and functions in drug discovery related assays and projects.

Above classifications of the data are performed by the domain experts and provided to me in excel sheets. We have further discussed other classification issues such as how one could classify mutated and modified proteins.I proposed some modeling solutions. It was decided that the best way to classify them was as a subclass of their wild-type proteins.

The following information about the protein classes are also identified as properties to model and axiomize.

- Kinase relationships

  protein-gene relationships

  protein-disease relationships

  protein-tissue relationships

  target development level relationships

has quality pseudokinase relationships

- GPCR relationships

  protein-gene relationships

  protein-disease relationships

  protein-tissue relationships

  target development level relationships

  has-ligand-type relationships

- IC relationships

  protein-gene relationships

  protein-disease relationships

  protein-tissue relationships

  target development level relationships

  has channel activity

  has gating mechanism

  has quaternary organization

  has topology

- NR relationships

  protein-gene relationships

  protein-disease relationships

  protein-tissue relationships

  target development level relationships

These different properties identified in the first step are later used to create meta-data, model the knowledge, and axiomize in the ontology building process.

### 4.2.2 Sub-language Recycling for DTO

While designing the ontology, we decided to add the UniProt IDs for the proteins and the ENTREZ IDs [83] for the genes. In addition to this, we wanted to include the textual definitions for the genes and the proteins. We also cross-referenced the synonymous names and symbols for the molecules that already exist in different databases. Because we want the Drug Target Ontology to be as comprehensive as possible with existing information about the biological and chemical molecules that the DTO contains. In this way, we aim to help the life-scientists query and retrieve knowledge derived for the different drug targets that they are working on. To do that, we wrote various scripts using Java to retrieve information from databases. These databases include UniProt and NCBI databases for ENTREZ IDs for the genes.

In addition to the publicly available databases and data, we also used the collaborators' databases (TCRD and Jensen Lab's databases) in order retrieve information about proteins, genes and their related target development levels( TDLs), as well as the tissue and disease information.

The information on the Jensen Lab's database is retrieved through text mining and has a scoring system [84].The lab also has information about the protein-disease, and protein-tissue relationships and scores based on lab experiments. We retrieved the proteins, with their tissue and disease relationships with the confidence scores that are given to the relationships. We put this data into our database and use this information while creating the ontology's axioms that refer to the probabilistic values of the relationships.

In addition to the larger scale information downloads from the databases mentioned above, a vast amount of manual curation for the proteins and genes is performed in the team by the domain experts. Most significantly improved drug target classification for kinases, ion channels, nuclear receptors, and GPCRs. For most protein kinases we followed the phylogenetic tree classification originally proposed by Sugen and the Salk Institute (available from http://www.kinase.com/). Protein kinases not covered by this resource were manually curated and classified mainly based

on information in UniProt and also the literature. Non-protein were curated and classified based on their substrate chemotypes. We also added pseudokinases, which are characterized by a catalytically inactive kinase domain and which are increasingly recognized and relevant drug targets. For 44 kinases we are still in the process of completing manual annotations and classification. Nuclear receptors were organized following the IUPHAR classification. GPCRs were classified based on information from several sources primarily using GPCRDB (http://www.gpcr.org/7tm/) and IUPHAR as we have previously implemented in our GPCR ontology. However, not all GPCRs were covered and we are aligning GPCR ontology with other resources to complete classification for 33 receptors. We are also incorporating ligand chemotype-based classification. A basic classification of ion channels is available in IUPHAR. However, a better classification is required including domain functions, subunit topology, and heteromer and homomer formation. We curated much of this information and are currently completing the classification based on this information. This manual classification is in progress for 342 ion channels.

Protein domains were annotated using the Pfam Web Service. The domain sequences and domain annotations were extracted using custom scripts. Several of the kinase domains were manually curated based on their descriptions. For nuclear receptors we identified and annotated the ligand binding domains, which are most relevant as drug targets. For GPCRs we identified 7tm domains for majority (780 out of 827) of GPCRs. Further work is needed to identify domains of interest for the remaining GPCRs. Ion channel domains were annotated and trans-membrane domains were identified; additional ion channel domains, such as regulatory and ligand binding are also relevant for ion channel drug targets. Further curation is required to classify and annotate them. In addition to the curated drug target family function-specific domain annotations, we generated comprehensive Pfam domain annotations for all TCRD drug targets and extracted domain sequences. The domain sequences were compared to the PDB chain sequences by BLAST and e-values were calculated. For significant hits we computed domain identities using the EMBOSS software suite. These results are currently processed and filtered to restrict the results to those domains that were identified as most relevant for each

target family. The domains are classified manually based on curated annotations to generate meaningful interpretable assertions in DTO.

### 4.2.3 Meta-Data Creation and Knowledge Modeling for DTO

Based on the Sub-language Analysis, the In House Unstructured Interview and Sub-language Recycling, the next step in formalizing the assays is creating a set of meta-data.

The meta-data creation step is a combination of analyzing the standards already existing, e.g. PFam annotations, and understanding the patterns of the data in hand. For the first version of the DTO, we decided the add the following axioms for the different protein classes:

- Kinase relationships
  - protein-gene relationships
  - protein-disease relationships
  - protein-tissue relationships
  - target development level relationships
  - has quality pseudokinase relationships
- GPCR relationships
  - protein-gene relationships
  - protein-disease relationships
  - protein-tissue relationships
  - target development level relationships
  - has-ligand-type relationships
- IC relationships
  - protein-gene relationships
  - protein-disease relationships

protein-tissue relationships

target development level relationships

has channel activity

has gating mechanism

has quaternary organization

has topology

- NR relationships

  protein-gene relationships

  protein-disease relationships

  protein-tissue relationships

  target development level relationships

Target development levels (TDL) were assigned using *has target development method* relationship and based on the following criteria:

1. Tclin are proteins targeted by approved drugs as they exert their mode of action. The Tclin proteins are designated drug targets under the context of IDG.

2. Tchem are proteins that can specifically be manipulated with small molecules better than bioactivity cutoff values (30 nM for kinases, 100 nM for GPCRs and NRs, 10 uM for ICs, and 1 uM for other target classes), which lack approved small molecule or biologic drugs. In some cases, targets have been manually migrated to Tchem through human curation, based on small molecule activities from sources other than ChEMBL or Drug Central.

3. Tbio are proteins that do not satisfy the Tclin or Tchem criteria, which are annotated with a Gene Ontology Molecular Function or Biological Process with an Experimental Evidence code, or targets with confirmed OMIM phenotype(s), or do not satisfy the Tdark criteria detailed in 4).

4. Tdark refers to proteins that have been described at the sequence level and have very few associated studies. They do not have any known drug or small molecule activities that satisfy the activity thresholds detailed in 2), lack OMIM and GO terms that would match Tbio criteria, and meet at least two of the following conditions:

    - A PubMed text-mining score is less than five
    - less than or equal to three Gene RIFs
    - less than or equal to 50 antibodies available per Antibodypedia [85]

Each protein has a target development level (TDL), i.e., Tclin, Tchem, Tbio and Tdark. The protein is linked to gene by *has gene template* relation (see the details of modeling in Figure 4.14).

The gene is associated with disease based on evidence from the DISEASES database. The protein is also associated with some organ, tissue, or cell line using some evidence from TISSUES database. Important disease targets by inference based on the protein - disease association, which were modeled as strong-, at least some-, or at least weak- evidence using subsumption. DTO uses the following hierarchical relations to declare the relation between a protein and the associated disease extracted from the DISEASES database. In the DISEASES database, the associated disease and protein are measured by a Z-Score. In DTO the relationships are translated as follows:

- *has associated disease with at least weak evidence from DISEASES* (translated for Z-Scores between zero and 2.4)
- *has associated disease with at least some evidence from DISEASES* (translated for Z-Scores between 2.5 and 3.5)
- *has associated disease with strong evidence from DISEASES* (translated for Z-Scores between 3.6 and 5)

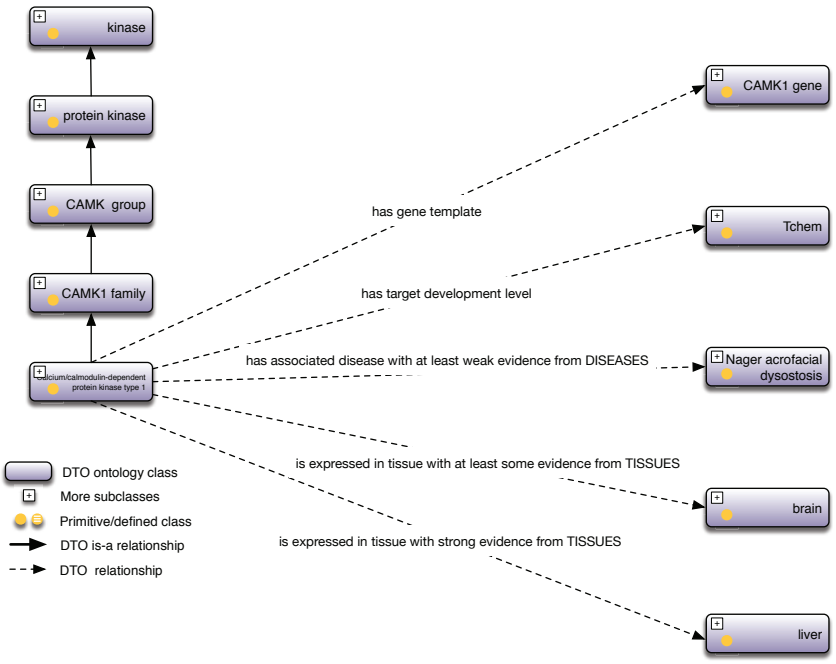



Figure 4.14: This figure shows how metadata is used for an example modeling of a protein. The relationships described above are added to their respective classes. In the figure it can be observed that the protein's hierarchy and its relationships are modeled based on the previous steps of KNARM

### 4.2.4 Structured Interview for DTO

Based on the meta-data created, I have interviewed the researchers in the group and outside of the group. This step is to confirm that the interpretation of the text data is correct and accurate. Additionally, this step can be used in combination with other methods in order to decide on a concept's proper name. In this case, we chose to use existing names in well-known databases such as UniProt.

With this step, the aim is to finalize names and types of concepts used in the meta-data. Furthermore, it is to make sure that the ontology engineer is on the same page as the domain experts before starting to write the axioms. Therefore, this step can be combined with the next step, i.e. Knowledge Acquisition Validation.

### 4.2.5 Knowledge Acquisition Validation (KA Validation) for DTO

In this case after the metadata creation and after the various interviews and reviews of the data the forms I have designed were filled. Before the axiomization of the assays, the forms were shared with the research scientists inside and outside of the SchÃ¼hrer group, especially with the scientists in the IDG project in order to make sure that the information contained was valid. Corrections if necessary were made on the excel sheets provided and sent back to me.

### 4.2.6 Database Formation for DTO

The previous experience in the LIFE ontology, the dealings with Excel files provide a very poor way of keeping track of the related data and updates. Furthermore, the frequency of the need for updates for DTO was higher than the need for updates for BAO and LIFE ontologies. It quickly became apparent that an efficient and less error-prone way to update the ontology was crucial. For the DTO, a new small MySQL database was formed to handle the data. Drug Target Ontology (DTO) uses various external databases and ontologies to retrieve information. The information from these databases is retrieved via web-based applications and in-house-built scripts. The data that is used to build DTO is then housed in our internal MySQL database.

The database schema below (Figure 4.15) is provided for the DTO ontologies staging database created by me. This database then used to automate some of the ontology creation and was used to extract data for the ontology's axioms.

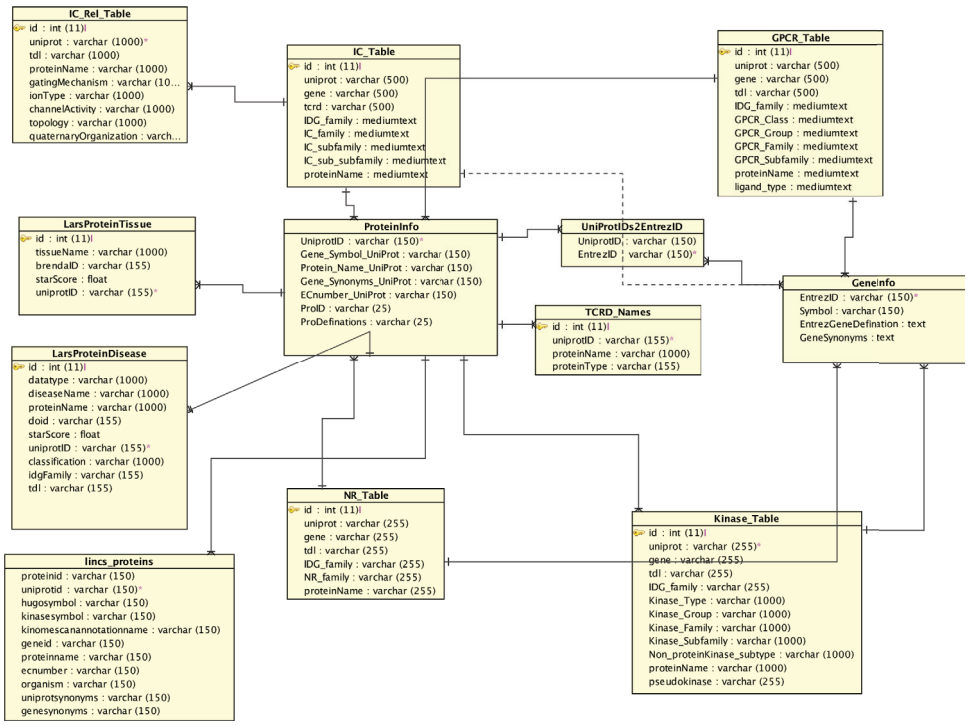



Figure 4.15: This is the Database Schema for the initial database created for building DTO 0.1. It was designed based on the different protein classes in DTO and the relationships and metadata that we wanted to capture for the ontology. This database was used to build the ontology in a semi-automated way. The data saved in this simple (un-optimized) MySQL database was queried and used for building the ontology using Java and OWL API.

### 4.2.7 Semi-Automated Ontology Building for DTO

#### 4.2.7.1 Knowledge Modeling of the Drug Target Ontology:

In BAO, the formal descriptions of assays are axiomized. LIFE formally describes the participants and their relationships to the LINCS assays. DTO, which is created for the IDG project, focuses on the bio-molecules and their natural properties, such the specific ions for ion-channeling proteins, as well as their relationships to the specific diseases and tissues.

The goal of the IDG project is to improve our understanding of the proteins that belong to the four most commonly drug-targeted four protein families (G-protein coupled receptors (GPCR), nuclear receptors, ion channels, and protein kinases), and yet, are not annotated with as many details as the commonly used drug-targets. In its pilot phase the program aims to create a data resource center, the Knowledge Management Center (KMC) that will catalog known information about the four

protein families and obtain additional information about their function(s). Ultimately, the KMC aims to have methods that allow the life-scientists to identify and prioritize the poorly annotated proteins for further study.

The major difference between the previously defined LINCS and the IDG projects' data is that IDG focuses on biomolecules and their tissue and disease relationships while LINCS main focus on the different assays that they create and run. Therefore, we focused more on modeling the biological and chemical molecules, changes that they have been through such as modifications, mutations, etc., rather than the assays they were used in.

As described above, we build modular ontologies for different life-sciences projects such as BAO and LINCS. However, the IDG project presented a new challenge which was massive amounts of data on protein and genes that we wanted to express as classes and axioms in OWL. Not only the amount of data, but also the frequency of data updates have been overwhelming. Therefore we had to automate the ontology building process as much as possible and the come up with a new way of modularization. We use Java, OWL API and Jena to build the ontology in a semi-automated way by using our local database in a new modularization architecture given in detail below.

#### 4.2.7.2 A New Modular Architecture for the Drug Target Ontology:

The modular architecture of the DTO is advanced over the modular architecture of BAO [4]. Because of the difference in data and the automated steps during implementation, we added a new layer in which we only generate modules that are built using the automated process. We then add the manually created modules with the help of a domain expert.

First, we determine the abstract horizon between TBox and ABox. TBox contains modules, which define the conceptualization without dependencies. These modules are self contained and well defined with respect to the domain and they contain concepts, relations, and individuals. We can have $n$ of these modules.

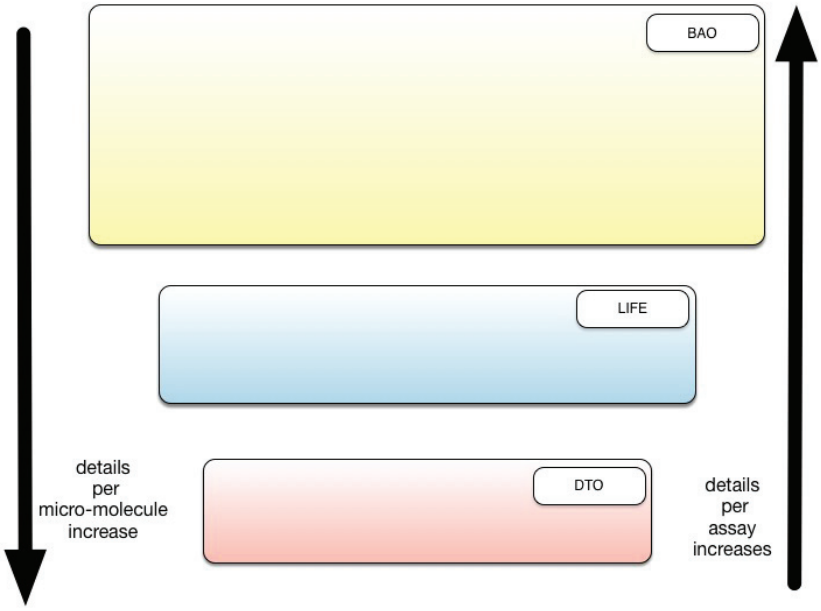



Figure 4.16: Building of our concordant ontologies was made possible by using KNARM and the same modular architecture consistently, and using the systematically deepening- modeling (SDM) approach with the three ontologies. Using this approach, we modeled bioassay related data in the BioAssay. For example we modeled and axiomized various bioassay related concepts such as assay format, assay design method, assay detection method and instruments in BAO. We added axioms that specify the assay participants for LINCS assays in LIFE ontology, such as kinases for KiNativ and KinomeSCAN assays. We modeled and axiomized various details about drug targets in the DTO ontology, such as their related diseases, tissues, and mutation information. With the help of our modularization approach and modular architecture we were able to align the drug targets in DTO with the various participants used in LINCS assays and LINCS assays with the general assay related concepts by using BAO. With this systematically deepening modeling approach, we aim to model and query knowledge without over-simplifying the knowledge and overwhelming the reasoners that help infer new knowledge.

Second, once the $n$ modules are defined, the modules with axioms that can be generated automatically are created. Those modules have interdependent axioms. At this level one could create any number of gluing modules, which import other modules without dependencies or with dependencies. It also is self-contained. This means that there is no outside term or relationship in the files.

Third level contains axioms created manually, however the axioms generated are independent and self-contained. The manual modules are an optional level and they inherit the axioms created automatically. A good example of axioms that may be seen in this level are axioms for protein modifications and mutations, which have been challenging modeling questions. At this level, the self-contained DTO_core is also generated with the existing modules.

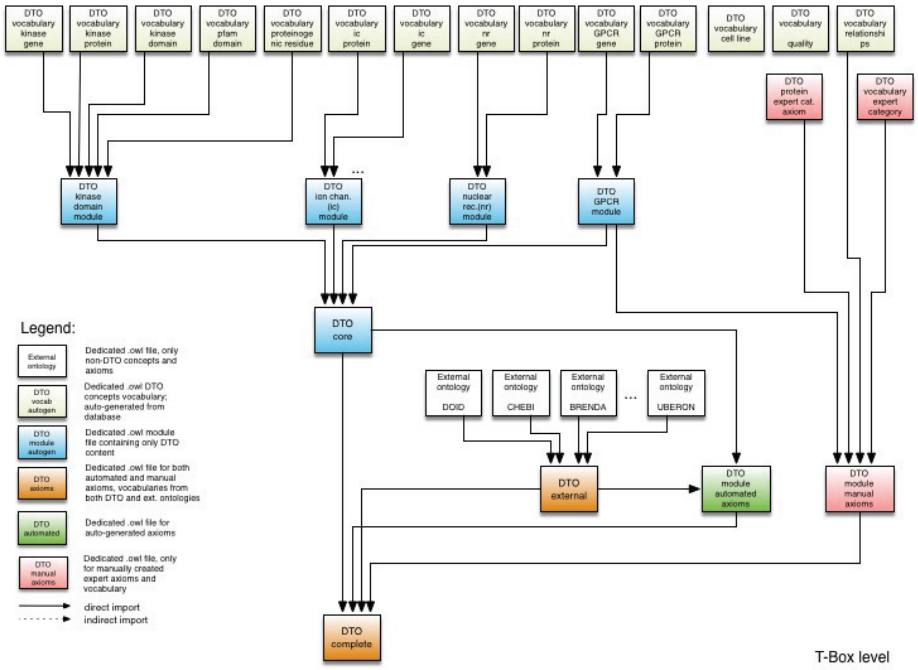



Figure 4.17: DTO Modularization. The modular architecture of the DTO is advanced over the modular architecture of BAO [4]. Because of the difference in data and the automated steps during implementation, we added a new layer in which we only generate modules that are built using the automated process. We then add the manually created modules with the help of a domain expert.

Forth, at this level we can design modules that import modules from our domain of discourse, and also from third party ontologies. Third party ontologies could be large, therefore a suitable module extraction method ( e.g., OWL API) can be used to extract only part of those ontologies ( *vide supra*). We would model this in the DTO_complete level. We can have one DTO_complete file or multiple files, each may be modeled for a different purpose, e.g. tailored for various research groups.

Once these ontologies are imported, the alignment takes place. The alignments are defined for concepts and relations using equivalence or subsumption DL constructs. The alignment depends on the domain experts and/or cross- references made in the ontologies. For DTO, the most significant alignment made is between UBERON and BRENDA ontologies for the tissue information.

Fifth, release the TBox based on the modules created from the third phase. Depending on the end-users, the modules are combined without loss of generality.

With this methodology we make sure that we only send out physical files that contain our (and the absolute necessary) knowledge.

Sixth, at this level, the necessary modules ABoxes (again $1 \ldots n$ ABoxes) are created. ABoxes can be loaded to a triple store or to a distributed file system (Hadoop DFS [70]) in a way that one could achieve pseudo-parallel reasoning.

At the seventh level, using modules, we define *views on the knowledge base*. These are files that contain imports (both direct and indirect) from various TBoxes and ABoxes modules for the end-user. It can be seen as a *view*, using database terminology.

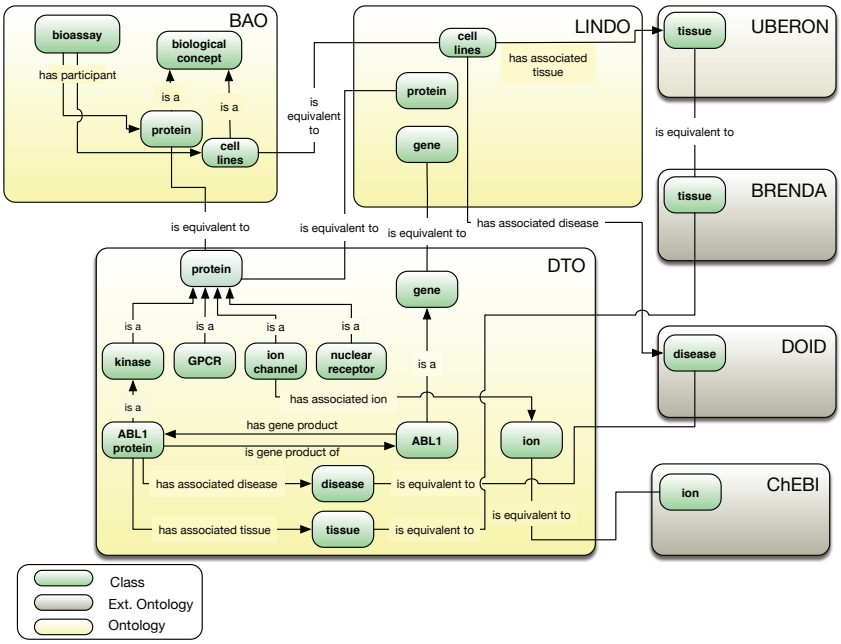



Figure 4.18: This figure shows a conceptual example modeled by using the concepts from BAO, LIFE and DTO as well as their connections to the external ontologies such as the Disease Ontology (DOID) and UBERON tissue ontology. As mentioned above with our Sub-language Recycling step, we try to reuse as many concepts as we could from existing ontologies. In this way we aim to utilize existing efforts, align our vocabulary with already established resources, and avoid duplication of efforts to reduce ambiguity for users.

### 4.2.8 Ontology Validation for DTO

After I created DTO 0.1, the ontology was shared with the larger IDG community for feedback. We also had an in-house feedback loop headed by Stephan Schürer

and Yu Lin. Based on the updates, the concept drug target is defined and Pfam domains are added to the ontology. Version 1.0 of DTO is publicly available on its web page [68] and on Bioportal. A paper introducing the first public version is also submitted [68] (with Hande Kücük-McGinty and Saurabh Mehta as first co-authors).

# CHAPTER 5

# Results

In this chapter, some questions that could be answered by using the three ontologies and their modular implementation are showcased. This section can also be viewed as part of the *Ontology Validation* step of the KNARM.

Figures and results for select SPARQL examples are presented and explained in detail to show how/which inferences lead to the results. Since BAO, LIFE, and DTO use a modular approach for modeling drug-discovery related data, we are able to create different *view*s that would help concentrate on their parts of interest, i.e. concepts and relationships directly related with use cases. Using ontologies' modular architecture, we extracted the LINCS assays from BAO by using Jena, and OWL API, used the cell line module from LIFE. While extracting the LINCS assays from BAO, the concepts used in the axioms for these assays were also extracted based on the RDF graph. The Drug Target Ontology (DTO) was combined with the parts from BAO and LIFE,in order to query the following use cases. The use cases described below were performed using various tools together. The system architecture used for performing these queries is described in Figure 5.1.

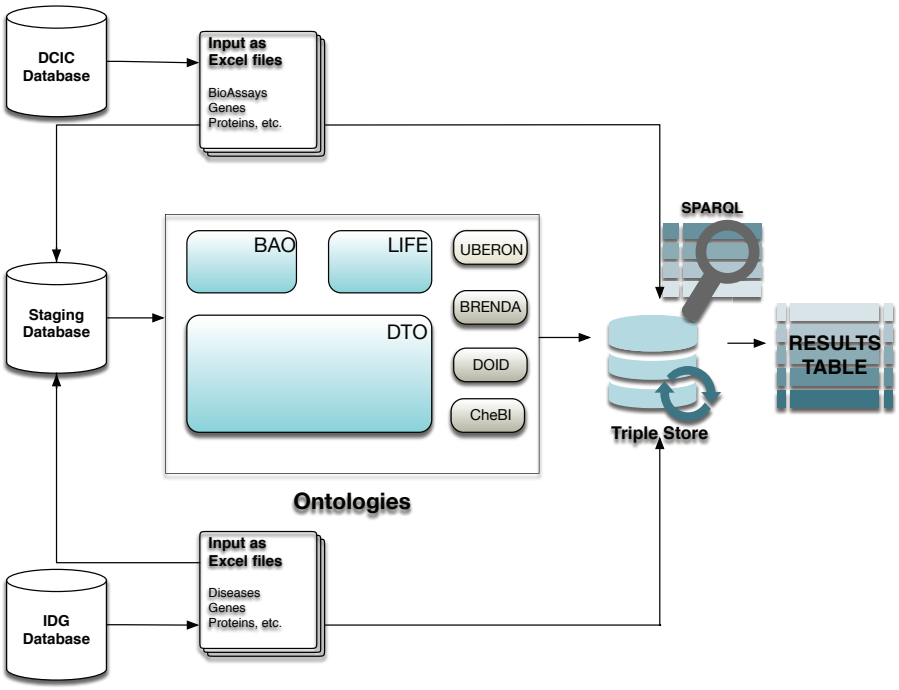



Figure 5.1: Various Tools are used to perform the queries described below. This diagram shows how the different tools and data were combined in order to retrieve results. The data used for the queries are extracted from the LINCS Data Portal, designed and implemented by Schürer Lab. The data extracted is aligned with the staging databases designed for BAO and DTO. Using the ontologies, the database alignments, the reasoners available, and the triple store on UM CS servers the query results are retrieved as tables.

## 5.1 Use Case Examples

### 5.1.1 Example Query 1

*Find LINCS assays that measure protein binding.*

```
PREFIX bao:  <http://www.bioassayontology.org/bao#>
PREFIX obo:  <http://purl.obolibrary.org/obo/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX dto: <http://www.drugtargetontology.org/dto/>
SELECT DISTINCT ?subject_label WHERE {
 #LINCS assays involving binding
 ?subject rdfs:subClassOf ?s1 .
  ?obj rdfs:subClassOf <http://purl.obolibrary.org/obo/GO_0005488> .
           ?s1 owl:onProperty bao:BAO_0003107; owl:someValuesFrom ?obj .
           ?subject rdfs:label ?subject_label .

}
LIMIT 100
```

Listing 5.1: SPARQL query

This is a simple query that works with one external ontology (Gene Ontology (GO) ) and classes from BioAssay Ontology. The inference and query result is based on T-Box reasoning based on the axioms.

First, we determine an abstract horizon between the A-box and the T-Box


Figure 5.2: Abstract horizon between A-Box and T-Box is denoted by dotted line. T-Box contains axioms while A-box contains individuals

Second, we determine the classes that we need for this particular question.

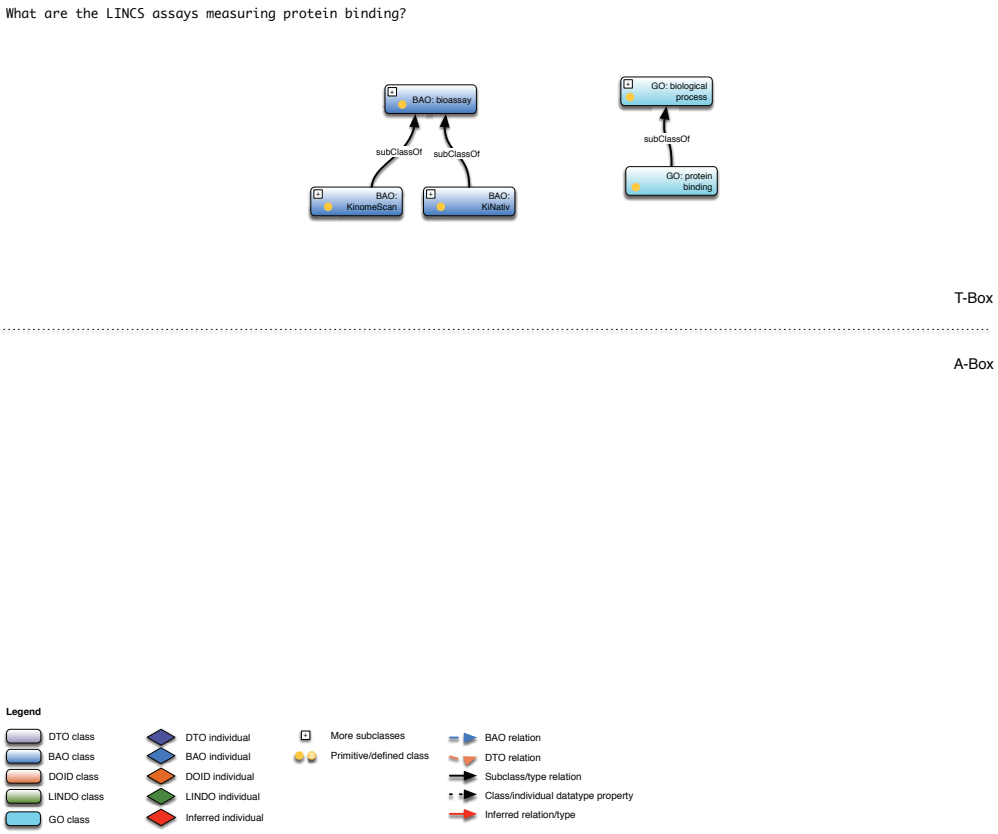



Figure 5.3: Figure shows the classes from BAO and GO that are related with this query.

The reasoner uses the axioms and individuals asserted for these classes (as seen in the figure) to determine the result set. The axioms in this case are already asserted in the ontology. Individuals are added for demonstration purposes.

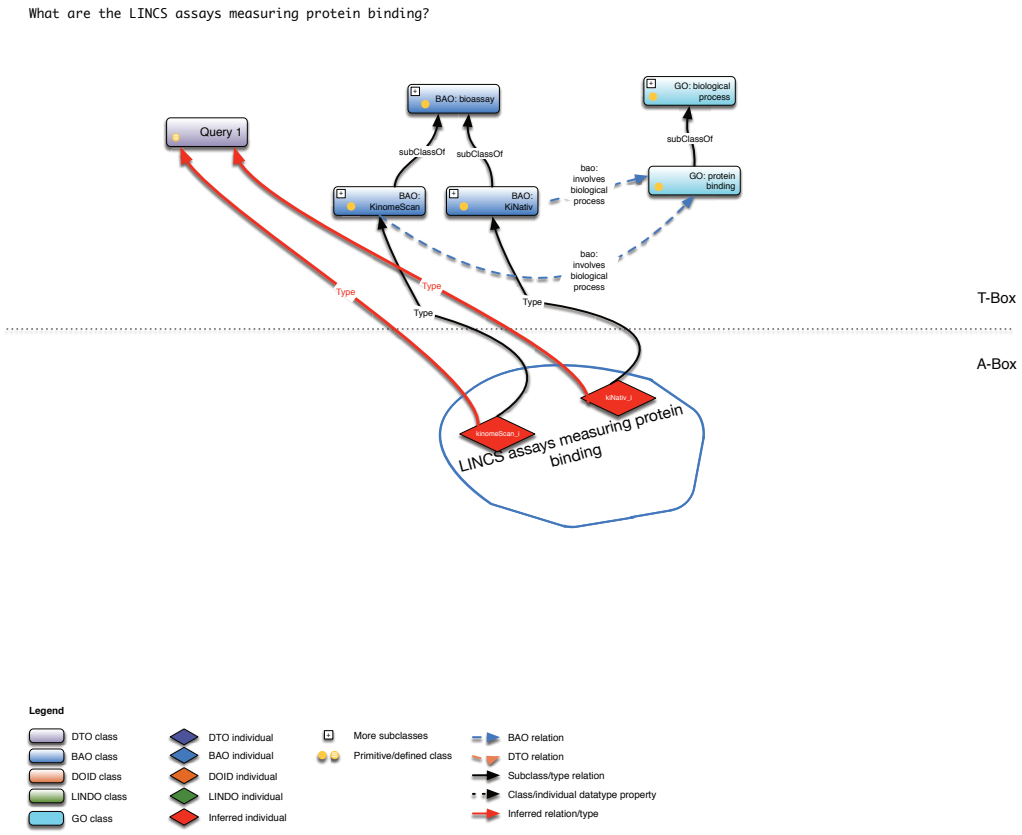



Figure 5.4: Classes from BAO and GO with their relationships (object properties) and their individuals.

### 5.1.2 Example Query 2

*Find proteins participated in LINCS binding assays and has target development Tdark.*

This is s slightly more complicated query generated based on the query defined in Query 1 above. This query depends on the axioms used above, i.e. which assays from the LINCS project (found in BAO and LIFE module extracted) and the participants of these assays (extracted from the LINCS Data Portal) are added as individuals to the A-Box.

The individuals added as participants of these assays are linked to DTO vocabulary based on their UniPort IDs. In this way, these individuals related with LIFE also become connected with DTO.

The resulting participants are found based on A-Box reasoning, while their *target developmental level (TDL)* (in this case *Tdark*) is found in the T-Box axioms. The T-Box axioms related with TDL are found in DTO. The axioms are asserted between protein classes and the TDL vocabulary classes.

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX dto: <http://www.drugtargetontology.org/dto/>
#proteins participated in LINCS binding assays and has target development Tdark
SELECT DISTINCT ?subject_label WHERE {

  #LINCS assays involving binding
  ?subject rdfs:subClassOf ?s3 .
  ?obj rdfs:subClassOf <http://purl.obolibrary.org/obo/GO_0005488> .
           ?s3 owl:onProperty bao:BAO_0003107; owl:someValuesFrom ?obj .
  ?subject rdfs:subClassOf ?s1 .
  #has target development Tdark
  ?s1 owl:onProperty <http://www.drugtargetontology.org/dto/DTO_91000020>; owl:someValuesFrom dto:DTO_00400004
     .


}
LIMIT 10000
```

Listing 5.2: SPARQL query

### 5.1.3 Example Query 3

*Find the kinases used in the LINCS assays that are measuring protein binding and have evidence that associates them with cancer.*

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX dto: <http://www.drugtargetontology.org/dto/>
PREFIX bao:  <http://www.bioassayontology.org/bao#>

#What are the kinases used in the LINCS assays measuring protein binding
#and have evidence that associates them with cancer
#and has target development Tchem?
SELECT DISTINCT ?subject_label WHERE {
 ?subject rdfs:subClassOf ?s4 .
  ?s4 owl:onProperty bao:BAO_0090013; owl:someValuesFrom bao:BAO_0002908 .
  #have evidence that associates them with cancer
  ?subject rdfs:subClassOf ?s3 .
  ?obj rdfs:subClassOf* <http://purl.obolibrary.org/obo/DOID_162> .
  ?s3 owl:onProperty dto:DTO_90100014; owl:someValuesFrom ?obj .
  ?subject rdfs:subClassOf ?s1 .
  #has target development Tchem
  ?s1 owl:onProperty <http://www.drugtargetontology.org/dto/DTO_91000020>; owl:someValuesFrom dto:DTO_00400002
      .
  ?subject rdfs:label ?subject_label
}
```

Listing 5.3: SPARQL query

This query works in two parts. In the first part we use the molecular function that is measured (i.e. protein binding) to infer the bioassays of interest. We then identify the kinases used in these assays. Finally, we get the intersection of this subgraph (i.e. subset of kinases) with the kinases that have strong evidence for associations with cancer.

This query aims to retrieve assay specific proteins based on the assays of interest. Assays with their molecular functions of interest are axiomized in BAO. Kinases have assay related axioms in LIFE which we retrieve as the second step in the query. We then explore more about the proteins by using the axioms related with their associated disease information encoded in the DTO.

First, we determine an abstract horizon between the A-box and the T-Box

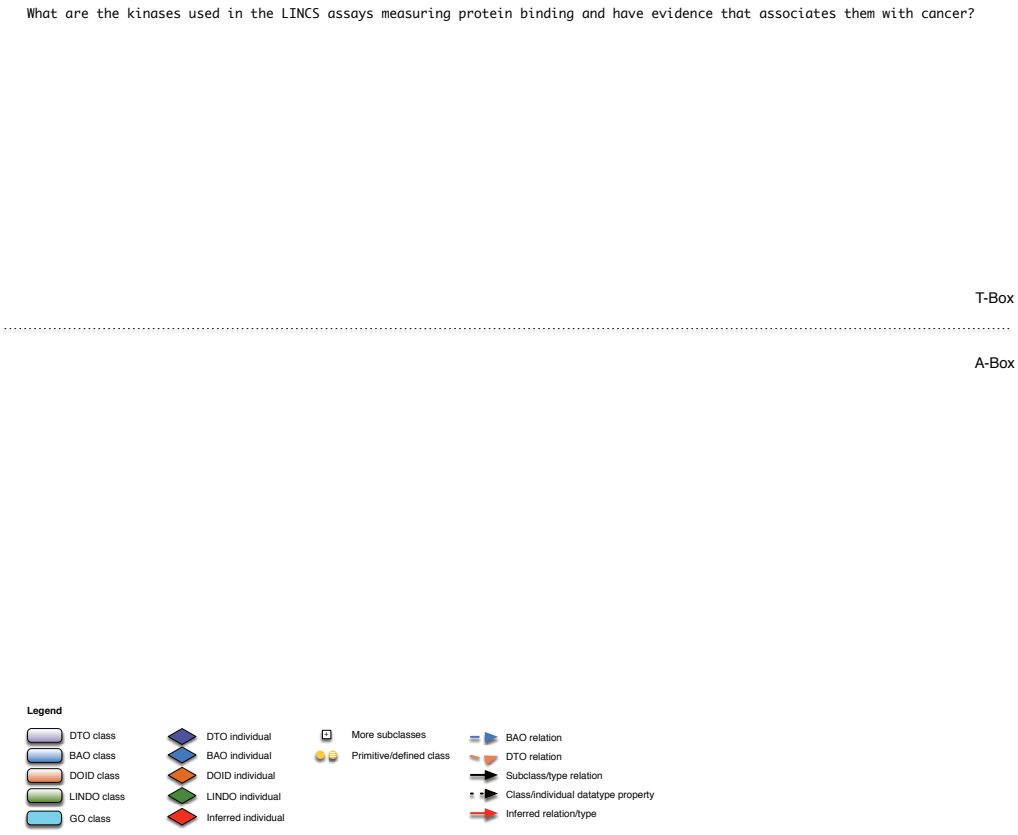



Figure 5.5: Abstract horizon between A-Box and T-Box

Second, we determine the classes that we need for this particular question.

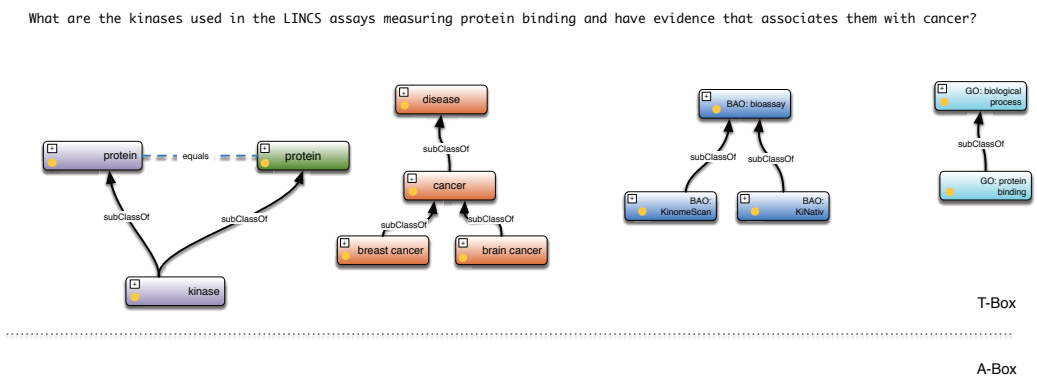




Figure 5.6: Classes from BAO, DTO, DOID and GO

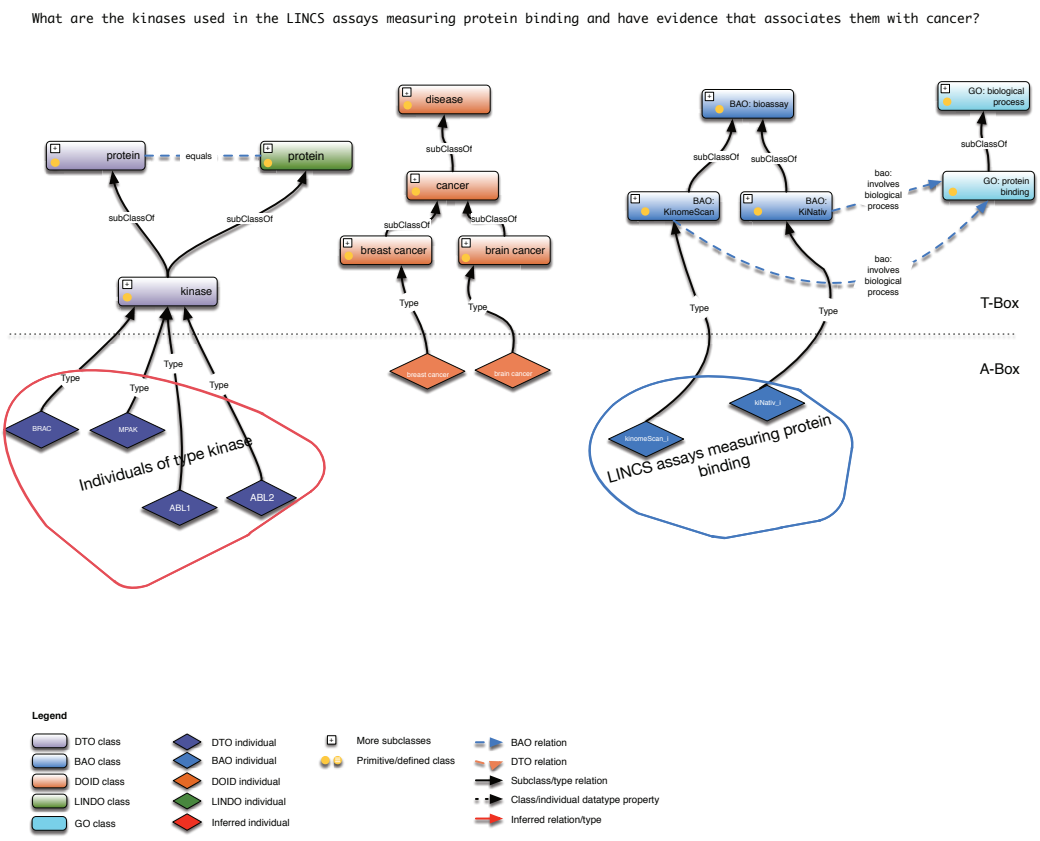



Figure 5.7: Reasoned sub-abstract classes that help for the ultimate

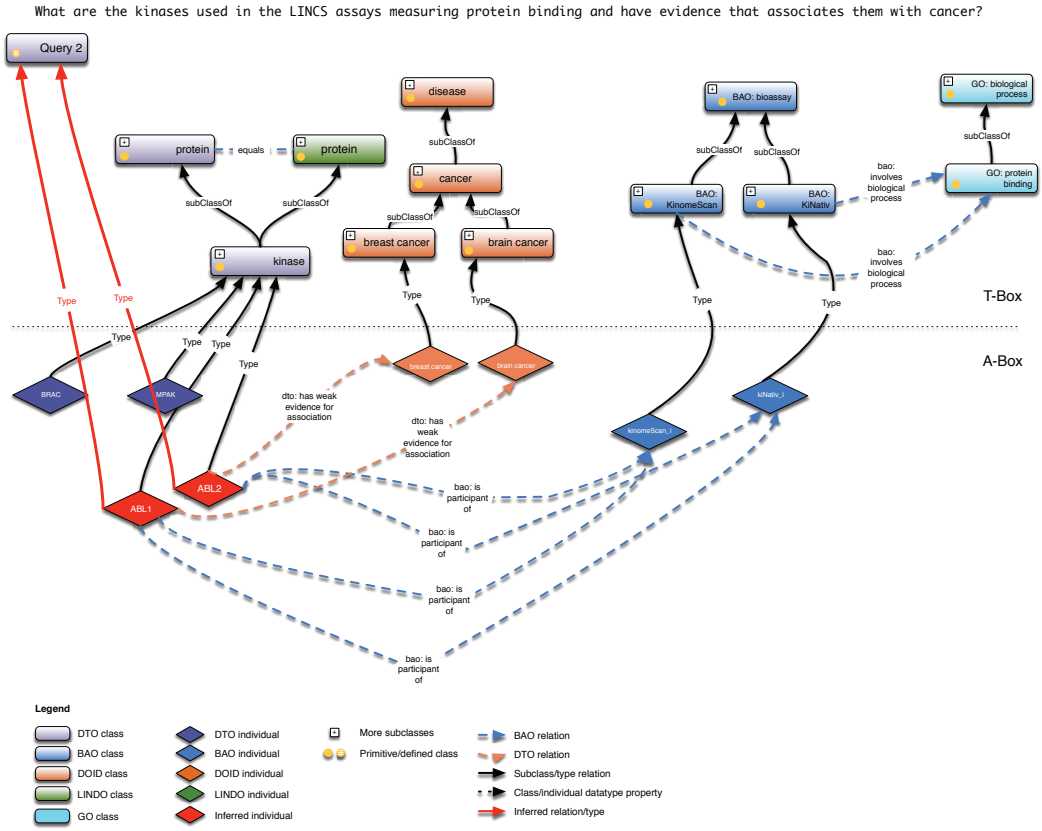



Figure 5.8: Query Result

### 5.1.4 Example Query 4

*Find proteins associated with lung with weak evidence and has target development Tchem.*

This query builds on Query 2 above. It is slightly more complicated because it adds an external ontology component, similar to Query 3. This query is generated based on the Query 2 and aims to showcase how ontologies work together as we keep adding components of interest as different ontology modules.

This query depends on the axioms related with assays from the LINCS project (found in BAO and LIFE module extracted) and the participants of these assays (extracted from the LINCS Data Portal) are added as individuals to the A-Box.

The individuals added as participants of these assays are linked to DTO vocabulary based on their UniPort IDs. In this way, these individuals related with LIFE also become connected with DTO.

The resulting participants are found based on A-Box reasoning, while their *target developmental level (TDL)* (in this case *Tchem*) is found in the T-Box axioms. The T-Box axioms related with TDL are found in DTO. The axioms are asserted between protein classes and the TDL vocabulary classes. This query also takes tissue association axioms into account during inferences with T-Box reasoning, which are axioms between protein classes of DTO and BRENDA tissues extracted for DTO.

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX dto: <http://www.drugtargetontology.org/dto/>
#proteins associated with lung with weak evidence and has target development Tchem
SELECT DISTINCT ?subject_label WHERE {


  #proteins associated with lung with weak evidence
  ?subject rdfs:subClassOf ?s3 .
  ?s3 owl:onProperty dto:DTO_90100006; owl:someValuesFrom <http://purl.obolibrary.org/obo/BTO_0000763> .
  ?subject rdfs:subClassOf ?s1 .
  #has target development Tchem
  ?s1 owl:onProperty <http://www.drugtargetontology.org/dto/DTO_91000020>; owl:someValuesFrom dto:DTO_00400002
     .
  ?subject rdfs:subClassOf ?s2 .
  #has strong evidence for disease glycogen storage disease
  ?s2  owl:onProperty <http://www.drugtargetontology.org/dto/DTO_90100015>; owl:someValuesFrom <http://purl.
     obolibrary.org/obo/DOID_2747> .
  ?subject rdfs:label ?subject_label .


}
```

```
23 LIMIT 10000
```

Listing 5.4: SPARQL query

### 5.1.5 Example Query 5

*Given participants of LINCS assays KinomeScan, L1000, and Cell Viability, find new possible drug targets for diseases of interest.*

As mentioned above, the example queries in this section could be viewed as part of the *Ontology Validation* step. In accordance with this, we realized that in order to explore from different angles, we need to connect small molecules , cell lines, and kinases and the assays that they participate in are added at the A-Box level. We further added diseases and tissues at the A-Box level and added the relationships they have with kinases and cell lines. In this way we went back to the *Meta-Data Creation and Knowledge Modeling* step and added relationships for better acquisition of knowledge and inferences based on data asserted.

With the help of A-Box and newly added relationship assertions on the A-Box level, we obtained inferences based on tissues and diseases of interest.

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX dto: <http://www.drugtargetontology.org/dto/>

SELECT DISTINCT ?subject_label WHERE {


  ?subject rdfs:subClassOf ?s3 .
  ?s3 owl:onProperty dto:DTO_90100006; owl:someValuesFrom <http://purl.obolibrary.org/obo/BTO_0000763> .
  ?subject rdfs:subClassOf ?s1 .
  ?s1 owl:onProperty <http://www.drugtargetontology.org/dto/DTO_91000020>; owl:someValuesFrom dto:DTO_00400002
     .
  ?subject rdfs:subClassOf ?s2 .
  ?s2  owl:onProperty <http://www.drugtargetontology.org/dto/DTO_90100015>; owl:someValuesFrom <http://purl.
     obolibrary.org/obo/DOID_2747> .
  ?subject rdfs:label ?subject_label .
  ?subject rdfs:subClassOf ?s1 .
  ?s4 owl:onProperty <http://purl.obolibrary.org/obo/RO_0000087>; owl:someValuesFrom dto:DTO_00000002 .
  ?subject2 rdfs:label ?subject_label2 .
  ?subject2 owl:equivalentClass ?s5 .
  ?s5 owl:intersectionOf ?list .
  ?list rdf:rest*/rdf:first ?l .
  ?l owl:onProperty dto:DTO_90000020; owl:allValuesFrom ?k .
  ?k rdfs:subClassOf* dto:DTO_61000000 .
  ?k rdfs:subClassOf?s3 .
  ?s3 owl:onProperty dto:DTO_90100056; owl:someValuesFrom <http://purl.obolibrary.org/obo/BTO_0000763> .


}
```

Listing 5.5: SPARQL query


Figure 5.9: First we designate an abstract horizon between A-Box and T-Box. In this way we aim to better show which pieces of data was added on which level so that we can trace the inferences better

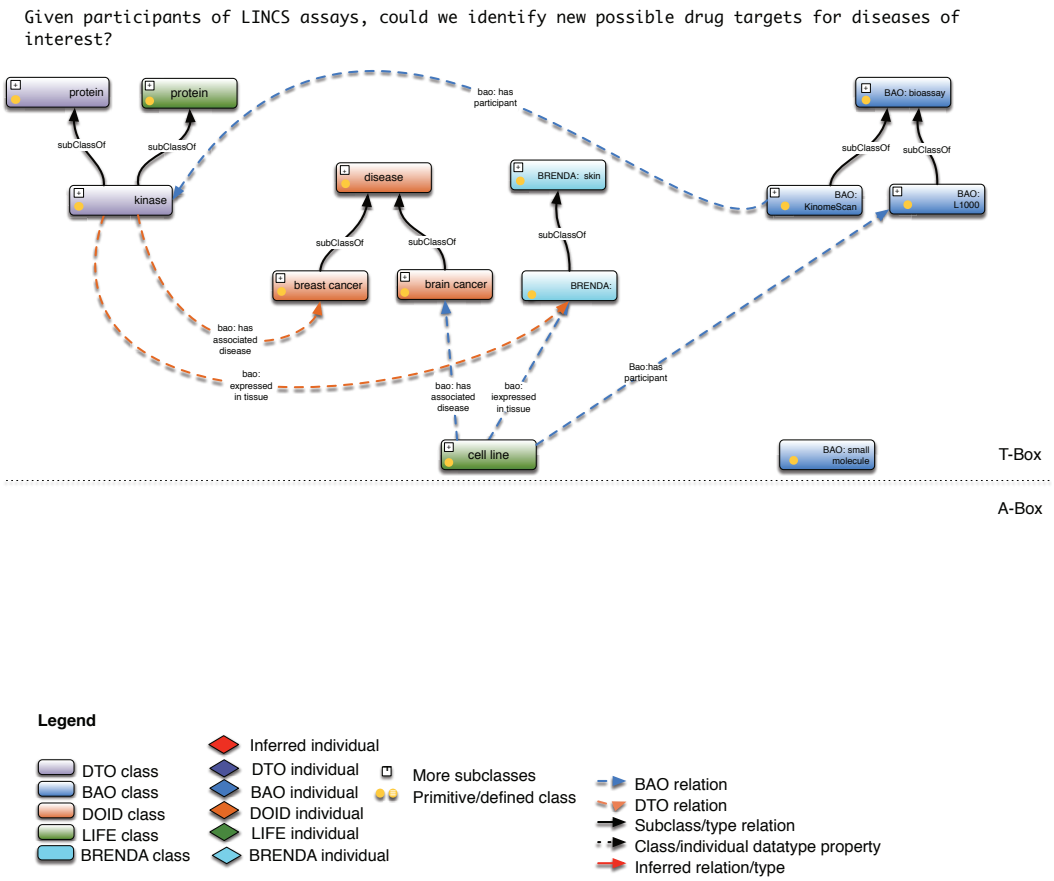



Figure 5.10: In this figure we are showing which TBox components already existed when we started thinking about this query. This is before we went back to the *Meta-Data Creation and Knowledge Modeling* step and added relationships for better acquisition of knowledge and inferences based on data asserted.

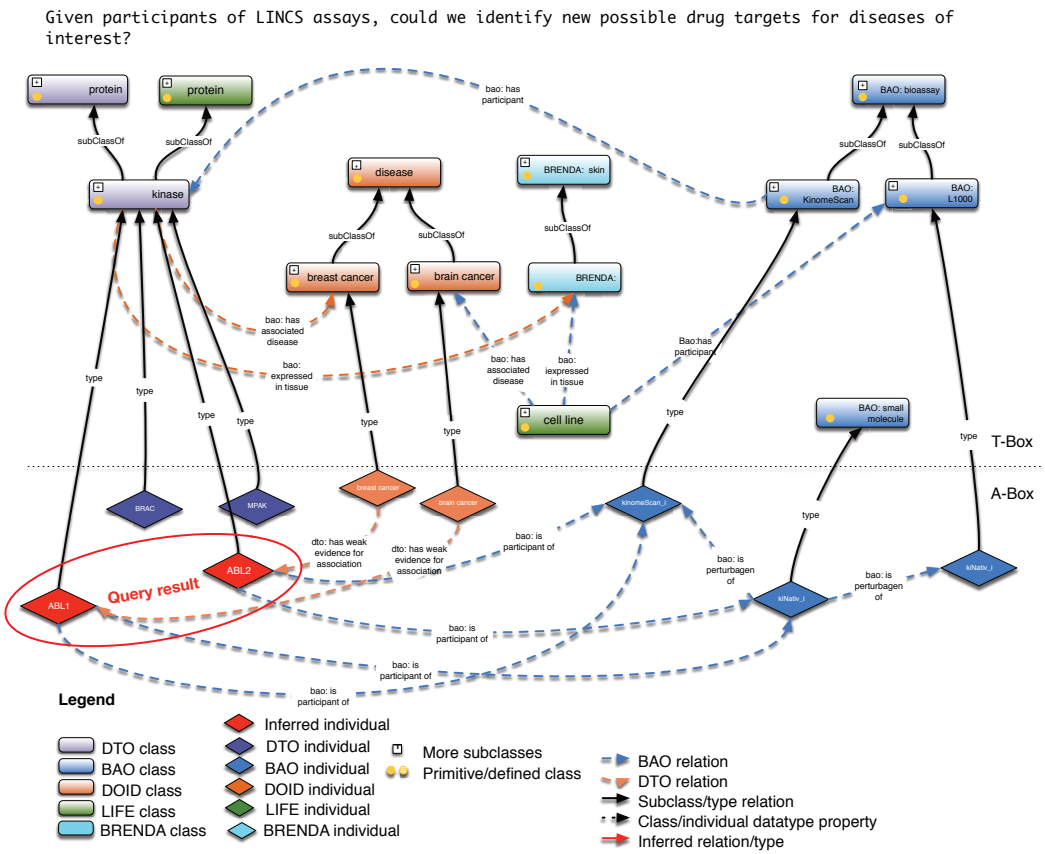



Figure 5.11: Reasoned sub-abstract classes which is provided based on the newly added assertions into the ABox and inferences in TBox and ABox levels

We used FaCT ++ 1.6.5 reasoner to reason the ontologies and queries that we created. We used Virtuoso as our local triple store and Apache Jena Fuseki as our SPARQL server to provide REST-style SPARQL HTTP Update.

FaCT++ reasoner was chosen because FaCT++ was able to handle all the ontologies used and the high expressivity levels of the ontologies. It was also chosen, because it is one of the reasoners provided by Protege. We wanted the users to be able to reuse and recreate the reasonings and queries. Furthermore, FaCT++ is an open source reasoner.

Virtuoso was used as the local triple store because of Virtuoso's ease of use with RDF data (loading and querying) and its scalability. We have explored other triple stores such as Apache Jena Fuseki for our server in the CS department of University of Miami. I also tried using Neo4j as an alternative because of the visualization options provided by Neo4j. However, Neo4j doesn't provide a variety of options available in Virtuoso. For example uploading RDF data is not a straightforward process. Moreover, the RDF data gets distorted and/or lost while uploading data. In addition to that, SPARQL queries are not supported on Neo4j. Neo4j has strong querying options, however, we wanted to have queries that are accessible and reusable for all ontology users. Virtuoso seems like the platform that can handle most of the demands of queries. However, there are still problems with some of the complex queries. Therefore I had to create various datasets for the triple stores so that queries can be performed without overwhelming the system. More research and exploration needs to be done in this area.

Our results showed us that with the three ontologies, BAO, LIFE, and DTO, we are able to connect different components about drug-discovery related data. Despite some problems with some complex queries, we are able to see that the three ontologies can provide various *views* of the knowledge based on the users' needs. We were also able to query information and retrieve related data based on different pieces of information such as assays, proteins, diseases, cell lines, etc. because of the modular architecture. This was possible because the uniform modular architecture allows us to combine different modules and create different views in order to reach the compo-

nents of interest faster. We may also choose to connect different ontologies in order to connect different pieces of data and create more meaningful pictures in the end.

BAO and DTO ontologies continue to grow both in the number of concepts (currently BAO has 7,125 concepts, DTO has 10077), and in the axioms (BAO has 98,010 and DTO has 137,648 axioms). Further axioms are waiting to be modeled and added to these ontologies, such as phosphorylation. As we add more axioms that formally define concepts, we aim to continue to help life-scientists understand, and analyze their data of interest better.

# CHAPTER 6

# Conclusion

## 6.1 Discussion and Conclusions

Life-sciences data keeps growing and fitting into the description of *big data* because it has become *high in volume*, *too complex*, and *too dynamic* for conventional data tools to store, manage, and analyze. As the growth continues, the need for building intelligent systems that will store, organize, and help scientists analyze the data is growing as well. Furthermore, challenges outlined for big data are also becoming challenges related to life-sciences data. These challenges include dealing with increasing volume, securing the data, and creating the infrastructure that allows analysis, in addition to extracting knowledge from available data [10–12, 17]. Ideally in intelligent systems that will store, organize, and help scientists analyze life-sciences data can provide an unambiguous understanding of what the data means by extracting the knowledge and providing semantic models related with the data; help build tools that could better aid life-scientists' need connect scattered pieces of information and acquire new knowledge,*inference of knowledge* that they didn't possess while building their tools and models –and achieve better acquisition and representation of knowledge while avoiding over simplification.

In this study we describe the KNowledge Acquisition and Representation Methodology (KNARM), as a guided approach involving domain experts and knowledge engineers, to build useful, comprehensive, and consistent ontologies that will enable *big data* approaches while avoiding oversimplification. This methodology is designed to help with the challenge of acquiring and representing knowledge in a systematic,

semi-automated way. This methodological approach utilizing description logic and semantic models addresses the knowledge acquisition bottleneck. KNARM is created and used for this research project by combining available methods for Database Management Systems (DBMS), Object Oriented Programming (OOP), and Knowledge Acquisition Methods. KNARM is designed, implemented, and used based on our needs and challenges related to our ongoing projects, however it can be applied and/or reused for different types of data or projects. KNARM is a hybrid methodology that combines human and machine capabilities for extracting knowledge and representing it in an ontology in a dynamic, semi-automated way. It is designed to handle both new and existing knowledge/data and allows building ontologies with high expressivity. The knowledge representation uses axioms in a systematically deepening modeling (SDM) approach for defining concepts in formal logic, detailed in sections five and six.

In this dissertation we outlined the existing efforts in generating widely accepted methodologies and best-practices principles in semantic web application and life-sciences data management domains. Although there exists some focused studies (such as OBO community tools and methodologies), we have seen that there is still room for best-practices approaches and methodologies, as this need was mentioned repeatedly in studies [2, 27, 33, 36, 37]. This review of the literature is followed by the details of KNARM's steps and how they were applied on three different projects and their respective ontologies: **B**io**A**ssay **O**ntology (BAO), and two nationwide projects, the **L**ibrary of **I**ntegrated **N** etwork-Based **C**ellular **S**ignatures (LINCS) project [71] (and its LIFE ontology) and the **I**lluminating **D**ruggable **G**enome [64] project and its (Drug Target Ontology (DTO)), which are currently creating data via wet-lab experiments and computers.

We then described the details of this methodology and its applications (i.e. the ontologies built:BAO, LIFE, and DTO). This study and applications can be viewed as a proof of concept study dealing one of the aspects related to *big data*: better extraction of knowledge by utilizing standardized vocabulary and description logic in order to aid analysis.

Building of our concordant ontologies was made possible by using KNARM and the same modular architecture consistently, and using the systematically deepening-modeling (SDM) approach with the three ontologies. Using this approach, we modeled bioassay related data in the BioAssay. For example we modeled and axiomized various bioassay related concepts such as assay format, assay design method, assay detection method and instruments in BAO. We added axioms that specify the assay participants for LINCS assays in LIFE ontology, such as kinases for KiNativ and KinomeSCAN assays. We modeled and axiomized various details about drug targets in the DTO ontology, such as their related diseases, tissues, and mutation information. With the help of our modularization approach and modular architecture we were able to align the drug targets in DTO with the various participants used in LINCS assays and LINCS assays with the general assay related concepts by using BAO. With this systematically deepening modeling approach, we aim to model and query knowledge without over- simplifying the knowledge and overwhelming the reasoners that help infer new knowledge. We exemplified some of these connections in the previous section for results. Further connections can be made using this data. One such connection would be combining the cell line and disease association information from the LINCS project with the disease, tissue, and protein information from the IDG project. In this way, we can provide more information about the cell lines from the IDG project by using the information that we acquired from the LINCS project, in an attempt to help life-scientists discover new information about their data.

Our modular architecture and SDM approach also allowed us to combine data from several related ontologies (e.g. Gene Ontology [25], Disease Ontology [19], Relationship Ontology [35]) and databases (e.g. UniProt [80], DISEASES Database [86], and TISSUES Database [87] ). Our aim was to build manageable chunks of information that are related or complementary based on the experiments performed by the data centers. By organizing the data into the modules we are able to make changes to the knowledge base easier, reuse and share the pieces of the ontologies better. Furthermore, we can use various upper ontologies, such as Basic Formal Ontology (BFO) [88] Suggested Upper Ontology (SUMO) [89], in order to merge existing efforts with our ontologies. An attempt to map BAO under BFO showed us

that there are pros and cons to using upper ontologies. The biggest motivation for using an upper ontology like BFO was to enable easier mapping of concepts from BAO to its related ontologies, for example OBI [35], better and faster, preferably in an automated way. However, there were challenges presented in our attempt to map concepts under the BFO upper ontology. The main challenge was not being able to conserve the core concepts related to BAO. This challenge was one of the reasons that led us to the build a modularization approach for BAO. This modular approach is improved for other ontologies such as LIFE and DTO. Another main challenge was the difference in the modeling approach.

In BAO we use SDM approach and model with core concepts. However, many of the *taxonomy-like ontologies* use *sentence-like* concept names while modeling their data. This difference in the approach was combined with the general approach of ontology building: i.e. more philosophical vs. more practical ontologies. In the philosophical approach used in BFO, concepts are grouped based on whether they are *continuant* or *occurrent*. This is different from our approach to model and classify concepts and knowledge based on their relation to one another and to their domain of interest.By using the shared modular approach, we were able to share and reuse data as well as mapping related data, for example our efforts to map UBERON data to BRENDA data helped us connect the tissue and organ information in the LINCS cell lines to the tissue information related with drug targets in the IDG project. The three projects, BAO, LINCS, and IDG present related data sets on different levels of detail. In their respective ontologies: BAO models the assays, LIFE provide more information about the bio-molecules and their related LINCS assays, and DTO provides details about the biomolecules and their related tissues and diseases by providing probabilistic information on the relationship level. With this semantic modeling approach, we were able to query complex information that we were not able to query before (see Use Cases section).

We demonstrated in the Results section how they work together in harmony with different queries varying in complexity from simple to complex. In simple queries we demonstrate how standardized and well-described knowledge be obtained over multiple ontologies. For example in a simple query, such as extracting LINCS assays

measuring certain biological processes or molecular functions, we demonstrated how TBox reasoning could be enough to create smaller new hierarchies of terms. In more complex queries such as finding proteins that were described as Tdark (identified using DTO axioms) also participated in LINCS assays (extracted via using axioms in BAO and LIFE), we aimed to show how possible new drug targets could be identified further in wet-lab environments via using the inferences and axioms presented in the ontologies. In most complex queries, multiple internal and external ontologies are utilized on both TBox and ABox levels. Such queries, for example identifying new possible drug targets using the disease associations found for different proteins and the protein associations found for different diseases, we wanted to demonstrate that given better connected data, we can jump between the pieces of data in order to retrieve new knowledge via inferences based on data provided in the ontologies.

In addition to the queries in the Results section, we performed a small scale experiment about the scalibility issues. During reasoning and querying we have observed that when we increase the size of the TBox and ABox, the time required for reasoning increased. We have quantified this observation in the tables given below. This experiment was aimed at understanding the possible bottlenecks in reasoning and querying of the ontologies. We used seven ontologies varying DL expressivity, size, and OWL version (please see Table 6.1 for different OWL versions, Table 6.2 for details of the ontologies and their ontology metrics). We used six reasoners, namely ELK 0.4.3 [90], FaCT++1.6.5 [91, 92], HermiT 1.3.8.413 [93], Pellet [94], Konclude [95], and KAON2 [96], to measure reasoning times over the seven ontologies as seen in Table 6.3. The capabilities of reasoners used in this experiment are briefly summarized below:

- **ELK 0.4.3 :** ELK is described as a high performance reasoner for OWL EL ontologies. The EL classification procedures, however, have several other strong indicators pointing to a good practical performance. Unlike conventional tableau-based procedures, which test unknown subsumptions by trying to construct counter-models, the EL procedures derive new subsumptions explicitly using inference rules. Although modern tableau-based reasoners, such as HermiT, FaCT++, Pellet, and RacerPro, incorporate many optimizations

that can reduce the number of subsumption tests and reuse the results of computations between the tests [34, 74, 104], they still cannot achieve the performance of specialized EL reasoners on EL ontologies

- **FaCT++1.6.5 :** FaCT++ implements a tableaux decision procedure for the well known $\mathcal{SHOIQ}$ description logic, with additional support for datatypes, including strings and integers. The system employs a wide range of performance enhancing optimisations, including both standard techniques (such as absorption and model merging) and newly developed ones (such as ordering heuristics and taxonomic classification). FaCT++ can, via the standard DIG interface, be used to provide reasoning services for ontology engineering tools supporting the OWL DL ontology language.

- **HermiT 1.3.8.413 :** HermiT is the first publicly-available OWL reasoner based on a novel *hypertableau* calculus which provides much more efficient reasoning than any previously-known algorithm. Ontologies which previously required minutes or hours to classify can often by classified in seconds by HermiT, and HermiT is the first reasoner able to classify a number of ontologies which had previously proven too complex for any available system to handle. HermiT uses direct semantics and claims that it passes all OWL 2 conformance tests for direct semantics reasoners.

- **Pellet :** Pellet is the first sound and complete OWL-DL reasoner with extensive support for reasoning with individuals (including nominal support and conjunctive query), user-defined datatypes, and debugging support for ontologies. It implements several extensions to OWL-DL including a combination formalism for OWL-DL ontologies, a non-monotonic operator, and preliminary support for OWL/Rule hybrid reasoning. Pellet is written in Java and is open source.

- **Konclude :** Konclude is a high-performance reasoner for the Description Logic $\mathcal{SROIQV}$ . The supported ontology language is a superset of the logic underlying OWL 2 extended by nominal schemas, which allows for expressing

arbitrary DL-safe rules. Konclude's reasoning core is primarily based on the well-known tableau calculus for expressive Description Logics. In addition, Konclude also incorporates adaptations of more specialised procedures, such as consequence-based reasoning, in order to support the tableau algorithm. Konclude is designed for performance and uses well-known optimisations such as absorption or caching.

- **KAON2 :** KAON2 does not implement the tableaux calculus. Reasoning in KAON2 is implemented by novel algorithms which reduce a $\mathcal{SHIQ}(\mathcal{D})$ knowledge base to a disjunctive datalog program. Unfortunately KAON2 is not compatible with OWL2, which is the OWL version used for the three ontologies in this study.

All reasoners except for Konclude and KAON2 were chosen based on their availability in Protege. In addition to their availability in Protege, reasoners were selected because of the difference in their capabilities and the algorithms they use in the background. Among all FaCT++ is the most up-to-date reasoner. Additionally, all the rest of the reasoners stated a comparison to FaCT++ reasoner, which lead us think that it's one of the state-of-the-art reasoners. One observation is that most of the reasoners available are not well maintained, last update was more than two years ago, websites and packages are not updated recently, packages can only work with previous versions of Java, web site contains broken links to interfaces and publications.

Reasoning times of the different ontologies showed a direct correlation between reasoning time and complexity. Additionally as the axiom count increases, the reasoning time increases. This was especially obvious in the ontologies we created for use cases. When we added axioms to classes related with assay participants to the ontologies (i.e. increased the size of TBox), the reasoning time was slower than when we added the same annotations as individuals and their respective annotations (i.e. increased the size of ABox). This brought up the idea of adding some of the classes, especially in DTO - for example classes created for specific proteins, as individuals. This might be especially efficient for an ontology used primarily for querying. Also

with this experiment, we have observed that some reasonings we expected to see (especially related with relationships that has inverses) are not computed based. We need to perform further experiments to identify the root of the problem, however, possible reasons for this lack of computations may be related to the size of the TBox and/or the way the semantic models were implemented (most classes included descriptive axioms to the *primitive classes* as opposed to *defined classes*). It's noteworthy to mention that reasoners that utilize OWL EL instead of OWL DL (such as ELK and Konclude) perform better in reasoning time. However, certain reasoning computations are affected by this, such as transitive relationships that we created to describe and reason over phosphorylation cascades.

Our overall aim has been to acquire the knowledge and represent it systematically in a fashion that is uniform and understandable by the many different data centers as well as the computers. In addition, we have implemented frameworks that will allow the life scientists to query, understand, and aid further analysis of their data. We understand that the most efficient way for analysis of data in this study (and *big data* in general) may not be using ontologies on their own. However, we believe that creating ontologies that artificial intelligence services and algorithms could use for inferences may lead to important findings. This aim is beyond the scope of this study. For the data frameworks, we collated data from the LINCS and IDG projects that complement and complete each other. Furthermore, we have reused extracted-data from the existing sources such as UniProt [80], Gene Ontology [25], and Disease Ontology [19]. Both LINCS and IDG projects use similar biological and chemical entities in their experiments. This allows us to combine, cross-validate, and understand data about the various entities used, such as proteins, genes, and small molecules, as well as their assays. More importantly, we aim to help life-scientists discover new data about their experiments and the experimental entities by using computer inference.

The amount of time between the public releases of the first version of BAO [82], and its second version [4] was three years. With the knowledge acquisition methodology, **KN**owledge **A**cquisition and **R**epresentation **M**ethodology (KNARM) and the semi-automated workflow we created, we are now able to revise and rebuild

| OWL Version | Usage |
|---|---|
| OWL 2 / Full | Most expressive of profiles. Might lead to problems in reasoning and inferences. |
| OWL 2 / EL | Second most expressive after OWL 2 / Full. Better reasoning performance despite large number of classes and complicated relationships. |
| OWL 2 / QL | OWL version designed to work in sync with relational databases and SQL |
| OWL 2 / RL | OWL version geared towards working with a rules engine |
| Earlier versions | OWL, OWL /Lite and other versions. These versions are not superseded by OWL 2 and its variations above. |

Table 6.1: Table summarizing OWL 2 versions and their best usage options. The different versions of OWL are s

| Ontology | DL expressivity | Num. of Classes | Num. of Axioms | Num. of Individuals |
|---|---|---|---|---|
| BAO 2.0 | $\mathcal{SOIQ}(\mathcal{D})$ | 7125 | 98,010 | 1 |
| DTO 1.0 | $\mathcal{SHI}(\mathcal{D})$ | 54343 | 137,648 | 3 |
| BAO+DTO | $\mathcal{SOIQ}(\mathcal{D})$ | 15,439 | 225,958 | 24 |
| BAO + DTO pruned and Use Case TBox Classifications | $\mathcal{SOIQ}(\mathcal{D})$ | 15,440 | 226,132 | 24 |
| BAO + DTO pruned and Use Case TBox and ABox Classifications | $\mathcal{SIQ}(\mathcal{D})$ | 15,998 | 226,750 | 199 |
| Use Cases Classes only Ontology TBox Classification | $\mathcal{SIQ}(\mathcal{D})$ | 11,898 | 184,802 | 24 |
| Use Cases Classes only Ontology TBox and ABox Classification | $\mathcal{SIQ}(\mathcal{D})$ | 11,898 | 184,802 | 4199 |

Table 6.2: Table summarizing different ontologies, expressivity levels, and ontology metrics)

| Ontology | ELK 0.4.3 | FaCT++1.6.5 | HermiT 1.3.8.413 | Pellet | Konclude | KAON2 |
|---|---|---|---|---|---|---|
| BAO 2.0 | 396 ms | 359 ms | 2047 ms | Error/Exception | 507 ms | cannot process |
| DTO 1.0 | 1925 ms | 22092 ms | Timeout after 5+min.s | 32273 ms | 999 ms | cannot process |
| BAO+DTO | 1768 ms | 33346 ms | Timeout after 5+min.s | Error/Exception | | cannot process |
| BAO + DTO pruned and Use Case TBox Classifications | 1545 ms | 33069 ms | Timeout after 5+ min.s | Error/Exception | 1773 ms | cannot process |
| BAO + DTO pruned and Use Case TBox and ABox Classifications | 1027 ms | 28862 ms | Timeout after 5+ min.s | Error/Exception | 2491 ms (Abox ignored) | cannot process |
| Use Cases Classes only Ontology TBox Classification | 2545 ms | 43078 ms | Timeout after 5+ min.s | Error/Exception | 1857 ms | cannot process |
| Use Cases Classes only Ontology TBox and ABox Classification | 1627 ms | 98862 ms | Timeout after 5+ min.s | Error/Exception | 2584 ms (Abox ignored) | cannot process |

Table 6.3: Table summarizing Run time comparisons based on different ontologies, expressivity levels, and different reasoners (reasoners selected based on reasoners available for Protege)

the ontologies, and their reusable modules within months. With this drastic improvement on ontology building process, we are now able to collaborate, revise, and improve more efficiently. On the Github location, it can be seen that various releases are only months apart [97]. Furthermore, we are leading the newly emerging *assay informatics* era to help scientists understand the available experimental data better and provide guidance for their challenge to link the experimental data and drugs to molecules, and molecules to phenotypes, diseases, and tissues. BAO provided the base line for the emerging *assay informatics* field. BAO was used in the BioAssay Research Database (BARD) software system and it was used in several projects and organizations after we initially demonstrated its use in the semantic software application BAOSearch (http://baosearch.ccs.miami.edu/). We have also used BAO to describe omics profiling assays in the LINCS program via the LINCS Information Framework (LIFE) (http://life.ccs.miami.edu/). DTO provides a formal classification of four protein families based on function and phylogenetic information. DTO describes their clinical classifications and relations to diseases and tissue expression. DTO is already used in the IDG main Portal Pharos (https://pharos.nih.gov/) and the TinX software application (http://newdrugtargets.org/) to prioritize drugs by novelty and importance. DTO is publicly available at http://drugtargetontology.org/, where it can be visualized and searched.

While technological innovations continue to drive the increase of data generation in the biomedical domains across all dimensions of big data, novel bioinformatics and computational methodologies will facilitate better integration and modeling of complex data and knowledge. Although the methodology described in this study is still a work in progress, it provided a systematic process for building concordant ontologies such as BioAssay Ontology (BAO) and Drug Target Ontology (DTO). The proposed method helps to find a starting point and facilitates the practical implementation of an ontology. The interview steps in our methodology, which involve domain experts' manual contributions are crucial to acquire the knowledge and formalize it accurately and consistently.

A critical current effort is to further formalize and automate this approach. Beyond the methodology for knowledge acquisition and semi-automated ontology build-

ing, we are also developing new tools to improve the interaction between ontology developers and users. This effort is because of rapidly advancing knowledge and the need for a more dynamic environment in which user requests can be incorporated in real time via direct information exchange with ontology developers.

## 6.2 Future Work

There is a growing interest in 'big data' research. Many research universities, such as University of Michigan [6], Stanford University [7], University of Virginia [8], among others are creating multidisciplinary centers for Data Science. There are many possible directions for the KNARM methodology and the ontologies built using this methodology.

This study and applications can be viewed as a proof of concept study dealing one of the aspects related to *big data*: better extraction of knowledge by utilizing standardized vocabulary and description logic in order to aid analysis. Here we described possible use cases in order to showcase how multiple modules and different pieces of data and knowledge could be used together to extract knowledge out of the bigger collection of data available. We believe that it's essential to describe methodologies that will develop pieces of standardized vocabulary and knowledge models that could be pluged-in to work together harmoniously.The first step before reapplying it on a new ontology would be integrating solutions for scaling the data so that more complex queries could be performed. Furthermore, based on the knowledge represented infer new and interesting knowledge.

Our aim in this study has been to acquire the knowledge and represent it systematically in a fashion that is uniform and understandable by the many different data centers as well as the computers. In addition, we have implemented frameworks that will allow the life scientists to query, understand, and aid further analysis of their data. We understand that the most efficient way for analysis of data in this study (and *big data* in general) may not be using ontologies on their own. However, we believe that creating ontologies that artificial intelligence services and algorithms

could use for inferences may lead to important findings. This aim is beyond the scope of this study.

Recently developed BioAssay Express (BAE) technology streamlines the conversion of human-readable assay descriptions to computer-readable information. BAE uses semantic standards to mark up bioprotocols, which unleashes the full power of informatics technology on data that could previously only be organized by crude text searching [98]. One of several annotation-support strategies within BAE is the use of machine learning models to provide statistically backed "suggestions" to the curator. We will describe our efforts to complement these models by applying axioms that are embedded within the underlying ontologies, which include the BioAssay Ontology (BAO), Gene Ontology (GO), Drug Target Ontology (DTO) and Cell Line Ontology (CLO). These axioms are a largely untapped resource that can be used to draw connections between biological concepts, thereby improving both curation and quality control. We have already created a resource of 3500 carefully curated assays from the PubChem collection, which we are using as a training set. We will explore how this resource will be used, in conjunction with models and axiom support, to encourage further semantic annotation of publicly available bioassay protocol data. These efforts are timely and important, as such datasets (released by both public and private organizations) are only increasing, with the volume already exceeding the ability of individual scientists to manage productively.

The outer circle of KNARM methodology , semi-automated evolving the ontology based on ontology validation, is already awarded a grant from NIH. University of Miami, Stanford University, and CDD are working together to further standardize the templates created for this work. Furthermore, the systematic approach we followed is being followed by the OBO foundry groups and we're in the process of aligning ontology design and development efforts.

Last but not least, the semi-automated ontology building and knowledge acquisition based on ontologies is patented with the BAE efforts. CDD is now attracting big pharmaceutical companies such as Astra-Zeneca and Pfizer with their assay informatics tools and their integrative approach.

Further experiments could be performed to better observe and compare reasoning times of the different ontologies created for this study. As mentioned above, in the experiment performed for this study, we have observed that some reasonings we expected to see (especially related with relationships that has inverses) are not computed based. Further experiments to identify the root of this problem could be performed. These could include comparing computations related to the size of the TBox and/or the way the semantic models were implemented (currently most classes included descriptive axioms to the *primitive classes* as opposed to *defined classes*). An idea for improving reasoning times would be employing parallel reasoning for the merged ontologies. In this was reasoners such as Hermit might perform better.

The long-term prospect is a global dynamic knowledge framework to integrate and model increasingly big datasets to help solving the most challenging biomedical research problems. With this methodology, KNARM, we can try to apply it to different projects and their ontologies. The aim would be to better integrate and model increasingly big datasets to help solving the most challenging biomedical research problems.

# APPENDICES

## .1 In-House Structured Interview and Meta-Data Creation Documents for LIFE and BAO

| KINOMEScan | |
|---|---|
| Participants | immobilized kinase substrate |
| Endpoint | % inhibition |
| Endpoint Mode of Action | competitive binding |
| Biological Process | protein binding, kinase inhibition (could be others) |
| Target | kinase |
| | |
| Detected Entity | DNA |
| Perturbagen | small molecule |
| Measured Entity | kinase |
| Detection Method | qPCR |
| Assay Design Method | immobilized kinase substrate competative binding |
| Time Points | 1 |
| Concentration of Perturbagen | 1 |
| Repeats | 1 |
| Other Reported Concepts | N/A |
| Assay Format | biochemical format |
| Model System | biochemical model system |
| Link | www.discoverx.com/**Kinomescan**/<br>https://lincs.hms.harvard.edu/data-types/drug-target-interactions/ |

| KiNativ | |
|---|---|
| Participants | kinase, ATP-probe (tagged), small molecule |
| Endpoint | Kd? |
| Endpoint Mode of Action | competitive binding |
| Biological Process | protein binding, kinase inhibition (could be others) |
| Target | kinase |
| | |
| Detected Entity | ATP? |
| Perturbagen | small molecule |
| Measured Entity | tagged (labeled) kinase fragment? |
| Detection Method | mass spectometry |
| Assay Design Method | no method is defined yet? , proteomics method -- needs to be defined! |
| Time Points | 1 |
| Concentration of Perturbagen | 4 |
| Repeats | 1 |
| Other Reported Concepts | labeling site (activation loop confirmation) , UniRef ID for kinase, peptide sequence of the labeled binding site |
| Assay Format | cell based format |
| Model System | cell lysate |
| Link | https://lincs.hms.harvard.edu/data-types/drug-target-interactions/<br>www.**kinativ**.com/ |

### .1.1 LINCS Assay data

| LINCS-specific secondary name (LIFEo, BAO optic | LINCS datasets (how they are called i |
|---|---|
| KiNativ | KiNativ |
| KINOMEScan | KINOMEScan |
| Apoptosis and Mitosis assay (Michison) | Tang Mitosis/Apoptosis |
| | Tang Proliferation/Mitosis |
| Apoptosis and Mitosis assay (Michison) | Moerke 2 Color Apoptosis |
| | Moerke 3 Color Apoptosis |
| Cell growth inhibition assay (MGH / Sanger) | MGH (CMT) Growth Inhibition |
| L1000 assay (Broad) | L1000 Transcriptional Response |
| CSR assay (Sorger) | Cue Signal Response |

**Tabs in LIFEwrx**

**Protein:**

In addition to Uma's comments either change the name of proteins, or remove the modificatior add organism after ID

Do we want to search with Kinase family, group, domain, etc?

Kinase category = other ?? - just leave blank or say N/A

**Cell Line**

In addition to Uma's comments:

Like Uma said DOID+Disease

Do we need "type"?

**Gene**

Organism : why are they numbers? We need names of organisms.

**Assay level: put markers?**

Assay tab, either to left most, and data right most or put Data and Assay next to each other

| **Tabs in LIFEwrx** |
| --- |
| Protein, Compound, Assay, Data (we have cell line info for this, where do we see it?, when we load) |
| Protein, Compound, Assay, Data |
| Cell line, Compound, Assay, Data, markers ? |
| Cell line, Compound, Assay, Data, markers ? |
| Cell line, Compound, Assay, Data, markers ? |
| Cell line, Compound, Assay, Data, markers ? |
| Cell line, Compound, Assay, Data |
| Cell line, Gene, Compound, Assay, Data |
| Cell line, Protein, Ligand, Compound, Assay, Data |

modification and mutation

**HMS LINCS Dataset Descriptions---January 21, 2014**

**KINOMEscan Assay**: The HMS LINCS Database currently holds 150 KINOMEscan datasets—95 datasets run at a single dose and 55 datasets run as dose response experiments. The KINOMEscan assay is run as a service by DiscoveRx (http://www.discoverx.com/services/drug-discovery-development-services/kinase-profiling/kinomescan) and measures small molecule (drug) binding to the ATP-binding site of purified kinase domains via a competition assay. A panel of >400 kinases is tested for each dataset. A typical single dose KINOMEscan experiment is described at https://lincs.hms.harvard.edu/db/datasets/20020/; a typical dose response KINOMEscan assay is described at https://lincs.hms.harvard.edu/db/datasets/20146/.

**KiNativ Assay**: The HMS LINCS Database currently holds 19 KiNativ assay results—6 datasets run as dose response experiments and 13 run at a single dose. The KiNativ assay is run as a service by KiNativ (http://www.kinativ.com/) and measures small molecule (drug)-kinase interactions in cell lysates using mass spec. Binding to ~200-300 independent kinase binding sites is monitored in each experiment. A typical dose-response KiNativ assay is described at https://lincs.hms.harvard.edu/db/datasets/20087/. A typical single dose KiNativ experiment is described at https://lincs.hms.harvard.edu/db/datasets/20093/.

**ELISA assay**: The HMS LINCS Database currently holds 2 datasets from plate-based ELISA assays:
-Dataset # 20137 https://lincs.hms.harvard.edu/db/datasets/20137/
-Dataset # 20140 https://lincs.hms.harvard.edu/db/datasets/20140/

**Microscopy/Imaging Assay**: HMS LINCS DB currently holds 6 datasets produced using high throughput microscopy/imaging assays where cells were fixed and stained with fluorescent dyes and/or immunostained in order to monitor various cell properties (shape, size, nuclear area) or protein levels (in the immunofluorescence experiments). The 6 datasets are:
-Dataset # 20001 https://lincs.hms.harvard.edu/db/datasets/20001/
-Dataset # 20002 https://lincs.hms.harvard.edu/db/datasets/20002/
-Dataset # 20003 https://lincs.hms.harvard.edu/db/datasets/20003/
-Dataset # 20004 https://lincs.hms.harvard.edu/db/datasets/20004/
-Dataset # 20138 https://lincs.hms.harvard.edu/db/datasets/20138/
-Dataset # 20139 https://lincs.hms.harvard.edu/db/datasets/20139/

**Analysis dataset**: Currently the HMS LINCS DB holds one dataset that is the product of re-analysis of experimental drug dose-response data published by non-LINCS investigators. Data in this analysis dataset were published in Fallahi-Sichani et al. (2013) Nature Chemical Biology. PMID: 24013279 and can be found at https://lincs.hms.harvard.edu/db/datasets/20120/.

**MGH-CMT Growth Inhibition Assay**: The CMT platform uses a DNA stain or Resazurin based assay to determine cell viability following 72H compound treatment. For most compounds, the effects on cell growth for hundreds of cell line are reported; compounds were tested at either 3 or 9 different doses. A typical MGH-CMT Growth Inhibition assays from the HMS LINCS DB is described at https://lincs.hms.harvard.edu/db/datasets/20010/main.

**Microfluidic Assay**: Three datasets produced using a single-cell microfluidics assay are present in the HMS LINCS DB, all produced by the Yale LINCS U01 Center and described in Lu, et al. (2013) Analytical Chem. PMID: 23339603:
-Dataset # 20121 https://lincs.hms.harvard.edu/db/datasets/20121/
-Dataset # 20122 https://lincs.hms.harvard.edu/db/datasets/20122/
-Dataset # 20123 https://lincs.hms.harvard.edu/db/datasets/20123/

**Bead-based ELISA assay**: Three datasets produced using bead-based ELISA assays (Luminex) monitoring phosphorylation state of intracellular proteins in liver cells after perturbation are available for download via the HMS LINCS website:
-Liver CSR 1 dataset: http://lincs.hms.harvard.edu/data/repository/liver-csr-1-2/
-Liver CSR 2 dataset: http://lincs.hms.harvard.edu/data/repository/liver-csr-1-2/
-Liver CSR 3 dataset: http://lincs.hms.harvard.edu/data/repository/liver-csr-3/

| KINOMEScan | |
|---|---|
| Participants | immobilized kinase substrate |
| Endpoint | % inhibition |
| Endpoint Mode of Action | competitive binding |
| Biological Process | protein binding, kinase inhibition (could be others) |
| Target | kinase |
| | |
| Detected Entity | DNA |
| Perturbagen | small molecule |
| Measured Entity | kinase |
| Detection Method | qPCR |
| Assay Design Method | immobilized kinase substrate competative binding |
| Time Points | 1 |
| Concentration of Perturbagen | 1 |
| Repeats | 1 |
| Other Reported Concepts | N/A |
| Assay Format | biochemical format |
| Model System | biochemical model system |
| Link | www.discoverx.com/**Kinomescan**/<br>https://lincs.hms.harvard.edu/data-types/drug-target-interactions/ |

| KiNativ | |
|---|---|
| Participants | kinase, ATP-probe (tagged), small molecule |
| Endpoint | Kd? |
| Endpoint Mode of Action | competitive binding |
| Biological Process | protein binding, kinase inhibition (could be others) |
| Target | kinase |
| | |
| Detected Entity | ATP? |
| Perturbagen | small molecule |
| Measured Entity | tagged (labeled) kinase fragment? |
| Detection Method | mass spectometry |
| Assay Design Method | no method is defined yet? , proteomics method -- needs to be defined! |
| Time Points | 1 |
| Concentration of Perturbagen | 4 |
| Repeats | 1 |
| Other Reported Concepts | labeling site (activation loop confirmation) , UniRef ID for kinase, peptide sequence of the labeled binding site |
| Assay Format | cell based format |
| Model System | cell lysate |
| Link | https://lincs.hms.harvard.edu/data-types/drug-target-interactions/<br>www.**kinativ**.com/ |

| Mitosis/Apoptosis | | |
|---|---|---|
| Participants | cell line (33) , small molecule (%interphase = total # of cells - (apoptotic+dead+mitotic)), (% apoptotic = NucView positive), (%dead = NucView and LysotrackerRed negative), (%mitotic = Lysotracker Red) | |
| Endpoint | | The LINCs web site suggests they use antibodies to many targets, such as phospho-ERK |
| Endpoint Mode of Action | inhibition | |
| Biological Process | apoptotic process, mitotic cell cycle, cell death | |
| Target | phenotypic assay (but implicitly some molecular entity in the cell) | |
| Perturbagen | SM | |
| Measured Entity | | |
| Detected Entity | caspase 3 , DNA, lysosome cell morphology is not an entity, but should we integrate it somehow? | |
| Detection Method | microscopy | |
| Assay Design Method | cell-cycle progression assessment method; phospholipid redistribution estimation; chromatin condensation visualization; mitochondrial membrane potential assessment; caspase activity determination | |
| Time Points | 3 (24 hrs. 48 hrs, 72 hrs) | |
| Concentration of Perturbagen | 12 | |
| Repeats | | 1 |
| Other Reported Concepts | cell-count | |
| Assay Format | cell based format | |
| Model System | cell line | |
| Link | https://lincs.hms.harvard.edu/data-types/microscopy-imaging/ | https://lincs.hms.harvard.edu/db/datasets/20003/ |

| Proliferation/Mitosis | | |
|---|---|---|
| Participants | cell line (17) , small molecule (EdU reaction cocktail, details?) (%proliferating = EdU positive, passes EdU threshold), (%mitotic = MPM2 positive), (%non-mitotic (% S arrested) = EdU positive, but not MPM2 positive, over time), (% G2 arrested = EdU and MPM2 positive, over time) | Should we label 'marker' and say has marker instead of has participant |
| Endpoint | | |
| Endpoint Mode of Action | inhibition | |
| Biological Process | mitotic cell cycle, mitotic S phase, mitotic G2 phase, proliferation | |
| Target | phenotypic assay (but implicitly some molecular entity in the cell) | |
| Perturbagen | SM | |
| Detected Entity | DNA (EdU), mitotic protein(MPM2) | MPM2 stains topoisomerase II alpha, microtubule associated protein, NimA and two Cdc2- inhibitory kinase http://e-immunohistochemistry.info/web/MPM-2.htm |
| Measured Entity | cell-cycle phase? | |
| Detection Method | microscopy imaging | |
| Assay Design Method | see the note below | |
| Time Points | 3 (24 hrs. 48 hrs, 72 hrs) | |
| Concentration of Perturbagen | 12 | |
| Repeats | 1 | |
| Other Reported concepts | cell-count | |
| Assay Format | cell based format | |
| Model System | cell line | |
| Link | https://lincs.hms.harvard.edu/data-types/microscopy-imaging/ | https://lincs.hms.harvard.edu/db/datasets/20004/ |

NOTE:
The problem with these assays is that we defined everything under 'mitotic cell cycle assay' and 'apoptosis assay'. However, their upper classes are different and unrelated in the ontology. In addition to that, proliferation/mitosis and mitosis/apoptosis assays' methods for cell cycle are different, but all are listed under one assay.
here are the assay methods listed and their definitions:
Phospholipid Redistribution Estimation: In a viable cell, the phospholipid, phosphatidyl serine (PS) is localized on the inner leaflet of the cell membrane facing the cytosol. When the cell is undergoing apoptosis, the cell membrane loses its phospholipid asymmetry and exposes PS on the outer layer of the membrane. Annexin V specifically binds PS. Hence, apoptotic cells can be detected by staining with FITC conjugated Annexin V followed by 'flow cytometry'.
Chromatin Condensation Visualization: Chromatin condensation is one of the morphological changes that occur during apoptosis. It is also known as pyknosis and can be visualized by staining with dyes followed my microscopy.
Caspase Activity Determination: Caspases are cysteine proteases which play an important role in apoptosis. They are synthesized and stored in cells as inactive procaspases. Once apoptosis is initiated in a cell, the procaspases get activated by proteolytic cleavage. This sets in motion a caspase cascade, which leads to the activation of proteins involved in the chromatin fragmentation and finally culminates in cell death.
Membrane Permeability Assessment: Live cells with intact membranes have the ability to exclude dyes that easily penetrate dead or damaged cells. Examples of the dyes include propidium iodide, 7-amino actinomycin D, and trypan blue. Cells stained with the former two can be quantified by flow cytometry, while the latter is by manual counting using hemocytometer.
Mitochondrial Membrane Potential Assessment: Activity of the mitochondrial respiratory chain results in the generation of a proton gradient which consists of two components: a membrane potential and a pHgradient. Disruption of delta-psi has been shown to be one of the first intracellular changes following the onset of apoptosis. Delta-psi is generally quantitated by staining with specific dyes whose binding is dependent on the potential and less commonly, by electrophysiological techniques.
In addition to these methods different cell cycle phases are detected by using different assay methods:
S Phase Assessment Method: With the help of the EdU, the cells' nucleic DNAs are stained. By using florescence microscopy, stained nuclei are detected and counted to identify cells that are undergoing the S phase of the mitotic cell cycle. This method is mainly used in the Proliferation/Mitosis Assay and the presence of EdU is evaluated with the presence or the absence of the MPM2 antibody. EdU positive cells are considered 'proliferating' and are associated with the 'percent proliferating' endpoint.
M Phase Assessment Method: With the help of the MPM-2 antibody and LysoTracker Red the M phase of mitotic cycle is detected. By using the MPM-2 antibody, the phosphorylated kinases of the M phase are detected with the help of fluorescence microscopy. The cells that are MPM-2 positive are either G2 arrested or mitotic and are associated with the endpoints 'percent G2 arrested' and 'percent mitotic'. For the cells to be labeled as 'G2 arrested', the cells need to be stained by both EdU and MPM-2 over time. LysoTracker Red is also used to identify mitotic cells, but it is only used in the 'Mitosis/Apoptosis' experiments. LysoTracker Red stains acidic organelles, mostly lysosomes
By using the above listed methods following endpoints are provided:
Same thing with the endpoints. They are not specifically defined for the two assay types.
*Percent Interphase:* This is determined by subtracting the apoptotic, dead, and mitotic cells from the total number of cells.
*Percent Apoptotic Cells:* This determined by the number of NucView positive cells.
*Percent Dead:* This endpoint is calculated by using the NucView and LysoTracker negative cells.
*Percent Mitotic Arrested Cells:* This endpoint is reached by using LysoTracker Red stained cells.
*Percent DNA replicated Cells (Percent Proliferating):* This represents the percentage of cells in a cell population that have undergone the S phase (completed DNA replication) in preparation for mitosis, as indicated by staining positive for EdU.
*Percent Mitotic Cells:* This represents the percentage of cells in a cell population that are in the M phase of the cell cycle. The cells in this phase can be detected using various markers, one of which is the characteristic phosphorylation of certain proteins that stain positive with MDM2 antibody.
*Percent G2 Arrested Cells:* This represents the percentage of cells in a cell population that have undergone the DNA synthesis in the S phase as indicated by staining positive for EdU, but have not undergone the M phase as they stained negative for MDM2. In other words, these cells were arrested in the G2/M cell cycle check-points after treatment with the respective perturbagen in the assay.

NOTE: these assays are not defined in BAO yet, the only assay related to these is the apoptosis assay, can these go as subclasses under that?

3 Color Apoptosis

| | | |
|---|---|---|
| Participants | Small Molecule | |
| Detected Entity | cell / nucleus | specific markers related to detection, endpoint and the biolgical process |
| Endpoint | % apoptosis, % dead/dying | |
| Endpoint Mode of Action | inhibition | |
| Biological Process | apoptotic process, cell death by other means | |
| Target | ? | |
| Perturbagen | small molecule | |
| Measured Entity | DNA (Hoechst 33342), active caspase 3, dead/dying cells (To-Pro3) | |
| Detection Method | microscopy imaging | |
| Assay Design Method | | needs to be a name to refer to the specific markers used and how it is detected |
| Time Points | Uma says 1 but I saw an assay with 2? (24 & 48 hrs) | Nuclei are counted to get total number of cells, caspace3 to count apoptotic cells, topro3 allows infereace about cells dying by other means (autophagy? Necrosis?) |
| Concentration of Perturbagen | 12 | |
| Repeats | 1 | |
| Other Reported Concepts | cell count | |
| Assay Format | cell based format | |
| Model System | cell line | |
| Link | http://lincs.hms.harvard.edu/db/datasets/20002/ | |

2 Color Apoptosis

| | | |
|---|---|---|
| Participants | Small Molecule | |
| Detected Entity | cell / nucleus | specific markers related to detection, endpoint and the biolgical process |
| Endpoint | % apoptosis | Nuclei are counted to get total number of cells, caspace3 to count apoptotic cells |
| Endpoint Mode of Action | inhibition | |
| Biological Process | apoptotic process | |
| Target | | |
| Perturbagen | small molecule | |
| Measured Entity | DNA (Hoechst 33342), active caspase 3 | |
| Detection Method | microscopy imaging | |
| Assay Design Method | | needs to be a name to refer to the specific markers used and how it is detected |
| Time Points | 3 (24, 48, 72 hrs) | |
| Concentration of Perturbagen | 13 | |
| Repeats | 1 | |
| Other Reported Concepts | cell count | |
| Assay Format | cell based format | |
| Model System | cell line | |
| Link | http://lincs.hms.harvard.edu/db/datasets/20001/ | |

Gene Expression

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | entity | role | entity | related to | via | |
| Participants | gene (978), siRNA, shRNA, cDNA, cell line, SM | RNAi | screened entity | RNA | Gene | codes | |
| | | small molecule | screened entity | RNAi | Gene | target | note that the detected mRNA / transcription is not the same as the target, but both relate to a (different) gene |
| | | Gene | RNAi target | protein | Gene | code | general, not specific to that assay |
| | | mRNA | measured entity | small molec | protein | target | general, not specific to that assay |
| | | cell/ cell line | model system | cell line | disease | (complex model; no | general, not specific to that assay |
| | | tagged bead | detected entity | | | | |

| | | |
|---|---|---|
| Endpoint | fold change (over expression and under expression), expression level | |
| Endpoint Mode of Action | transcription regulation | inhibition or activiation |
| Biological Process | gene expression, transcription regulation, transcription | |
| Target | gene here is the indicator or the biological process, like the m | not known except for RNAis but even these can have off-targets (it's a cell--based assay) |
| Perturbagen | siRNA, shRNA, SM | Is the perturbagen the cDNA, Uma? I think we should consider the cell line as being modified by the cDNA construct. |
| Detected Entity | just seq. then DNA seq. , if microarray then mRNA or maybe both? | |
| Measured Entity | (maybe mRNA?) | we are measuring RNA indirectly via tagged immobalized DNA |
| Detection Method | Flow cytometry | |
| Assay Design Method | dna-coated bead based method | |
| Time Points | 6 (6, 24, 75, 96, 120, 144) | |
| Concentration of Perturbagen | 1?? | |
| Repeats | 1 | |
| Other Reported Concepts | | |
| Assay Format | cell based format | |
| Model System | cell line | |
| Link | http://lincscloud.org/the-data-set/the-current-data/ | |

Cue Signal Response

| | |
|---|---|
| Participants | protein (9), carbohydrte(ligand) (1), small molecules |
| Endpoint | fold change of protein phosphorylation, fold change of cytokine secretion fold inhibition / activation; percent inhibitio |
| Endpoint Mode of Action | fold change |
| Biological Process | phosphorylation, cytokine secretion |
| Target | not known; can be kinase or other target; molecular targets need to be described indirectly |
| Detected Entity | phosphorylated proteins, cytokine (protein) |
| Perturbagen | small molecule, protein, lipid |
| Measured Entity | phosphorilated proteins, cytokine (protein) phosphoprotein / phosphorylated protein; c |
| Detection Method | fluorescence intensity |
| Assay Design Method | antibody-coated bead based method? Maybe ELISA in later assays are they using specific antibodies and the ac |
| Time Points | 3(0 min, 180 min, 1440 min) |
| Concentration of Perturbagen | 1 |
| Repeats | 1 |
| Other Reported Entities | |
| Assay Format | cell line based |
| Model System | cell line |
| Link | http://lincs.hms.harvard.edu/data/repository/liver-csr-1-2/ |

participants

molecular entity role(s)

ent inhibition / activation?

small molecule screened perturbagen

protein (growth factor, cyt inducer; cytokine, growth factor, ligand, perturbagen

need to be linked to antibody for detection We can just call it anti-AKT, etc until we get the correct antibody name

how do we name this?

# .2 Structured Interview - Feedback from Harvard Medical School for Aligning their definitions with ours for LIFE and BAO

**HMS LINCS Dataset Descriptions---January 21, 2014**

**KINOMEscan Assay**: The HMS LINCS Database currently holds 150 KINOMEscan datasets—95 datasets run at a single dose and 55 datasets run as dose response experiments. The KINOMEscan assay is run as a service by DiscoveRx (http://www.discoverx.com/services/drug-discovery-development-services/kinase-profiling/kinomescan) and measures small molecule (drug) binding to the ATP-binding site of purified kinase domains via a competition assay. A panel of >400 kinases is tested for each dataset. A typical single dose KINOMEscan experiment is described at https://lincs.hms.harvard.edu/db/datasets/20020/; a typical dose response KINOMEscan assay is described at https://lincs.hms.harvard.edu/db/datasets/20146/.

**KiNativ Assay**: The HMS LINCS Database currently holds 19 KiNativ assay results—6 datasets run as dose response experiments and 13 run at a single dose. The KiNativ assay is run as a service by KiNativ (http://www.kinativ.com/) and measures small molecule (drug)-kinase interactions in cell lysates using mass spec. Binding to ~200-300 independent kinase binding sites is monitored in each experiment. A typical dose-response KiNativ assay is described at https://lincs.hms.harvard.edu/db/datasets/20087/. A typical single dose KiNativ experiment is described at https://lincs.hms.harvard.edu/db/datasets/20093/.

**ELISA assay**: The HMS LINCS Database currently holds 2 datasets from plate-based ELISA assays:
-Dataset # 20137 https://lincs.hms.harvard.edu/db/datasets/20137/
-Dataset # 20140 https://lincs.hms.harvard.edu/db/datasets/20140/

**Microscopy/Imaging Assay**: HMS LINCS DB currently holds 6 datasets produced using high throughput microscopy/imaging assays where cells were fixed and stained with fluorescent dyes and/or immunostained in order to monitor various cell properties (shape, size, nuclear area) or protein levels (in the immunofluorescence experiments). The 6 datasets are:
-Dataset # 20001 https://lincs.hms.harvard.edu/db/datasets/20001/
-Dataset # 20002 https://lincs.hms.harvard.edu/db/datasets/20002/
-Dataset # 20003 https://lincs.hms.harvard.edu/db/datasets/20003/
-Dataset # 20004 https://lincs.hms.harvard.edu/db/datasets/20004/
-Dataset # 20138 https://lincs.hms.harvard.edu/db/datasets/20138/
-Dataset # 20139 https://lincs.hms.harvard.edu/db/datasets/20139/

**Analysis dataset**: Currently the HMS LINCS DB holds one dataset that is the product of re-analysis of experimental drug dose-response data published by non-LINCS investigators. Data in this analysis dataset were published in Fallahi-Sichani et al. (2013) Nature Chemical Biology. PMID: 24013279 and can be found at https://lincs.hms.harvard.edu/db/datasets/20120/.

**MGH-CMT Growth Inhibition Assay**: The CMT platform uses a DNA stain or Resazurin based assay to determine cell viability following 72H compound treatment. For most compounds, the effects on cell growth for hundreds of cell line are reported; compounds were tested at either 3 or 9 different doses. A typical MGH-CMT Growth Inhibition assays from the HMS LINCS DB is described at https://lincs.hms.harvard.edu/db/datasets/20010/main.

**Microfluidic Assay**: Three datasets produced using a single-cell microfluidics assay are present in the HMS LINCS DB, all produced by the Yale LINCS U01 Center and described in Lu, et al. (2013) Analytical Chem. PMID: 23339603:
-Dataset # 20121 https://lincs.hms.harvard.edu/db/datasets/20121/
-Dataset # 20122 https://lincs.hms.harvard.edu/db/datasets/20122/
-Dataset # 20123 https://lincs.hms.harvard.edu/db/datasets/20123/

**Bead-based ELISA assay**: Three datasets produced using bead-based ELISA assays (Luminex) monitoring phosphorylation state of intracellular proteins in liver cells after perturbation are available for download via the HMS LINCS website:
-Liver CSR 1 dataset: http://lincs.hms.harvard.edu/data/repository/liver-csr-1-2/
-Liver CSR 2 dataset: http://lincs.hms.harvard.edu/data/repository/liver-csr-1-2/
-Liver CSR 3 dataset: http://lincs.hms.harvard.edu/data/repository/liver-csr-3/

# LINCS Data Levels

LINCS DSGC and BD2K-LINCS DCIC (support at lincs-dcic.org)
{working version: last modified May 16 2016}
{document content will reside in: http://www.lincsproject.org/data/}

The LINCS Data and Signature Generation Centers (DSGCs) produce a variety of data for the library. For such data to be standardized, integrated, and coordinated in a manner that promotes consistency and allows comparison across different cell types, assays and conditions, the BD2K-LINCS DCIC together with the DSGCs develop and employ data standards.

Once collected, LINCS data is made available to the research community in various formats so that it can be used in different types of analyses.

The data standards page describes the data structures that are being developed by the LINCS Data Working Group.

The data releases page describes the collections of data released and planned to be released to the public by the LINCS consortia with instruction on how to access, download and cite it.

## Data Levels

The LINCS resource (and the resulting data matrix) has three dimensions: cell types, perturbations, and assay types. LINCS approaches this problem using tools from systems biology, chemical biology, computational biology, and other disciplines, including both high-throughput experimentation and sophisticated mathematical analysis. The concept of data levels is also borrowed from the success of this approach by The Cancer Genome Atlas (TCGA) project. Definitions for data levels for all the LINCS assays are currently being developed by the BD2K-LINCS DCIC and the LINCS DSGCs and will be posted here soon.

## Data Levels per assay type

| Assay Type | Center | Data Level | Result Type | File Type | Important Metadata |
|---|---|---|---|---|---|

| | | | | | |
|---|---|---|---|---|---|
| KINOMEScan | HMS LINCS | 2 | Relative kinase inhibition (normalized to negative control) | API; Download (xlsx, csv) | • Small molecules<br>• Proteins |
| KINOMEScan | HMS LINCS | 3 | Kds were determined using 11 serial threefold dilutions of test compound and a DMSO control. | API; Download (xlsx, csv) | • Small molecules<br>• Proteins |
| KiNativ | HMS LINCS | 2 | Protein binding profile | API; Download (xlsx, csv) | • Cell lines<br>• Small molecules<br>• Proteins |
| L1000 mRNA profiling assay | Broad T LINCS | 1 | Raw, unprocessed flow cytometry data from Luminex scanners. One LXB file is generated for each well of a 384-well plate, and each file contains a fluorescence intensity value for every observed analyte in the well | LXB | • Cell lines<br>• Genes<br>• Small molecules |
| L1000 mRNA profiling assay | Broad T LINCS | 2 | Gene expression values per 1,000 genes after de-convolution from Luminex beads | GEX | • Cell lines<br>• Genes<br>• Small molecules |
| L1000 mRNA profiling assay | Broad T LINCS | 3 | Gene expression profiles of both directly measured landmark transcripts plus imputed genes. | Q2NORM | • Cell lines<br>• Genes<br>• Small molecules |

2

| | | | | | |
|---|---|---|---|---|---|
| | | | Normalized using invariant set scaling followed by quantile normalization | | |
| L1000 mRNA profiling assay | Broad T LINCS | 4 | Signatures with differentially expressed genes computed by robust z-scores for each profile relative to population control | GCTX | • Cell lines<br>• Genes<br>• Small molecules |
| RNA-Seq | LINCS consolidated | 1 | Raw sequence (fastq) and aligned sequences (bam) | FASTQ<br>BAM | • Antibodies<br>• Primary cells<br>• Small molecules |
| RNA-Seq | LINCS consolidated | 3 | All feature expression summaries (raw counts, and any other version of counts data) | | • Antibodies<br>• Primary cells<br>• Small molecules |
| RNA-Seq | LINCS consolidated | 4 | Differential feature expression profiles (log fold change, p-values,...) | | • Antibodies<br>• Primary cells<br>• Small molecules |
| | | | | | |
| | | | | | |

References and relevant links

LINCS Metadata Specifications
LINCS Project Website
BD2K LINCS Website

DWG Data Level Site (includes various document and comments)
TCGA Data Levels and Data Types
NCI Data Level Classification
L1000 Data Levels
EPA ToxCast workflow
ToxCast Analysis Presentation (including data their data levels)
AGM Book Chapter Data Standardization for Results Management

4

# .3 Ontology Validation and Evolution Related documents - Reports on Protocols for Updating BAO and DTO

# Manual Steps for Creating a new BAO Version

## Summary of Key Steps (As of February 2017)

1. **Original BAE Absence Report is generated at:**
   **http://www.bioassayexpress.com/BioAssayExpress/diagnostics/absence.jsp**

2. **Absence reported is exported to Excel, and then reviewed by a content (domain) expert to QC, filter for unique new terms, which are then exported to a 'new term template' (BAO_newterm_template.xlsx with link : https://drive.google.com/open?id=0B2oTJxSU7CWrTllYdXZsbndoX0U ) to further clarify by adding required fields--definitions, parent BAO class, relevant references/ hyperlinks. This List of Requested Terms is then shared with University of Miami BAO.**

3. **A survey of content experts could be used to decide on the final labels for terms.**
   **Attention: Currently this step is not performed!**
   **The links to surveys are in a document and they live in 'SurveysAndResults' folder in Google Drive (https://drive.google.com/open?id=0B2oTJxSU7CWrOFlBQ2MwSTdhR1k)**

4. **The List of Requested Terms from Step 2 above is divided and transformed into appropriate separate .csv files by a University of Miami BAO domain expert along with the ontology engineer, inputting the new BAO ID to be assigned and the appropriate BAO parent class ID, using the template (bao_vocab_template.csv) which lives under the GitHub location: (https://github.com/BioAssayOntology/BAO/tree/master/BAOdev/InputFiles).**
   a. **Attention must be paid to terms that already exist in external ontologies (but need to be added to BAO) and terms for which BAO needs to coordinate with external ontologies (e.g., DO, CLO) to request external IDs. All .csv files are placed in the same 'InputFiles' folder on GitHub above (See Part B.2 for input file creations for existing external ontology terms).**

5. **Output files are created in .owl format by the ontology engineer at UM and they live in 'OutputFiles' (https://github.com/BioAssayOntology/BAO/tree/master/BAOdev/OutputFiles ) and external ontology files are added to ../**BAOdev/OutputFilesForExternalOntologyImport **( see Part B.2)**

6. **A checklist of steps for the BAO update process is prepared by the domain expert and the ontology engineer working together. An example called "Example_BAO_building_QC_sheet" is under this link (just to clarify, this is an example document, it should be edited for the current updates as needed):**
   **https://docs.google.com/spreadsheets/d/1Ty1OY48ask1XkKLh1a7WIGPHHMXTvqDY9eH2Ympkf9c/edit?usp=sharing**

7. **Output files are merged with the appropriate vocabulary files by the ontology engineer in UM (per mapping here: https://docs.google.com/spreadsheets/d/1tsxq-j5vLvqTb8FbCW6En8Ei9TDyPYB3IW24HYW4rB4/edit?usp=sharing ) are uploaded to the BAO GitHub (https://github.com/BioAssayOntology) and merged into BAO complete for initial check-up and QC run by the Java Programs and Pipeline Pilot Scripts. Manual check is also performed using Protege.**

8. **After final corrections (changes may be needed, iterating back to step 4 and performing steps 4,5,6 again or manually editing the .owl files), final bao_complete.owl is created.**

9. **After the finalized BAO_complete is created, all files are updated on GitHub and the BioAssay homepage (bioassayontology.org) for BioPortal to collect the new version by the ontology engineer at UM.**

## Background/ Detailed Steps

This document below describes the detailed steps taken to update BAO 2.2.2 with new terms arising from the BAE (BioAssay Express) project--into BAO 2.3.1. It is intended to capture all major steps required to update BAO with new terms and/or other revisions from any source.

We should note that much of the work to compile and merge these new terms spanned from August 2016 to Dec 2016, with the initial resulting merged updated BAO--called BAO2.3 at the time--released onto BioPortal 12/16/16 (though listed as '2.0' on BioPortal). Although all the new terms worked for BAE, we found multiple errors in this build, which ultimately appeared to derive from previous issues in the reference BAO files into which the new terms were merged. Thus in January 2017, a major effort was undertaken by all at UM to identify and 'clean up' / restore the previous BAO version that had been on BioPortal since 11/18/14 (BAO 2.0--see https://sites.google.com/site/baocollaborativedevelopment/home/operational-process/timeline This restored version was exhaustively and comprehensively QC'd, called BAO 2.2.2, and uploaded to BioPortal 1/27/17.

The update reviewed here was based on the BAE Absence Reports created during the curation of 3500 PubChem assays using the BioAssay Express (http://www.bioassayexpress.com/ ). In the course of this curation project, curators would make suggestions for terms they thought were absent from, and merited adding to, BAO. The complete set of these is found in the 'Absence Report'. This Absence Report requires significant filtering and review by a content expert (in both BAO and biology/ HTS assays): some 'suggestions' may not be necessary (either already exist in BAO, are synonymous/ redundant, or may be too detailed and can be adequately covered by a different term.) The net result of 4 'sets' of Absence Reports (snapshots) was a total of ~200 new terms (which includes core and external ontologies).

To perform the updates on BAO, one should ensure that the baseline, i.e. the version of BAO that is currently in use, is usable for updates (which means the current version is free of bugs and inconsistencies). **Currently the only repository and version control is done via GitHub (https://github.com/bioassayontology ).**

The process described below will be slightly modified with the introduction of BAO database which keeps the BAO vocabulary. With the introduction of BAO database, a check for overlapping IDs should no longer be required. However, there should be a check for using the correct ID ranges (for the ID ranges please refer to this document: https://drive.google.com/open?id=1BjUcJTqbvoVpCutI_wRxK56WuXVyDFVMH8n8Kqpro5A).

**Overview of 3-step process (summer 2016)**

A. Extract Terms from BAE; Triage/Analyze for Content; Prepare 'Proposed Terms' Template

B. New terms from single template spreadsheet → Divide into multiple Input (csv) files, based on class/ vocabulary modules→ Generate individual output (owl) files
C. Merge 'new term owl files' with existing BAO vocab files to generate complete owl file and release (to BioPortal via bioassayontology.org)

Above steps A-C were documented (by each responsible party) as separate Word docs, which are compiled below.

**Part A: Extract Terms from BAE; Triage/Analyze for Content; Prepare 'Proposed Terms' Template (by Janice Kranz, CDD)**

1. Go to BAE Absence Report http://www.bioassayexpress.com/BioAssayExpress/diagnostics/absence.jsp
2. Export to xls (via 'Copy to Clipboard' or Select/ Copy All)
3. In Excel: sort by date
    a. 'discard' all covered in previous updates
4. Save remaining= current (unprocessed) set
5. Sort by
    a. Absence Type (i.e., 'needs checking' vs. 'requires term')
    b. Assignment (CAT field/ class)
    c. Description (to group multiple AIDs w same proposed term)
    d. Date
    e. PubChem AID
6. Select 'Absence Type=Needs Checking'
    a. Create new set
    b. Needs QC (Jan or other curator)
7. Select 'Assignment = Target'
    a. Create new set
    b. Ignore (or could use 'fauxtology' to assign)
    c. Reasoning: Target updates in DTO are being done by UM group more efficiently by protein class (eg., enzymes, phosphatases…) Targets await creation of RDF triples.
8. Remaining set= candidates for new terms. **"x# terms no targets"** Now do manual 'triage'
    a. In a new column, note 'x' to select a 'representative' row (i.e., PubChem AID) to generate a set of unique terms
        i. Note that often there are instances of copy/ paste or small typos/ errors, or just different curators noting the same concept/ term that preclude machine auto-detection…I've found 6 'flavors' of the same term
    b. Filter based on this 'x' column (select all x's; ignore blanks)
9. Remaining set= unique list of candidate terms (1 row per term)
10. Manual triage
    a. In a new column, categorize with 1 of 4 flags:
        i. a = already updated (i.e, term included in previous update list)
        ii. b = needs checking (by Jan/ curator; e.g., likely to be covered by existing BAO term)→QC
        iii. c = likely needed as a new term in BAO
        iv. d = needs discussion with domain experts and/or BAO
11. For subset 'c': **Prepare Term List using BAO Update Template**
    **a. This is the labor-intensive step**

**b. Template is here (excel file named 'BAO_new_term_template.xlsx')**

**Screenshot of example:**

***(green=required by BAO for input; light green= optional; blue=for our reference (from Absence Report))***

| proposed term's name | PubChem AID | Description from BAE curator (DO NOT UPLOAD) | existing BAO class for proposed (super) class | BAO parent ID | proposed super class for proposed term (optional) | definition | hyperlinks | Notes (optional)--those are just for Jon/ Hande; do not upload; can provide to outside ontologies, though | date &organizati on |
|---|---|---|---|---|---|---|---|---|---|
| 293 GripTite MSR cells | 602478 | 293 MSR Cell Line (invitrogen #R795-07) | immortal cell line cell | | | The GripTite 293 MSR cell line is a proprietary genetically engineered HEK293 cell line that expresses the human macrophage scavenger receptor and strongly adheres to standard tissue culture plates. | https://www.thermofisher.com/order/catalog/product/R79507 | NOT in CLO or other ontology. | 9/30/16 |
| SU-DHL-10 cell | 493058 | DHL-10 Cells | immortal cell line cell | | | SU-DHL-10 (aka DHL-10) is an immortal human lymphocyte cell line that is a disease model for some non-Hodgkin lymphoma | http://purl.obolibrary.org/obo/CLO_0037059 | in CLO; not in BAO | 9/30/16 |
| Macaca fascicularis | 1190 | primate species needed: rhesus and cynomolgus , bush baby, tree shrew | mammalian | http://www.bioassayontology.org/bao#BAO_00362 | | cynomolgus monkey; long-tailed macaque | http://purl.bioontology.org/ontology/NCBITAXON/9541 | | 9/30/16 |
| Marburg hemorrhagic fever | 720532 | Under viral infectious disease: "marburg hemorrhagic fever". | disease | | | | DOID:4327 | in DOID but needs to be added to DTO | 9/30/16 |
| thermal shift assessment method | 651658 | Under binding assay "ligand-induced thermodynamic stabilization of protein"? | binding assessment method | http://www.bioassayontology.org/bao#BAO_00123 | | Changes in protein thermal stability can be induced by ligand binding, and this thermodynamic stabilization can be measured by a variety of methods, often reported as a delta Tm (or change in the melting temperature of the protein). | | | 9/30/16 |
| Ub-CHOP2-Reporter-Kit (LifeSensors Inc.) | 652174 | Ub-CHOP2-Reporter-Kit (LifeSensor, Inc.) | assay kit | | | An assay for deubiquitination activity, employing a reporter enzyme fused to ubiquitin (making the reporter catalytically inactive). Following cleavage of the Ub-reporter system by the isopeptidase, the free reporter can act upon its fluorescent substrate. | | | 9/30/16 |

If cell line

i. Check in CLO http://purl.bioontology.org/ontology/CLO
ii. If in CLO (and not in BAO)
    1. Note CLO IRI
    2. Term will be added to BAO with CLO IRI
iii. If not in CLO
    1. Do 'detective work'
        a. Find PubMed and /or other hyperlinks for reference
            i. can check BTO or Cellosaurus
        b. Write definition
    2. Term will be added to BAO
    3. BAO will share with CLO for CLO to incorporate into CLO
        a. CLO needs as much info as possible (relevant PMIC or PMCID, PubChem AID(s), Cellosaurus or other URLs)
    4. If/when CLO assigns an IRI, they will notify BAO; BAO will update; should not affect BAO-assigned IRI

c. If disease

i. Check in Disease Ontology http://disease-ontology.org/
ii. If in DO
    1. Note DOID
    2. Term will be added to BAO (or, actually, DTO??) with DOID
iii. If not in DO (THIS IS RARE!! Make Sure to check for synonyms)
    1. Do 'detective work'
        a. Find PubMed and /or other hyperlinks for reference (OMIM, Wikipedia…)
        b. Write definition
    2. Term will be added to BAO (or, actually, DTO??) with BAO ID
    3. BAO will share with CLO for CLO to incorporate into CLO

4. If/when CLO assigns an IRI, they will notify BAO; BAO will update; should not affect BAO-assigned IRI

d. If organism
   i. Check in NCBI Taxon
   ii. Should be foundin NCBI Taxon http://www.ontobee.org/ontology/NCBITaxon
      1. Note NCBI Taxon ID
      2. **For parent (superclass): use abbreviated superclass structure from BAO (to spare having dozens of layers deep)
         a. Check in BAO in BioPortal: (expand organism tree)
         b. Write BAO ID and/or name of BAO organism superclass in template
      3. Term will be added to BAO with NCBI Taxon ID AND with BAO parentID
   iii. If not in NCBI Taxon: highly highly unlikely!

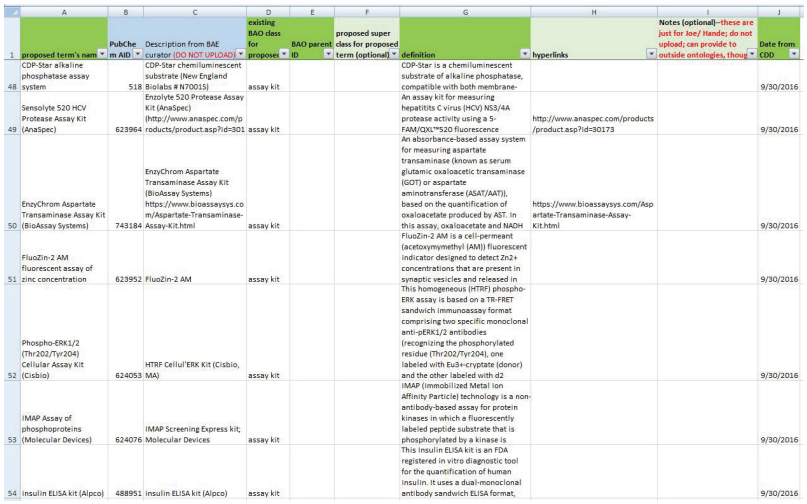
e. All other fields
   i. Evaluate provided 'description' (refer to BAE record and PubChem for context if needed)
   ii. If new term fits into existing BAO superclass
      1. Specify proposed new term
      2. Note name of existing BAO superclass (parent) to which it should be placed under
      3. Write a definition for the new term (use Google liberally!)
      4. Provide hyperlink(s) for reference if useful
   iii. If new term requires a new superclass
      1. Create a new row in the template for the proposed new superclass
         a. Enter the new superclass term
         b. Enter the existing BAO superclass (parent) to which it should be placed under
         c. Write a definition for the new term (use Google liberally!)
         d. Provide hyperlink(s) for reference if useful
      2. Immediately below this new proposed superclass, continue with the new term (as in 11.d.ii), noting the name of the new superclass

12. Send file to Joe/ Hande for next step(s)

Example for 'Set3' of Terms:
1368 terms from Absence Report
660 terms unprocessed (not included in sets 1-2)
586 terms excluding 'target' type (33) and 'needs checking' (41)
145 unique terms
48 terms (a)—covered in sets 1-2
27 terms (b)—needing QC/ likely exist in BAO
68 terms (c)—candidates for BAO
2 terms (d)—need further discussion

**Part B: BAO development documentation: New terms → Input files → Output files pipeline (by Joseph Ostrow)**

**UniqueBAOTerms_090816BAE Report_093016.xlsx** (Example new term spreadsheet)



| | A | B | C | D | E | F | G | H | I | J |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | proposed term's nam | PubChem AID | Description from BAE curator (DO NOT UPLOAD) | existing BAO class for proposed | BAO parent ID | proposed super class for proposed term (optional) | definition | hyperlinks | Notes (optional)--these are just for Joe/ Hande; do not upload; can provide to outside ontologies, thoug | Date from CDD |
| 48 | CDP-Star alkaline phosphatase assay system | 518 | CDP-Star chemiluminescent substrate (New England Biolabs #N70015) | assay kit | | | CDP-Star is a chemiluminescent substrate of alkaline phosphatase, compatible with both membrane- | | | 9/30/2016 |
| 49 | Sensolyte 520 HCV Protease Assay Kit (AnaSpec) | 623964 | Enzolyte 520 Protease Assay Kit (AnaSpec) (http://www.anaspec.com/products/product.asp?id=301 | assay kit | | | An assay kit for measuring hepatitits C virus (HCV) NS3/4A protease activity using a 5-FAM/QXL™520 fluorescence | http://www.anaspec.com/products/product.asp?id=30173 | | 9/30/2016 |
| 50 | EnzyChrom Aspartate Transaminase Assay Kit (BioAssay Systems) | 743184 | EnzyChrom Aspartate Transaminase Assay Kit (BioAssay Systems) https://www.bioassaysys.com/Aspartate-Transaminase-Assay-Kit.html | assay kit | | | An absorbance-based assay system for measuring aspartate transaminase (known as serum glutamic oxaloacetic transaminase (GOT) or aspartate aminotransferase (ASAT/AAT)), based on the quantification of oxaloacetate produced by AST. In this assay, oxaloacetate and NADH | https://www.bioassaysys.com/Aspartate-Transaminase-Assay-Kit.html | | 9/30/2016 |
| 51 | FluoZin-2 AM fluorescent assay of zinc concentration | 623952 | FluoZin-2 AM | assay kit | | | FluoZin-2 AM is a cell-permeant (acetoxymymethyl (AM)) fluorescent indicator designed to detect Zn2+ concentrations that are present in synaptic vesicles and released in | | | 9/30/2016 |
| 52 | Phospho-ERK1/2 (Thr202/Tyr204) Cellular Assay Kit (Cisbio) | 624053 | HTRF Cellul'ERK Kit (Cisbio, MA) | assay kit | | | This homogeneous (HTRF) phospho-ERK assay is based on a TR-FRET sandwich immunoassay format comprising two specific monoclonal anti-pERK1/2 antibodies (recognizing the phosphorylated residue (Thr202/Tyr204), one labeled with Eu3+-cryptate (donor) and the other labeled with d2 | | | 9/30/2016 |
| 53 | IMAP Assay of phosphoproteins (Molecular Devices) | 624076 | IMAP Screening Express kit; Molecular Devices | assay kit | | | IMAP (Immobilized Metal Ion Affinity Particle) technology is a non-antibody-based assay for protein kinases in which a fluorescently labeled peptide substrate that is phosphorylated by a kinase is | | | 9/30/2016 |
| 54 | insulin ELISA kit (Alpco) | 488951 | insulin ELISA kit (Alpco) | assay kit | | | This insulin ELISA kit is an FDA registered in vitro diagnostic tool for the quantification of human insulin. It uses a dual-monoclonal antibody sandwich ELISA format, | | | 9/30/2016 |

1. Begninng with above spreadsheet of new terms, organize terms into separate files based on existing vocabulary "input" files: https://github.com/BioAssayOntology/BAO/tree/master/BAOdev/InputFilesForExternalOntologyImport/bao_vocabulary_dev
    a. If term does not classify as a child of one the existing vocabulary files, refer to "Log for last IDs used in vocab files" spreadsheet on Google Drive and confer with group
    b. For example, above terms would be appended to existing bao_vocabulary_assaykit_dev.csv

**Creating an input file from spreadsheet of new terms (refer to Sample input .csv below):**

2. Add 'Term ID' column (column A)
    a. Use vocabulary ID ranges from 'Log for Last IDs used in vocab files' to assign IDs to terms in form: http://www.bioassayontology.org/bao#BAO_xxxxxxx, beginning with the ID after the last one used

3. Add BAO parent ID based on existing parent class in BAO (use ID already assigned in ontology)
    a. If parent class is also a new term, use its newly generated ID
4. Add 'Class Type' column (Column I)
    a. Assign each term the class type 'subclass' or 'equivalent' based on their definitions in 'Template Strings' document here: https://github.com/ontodev/robot/blob/master/docs/template.md#template-strings\
5. If "Date from CDD" is blank, add in date of .owl file creation (this column is essentially a way to track the most recent additions)
6. Add header row (highlighted in below table)
7. **These headers identify which columns will be interpreted by the Robot command line tool and included in the final .owl file. There are columns in the above spreadsheet (e.g. "Description" or "Notes") with comments not to include in the final .owl file. I have deleted these in the .csv below for clarity's sake, but one could keep them and just not add a header to that column, and they still would not be included in the final .owl output file.
8. Scan text for special characters or symbols (e.g. ®) and delete them, as these often are not interpreted correctly when file is exported
9. Export as .csv
10. Upload the .csv files to GitHub to this folder: https://github.com/BioAssayOntology/BAO/tree/master/BAOdev/InputFiles

**Sample input .csv**

| | A | B | C | D | E | F | G | H | I | J |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Term ID | Term ID | Term ID | existing BAO class | BAO parent ID | proposed super class | definition | hyperlinks | Class Type | Date from CDD |
| 2 | ID | A rdfs:label | A IAO:0000119 | | CI | | A IAO:0000 | A IAO:0000 | CLASS_TYPI | A IAO:0000116 |
| 3 | http://www.bioassayontology.org/bao#BAO_0140009 | CytoTox-Glo (Promega) | 435031 | assay kit | http://www.bioass | Glo'-type assay by Pro | CytoTox-Gl | https://www | subclass | 6/22/2016 |
| 4 | http://www.bioassayontology.org/bao#BAO_0140010 | CAT ELISA (Roche/ Sigma-Al | 624301 | assay kit | http://www.bioassayontology.org/bao#B | The CAT ELI | http://www | subclass | | 6/1/2016 |
| 5 | http://www.bioassayontology.org/bao#BAO_0140011 | ACTOne cAMP biosensor a | 651719 | assay kit | http://www.bioassayontology.org/bao#B | This assay | http://code | subclass | | 6/7/2016 |
| 6 | http://www.bioassayontology.org/bao#BAO_0140012 | CellTiter-Blue Cell Viability | 1063 | assay kit | http://www.bioassayontology.org/bao#B | The CellTite | https://www | subclass | | 5/27/2016 |
| 7 | http://www.bioassayontology.org/bao#BAO_0140013 | Gal-Screen Beta-Galactosi | 2788 | assay kit | http://www.bioassayontology.org/bao#B | The Gal-Scr | https://www | subclass | | 6/22/2016 |
| 8 | http://www.bioassayontology.org/bao#BAO_0140014 | QuantiTect SYBR Green PCF | 504907 | assay kit | http://www.bioass | qPCR system | Real-time P | https://www | subclass | 6/26/2016 |
| 9 | http://www.bioassayontology.org/bao#BAO_0140015 | TaqMan Real-Time PCR ass | 720492 | assay kit | http://www.bioass | qPCR system | Real-time P | https://www | subclass | 6/13/2016 |
| 10 | http://www.bioassayontology.org/bao#BAO_0140016 | Ultra-Glo Luciferin Detecti | 488901 | assay kit | http://www.bioass | Glo'-type assay for m | Ultra-Glo (Promega) is | | subclass | 6/26/2016 |
| 11 | http://www.bioassayontology.org/bao#BAO_0140017 | Luciferase-based ('Glo') As | 624307 | assay kit | http://www.bioassayontology.org/bao#B | An assay system (Prom | | | subclass | 6/1/2016 |
| 12 | http://www.bioassayontology.org/bao#BAO_0000690 | ADP Glo Kinase Assay | | | http://www.bioass | Luciferase-based ('Glo') Assay System (Promeg | | | subclass | 6/1/2016 |
| 13 | http://www.bioassayontology.org/bao#BAO_0000688 | BacTiter-Glo Microbial Cell Viability Assay | | | http://www.bioass | Luciferase-based ('Glo') Assay System (Promeg | | | subclass | 6/1/2016 |
| 14 | http://www.bioassayontology.org/bao#BAO_0000687 | Beta-Glo Assay System | | | http://www.bioass | Luciferase-based ('Glo') Assay System (Promeg | | | subclass | 6/1/2016 |

**Using Robot tool to create .owl output file from input file:**

1. Follow the instructions here to download the Robot command line tool: https://github.com/ontodev/robot. This will convert the .csv input file into the vocabulary .owl file ("output file"), which ultimately will be merged with the complete BAO .owl file.
2. Navigate to the directory of your .csv input file (i.e. ../BAO/BAOdev/InputFiles/bao_vocabulary_dev)
3. Run the following command (inserting correct file and vocabulary names) to create the .owl file:

```
robot template --template bao_vocabulary_assaykit_dev.csv
--ontology-iri
```

```
"http://www.bioassayontology.org/bao/bao_vocabulary_assaykit_dev.owl"
--output ../OutputFiles/bao_vocabulary_assaykit_dev.owl
```

a. `--template` specifies the input file template you are using to create the .owl file
b. `--ontology-iri` specifies the unique ontology IRI (**IRI standard for vocabulary** http://www.bioassayontology.org/bao/bao_vocabulary_ fileName_dev.owl )
c. `--output` specifies the name and location of the .owl file you want to create

4. Confirm .owl file has been generated.
5. Open .owl file in Protege and compare with input .csv to confirm all information is correct
6. Commit changes to GitHub to this location: https://github.com/BioAssayOntology/BAO/tree/master/BAOdev/OutputFiles

**Developer Note:**

1. **The robot tool is used for BAO_core vocabularies only. BAO_external vocabularies are created using OntoFox**
2. **Previously the OntoRat tool was used, but we found it was not very stable (was unavailable for weeks), so the robot tool was used locally.**

**Example files are here (please note that these are not up-to-date files and are linked for example purposes only):**

1. **Joe transforms the Absence Report into OntoRat input files which live under 'OntoRatInputFiles' folder in the CDD-UM Google Drive (https://drive.google.com/open?id=0B2oTJxSU7CWrTFdLSlFHYnNtMkU)**

2. **OntoRat output files are created by Joe and they live in 'OntoRatOutputFiles' folder (https://drive.google.com/open?id=0B2oTJxSU7CWrZ1FYNlFfQ2NSTDQ)**

**Part B.2 Creation of External Ontology Extractions using OntoFox (by Hande Kucuk-McGinty)**

1. Use the existing input files for OntoFox (currently in GitHub under https://github.com/BioAssayOntology/BAO/tree/master/BAOdev/InputFilesForExternalOntologyImport )
2. add the new terms from external ontologies such as DOID or CLO to the list
3. save and upload to GitHub to this location: https://github.com/BioAssayOntology/BAO/tree/master/BAOdev/InputFilesForExternalOntologyImport
4. upload the input files to Ontofox site (ontofox.hegroup.org) to create the .owl files
5. save and upload .owl files to here: https://github.com/BioAssayOntology/BAO/tree/master/BAOdev/OutputFilesForExternalOntologyImport

**Part C. Merge 'new term owl files' with existing BAO vocab files to generate complete owl file and release (by Hande Kucuk-McGinty)**

**Developer Note:**

**Before the merging,we previously suggested to perform surveys for finalizing labels for the new terms. But currently this step is not performed!**

**The surveys have not been out yet, the link to surveys are in a document and they live in 'SurveysAndResults' folder in Google Drive (https://drive.google.com/open?id=0B2oTJxSU7CWrOFlBQ2MwSTdhR1k)**

**Merging Details:**
**Output files are merged with vocabulary files that live at the BAO GitHub (https://github.com/BioAssayOntology) (changes maybe done going back to Part B and performing re-creation of input and output files again or manually)**

**For the merging, see more detailed notes on BAO2.3.1 Update:**
https://sites.google.com/site/baocollaborativedevelopment/home/operational-process/bao-updates
(assumes starting with new terms from BAE in template)

1. Assuming the input (.csv files) and output files (bao_vocabulary_x_dev.owl) are correct, merge "x_x_dev.owl" vocab files with appropriate vocab file from most recent BAO build.

   See Table for mappings:

| Development File (new terms to be added) | BAO Vocabulary (form most recent BAO release on git) |
|---|---|
| bao_vocabulary_cellline_dev.owl | bao_vocabulary_biology.owl/ BAO_CLO_import.owl |
| bao_vocabulary_method_dev.owl | bao_vocabulary_method.owl |
| bao_vocabulary_format_dev.owl | bao_vocabulary_biology.owl |
| bao_vocabulary_assaykit_dev.owl | bao_vocabulary_assaykit.owl |
| bao_vocabulary_assay_dev.owl | bao_vocabulary_assay.owl |
| bao_vocabulary_detection_dev.owl | bao_vocabulary_detection.owl |
| bao_module_organism_dev.owl | bao_vocabulary_biology.owl/ BAO_NCBITaxon_import.owl |
| bao_vocabulary_result_dev.owl | bao_vocabulary_result.owl |
| bao_vocabulary_instrument_dev.owl | bao_vocabulary_instrument.owl |
| bao_vocabulary_screenedentity_dev.owl | bao_vocabulary_screenedentity.owl |
| bao_vocabulary_unit_dev | BAO_UO_import.owl |
| bao_module_disease_dev.owl | bao_vocabulary_biology.owl/ |

| | BAO_DOID_import.owl |
|---|---|

For this exercise, BAO base version is BAO2.2.2

- 1.1 Open _dev file and BAO2.2.2 vocab file in Protege
- 1.2 In Protege while viewing BAO2.2.2 vocab file, click on 'Direct Imports' (+)--> hit '+' and import the _dev owl file
- 1.3 View to check (check number of classes)
- 1.4 Under 'Refactor'--> choose 'Merge' (select the 2; merge into existing ontology)
- Check; now delete the import of the _dev file
- Save as the bao_vocabulary_x.owl

For the 3 external vocabs

1. Cell Lines
   - Merge bao_vocabulary_cellline_dev.owl into BAO2.2.2
     1. Note: this vocab contains cell lines assigned BAO IDs (not (yet) in CLO)
   - Merge (from BAO/BAOdev/OutputFiles/) BAO2.3_CLO_import.owl into above merged file
     1. Note: this CLO import file contains cell lines found in BAE that were in CLO but need to be added to BAO
2. Disease
   - Merge bao_vocabulary_DISEASE_dev.owl into BAO2.2.2
     1. Note: this vocab contains cell lines assigned BAO IDs (not (yet) in DOID)
   - Merge (from BAO/BAOdev/OutputFiles/) BAO2.3_DOID_import.owl into above merged file
     1. Note: this DOID import file contains cell lines found in BAE that were in DOID but need to be added to BAO
3. Organism (NCBI Taxon)
4. Open bao_external.owl (from BAO2.2.2)
   - Merge with above 3 files

To create complete_merged

5. Open bao_core
   - Control if the bao_core.owl contains(imports) all the vocabulary files and make sure all the files are imported correctly.
6. Open bao_external.owl and bao_complete.owl
7. bao_complete_merged.owl shoudl contain the bao_core, bao_external and bao_metadata.
8. Edit all files to contain new version, release date, names, best way to do this using a text editor that can replace in multiple files at once.
9. Edit Release Notes doc and add to developer note folder
10. Commit onto git : https://github.com/BioAssayOntology/BAO
11. Publish as PRE-RELEASE on github: BAO2.3.1
12. After running QC with Java code and Stephan's QC scripts, confirm with Stephan to release and confirm.
    - If not a confirm, one might have to go back to Part B and perform Part B and Part C for another round of QC.

**3.** BAO release (by Caty Chung, taken from GoogleSite page here)

| URL | Description | State |
|---|---|---|

| | | |
|---|---|---|
| http://web.ccs.miami.edu/repos/ | UM svn | 2.0.x |
| https://github.com/BioAssayOntology/BAO/releases | git | 2.3.1 |

1.- Commit a release: https://github.com/BioAssayOntology/BAO/releases
2.- FTP files to **http://www.bioassayontology.org/bao/[file name]**
**Update release notes:**
**https://docs.google.com/document/d/1Vf4BEejEZ7vuEdT1QNAUlRBpRpuPHSDdOde1SwHjSr4/edit**
3.- Check **http://www.bioassayontology.org/bao/**
4.- BioPortal has a routine job to pick up new files, the core file needs to be updated:
http://www.bioassayontology.org/bao/bao_complete.owl
5.- Check BioPortal

| Steps \ initials | | Janice (JK) | John (JPT) | Hande (HKM) | File Location(s) | Description | Results (e.g., # new terms; # BAO IDs, # external | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | ~~1/18/17~~ 1/19/17 | (John needs to v | 02/06/2017 | | | | | | | | | | | | | |
| ID ranges file | | ok | ok | ok | https://docs.google.com/spreadsheets/d/1BjUcJTqbvoVpCutI_wRxK56WuXVyDFVMH8n8Kqpro5A/edit#gid=1113641407 | | | | | | | | | | | | | |
| BAO 2.0 files--LOTS of cleanup needed! 1/26/17 and 1/23/17 | | ok | | ok | | | | | | | | | | | | | |
| **QC of BAO2.0 (for baseline--will be called 'BAO2.2.2')** | 1/23/17 | | | | | | | | | | | | | | | | |
| see Google Site BAO2.2.2 steps for details | | ok | ok | | | Notes from GoogleSite: https://sites.google.com/site/baocollaborativedevelopment/home/operational-process/bao-release/bao2-2-2-steps | | | | | | | | | | | | |
| after Stephan QC, need to merge; then ready for upload | | | | | | | | | | | | | | | | | |
| review in Protege | | ok | | ok | | | | | | | | | | | | | |
| Build complete_merged file | | | | ok | | | | | | | | | | | | | |
| ~~use script by Hande~~ | | | | | | Did not work/ did not use; no need since Protege works fine and allows checking intermediate steps | | | | | | | | | | | | |
| commit (release) on git | | | | | 2.2.2 on github | | | | | | | | | | | | |
| restore to BAO project site | | | | | | John & Caty did 1/27/17 | | | | | | | | | | | |
| check on BioPortal for new submission | | | | | | 1/29/17: ok | | | | | | | | | | | |
| **ADDING NEW (BAE) TERMS FROM 'BAO2.3' (DEC 2016)** | | | | | | | | | | | | | | | | | |
| **Development input files (csv)** | | | | | https://github.com/BioAssayOntology/BAO/tree/master/BAOdev/InputFiles | | | | | | | | | | | | |
| On github | bao_vocabulary_assay_dev | ok | | ok | | | 20 | | | | | | | | | | |
| ~~from Jan 1/18/17~~ now on github1/19/17 | bao_vocabulary_assaykit_dev | ok | | ok | | | 52 | (includes 18 existing but new parent assignment); deleted BAO_0140021 | | | | | | | | | |
| now on github1/19/17 | bao_vocabulary_detection_dev | ok | | ok | | | 2 | | | | | | | | | | |
| now on github1/19/17 | bao_vocabulary_format_dev | ok | | ok | | | 3 | | | | | | | | | | |
| now on github1/19/17 | bao_vocabulary_instrument_dev | ok | | ok | | | 40 | | | | | | | | | | |
| now on github1/19/17 | bao_vocabulary_method_dev | ok | | ok | | | 14 | | | | | | | | | | |
| now on github1/19/17 | bao_vocabulary_result_dev | ok | | ok | | | 9 | | | | | | | | | | |
| now on github1/19/17 | bao_vocabulary_screenedentity_dev | ok | | ok | | | 2 | | | | | | | | | | |
| ~~from Jan 1/18/17 bao_module_cellline_dev_hkm_011917.csv~~ now ongithub 1/19/17 | bao_module_cellline_dev | ok | | ok | | | 49 | 14 from Aug with BAO and (new) CLO IDs--want to use ONLY CLO ID; 10 CLO IDs but not in BAO; 25 with new BAO Ids--should be sent to CLO for addition | | | | | | | | | |
| from Jan 1/18/17 | bao_module_disease_dev | ok | | ok | | | 2 | correct DOID225 (not DOID4) but needed to delete 1 disease | | | | | | | | | |
| from Jan 1/18/17 | bao_module_organism_dev | ok | | ok | | | 26 | corrected Cavia NCBI ID; all parent IDs correct. | | | | | | | | | |
| ~~from 1/19/17~~ now ongithub 1/19/17 | separated OntoFoxInput file CLO_import_input_file_01.19.2017 | ok | | ok | | txt file | | | | | | | | | | | |
| ~~from 1/19/17~~ now ongithub 1/19/17 | separated OntoFoxInput file BAO_DTO_DOID_import_input_file_01.19.2017 | ok | | ok | | txt file | | | | | | | | | | | |
| ~~from 1/19/17~~ now ongithub 1/19/17 | OntoFoxInput file NCBITaxon_import_input_file_01.19.2017 | ok | | ok | | txt file | | | | | | | | | | | |
| | | | | | | | 219 | | | | | | | | | | |
| **Development output files (owl)** | | | | | https://github.com/BioAssayOntology/BAO/tree/master/BAOdev/OutputFiles | | | | | | | | | | | | |
| ~~from Hande 1/18/17~~ now on github1/19/17 | bao_vocabulary_assay_dev | ok | | | | | 20 | (bioassay 6; footprint 13; screening camp stage 1) | | | | | | | | | |
| ~~from Hande 1/18/17~~ now on github1/19/17 | bao_vocabulary_assaykit_dev | ok | | | | | 52 | | | | | | | | | | |
| ~~from Hande 1/18/17~~ | ~~bao_vocabulary_biology_dev~~ | NOT NEEDED | | | | | | | | | | | | | | | |
| ~~from Hande 1/18/17~~ now on github1/19/17 | bao_vocabulary_detection_dev | ok | | | | | 2 | | | | | | | | | | |
| ~~from Hande 1/18/17~~ now on github1/19/17 | bao_vocabulary_format_dev | ok | | | | | 3 | | | | | | | | | | |
| ~~from Hande 1/18/17~~ now on github1/19/17 | bao_vocabulary_instrument_dev | ok | | | | | 40 | | | | | | | | | | |
| ~~from Hande 1/18/17~~ now on github1/19/17 | bao_vocabulary_method_dev | ok | | | | | 14 | | | | | | | | | | |
| ~~from Hande 1/18/17~~ now on github1/19/17 | bao_vocabulary_result_dev | ok | | | | Actually contained terms for 'result' and for 'unit' vocabs--need to separate. nit terms are subclasses of UO_IDs, so these need to be noted in the complete file (as they are not BAO core classes) | 9 | | | | | | | | | | | |
| ~~from Hande 1/18/17~~ now on github1/19/17 | bao_vocabulary_screenedentity_dev | ok | | | | | 2 | | | | | | | | | | |
| ~~from Hande 1/18/17~~ | bao_vocabulary_cellline_dev.owl | ok | | | in BAOdev/OutputFiles/external_vocabulary_dev | | 25 | | | | | | | | | | |
| now on github1/19/17 | BAO2.3_CLO_import.owl | ok | | | in BAOdev/OutputFilesForImports | | | all CLO terms (new plus form previous BAO versions) | | | | | | | | | |
| ~~from Hande 1/18/17~~ | bao_vocabulary_disease_dev.owl | ok | | | in BAOdev/OutputFiles/external_vocabulary_dev | | 2 | | | | | | | | | | |
| now on github1/19/17 | BAO2.3_DOID_import.owl | ok | | | in BAOdev/OutputFilesForImports | | | all DOID terms (new plus form previous BAO versions) | | | | | | | | | |
| now on github1/19/17 | BAO2.3_NCBITaxon_import.owl | ok | | | in BAOdev/OutputFilesForImports | | | all organism (NCBI Taxon) terms (new plus form previous BAO versions) | | | | | | | | | |
| **Other (TBD)** | | | | | | | | | | | | | | | | | |
| ID updates file | | | | | https://docs.google.com/spreadsheets/d/1BjUcJTqbvoVpCutI_wRxK56WuXVyDFVMH8n8Kqpro5A/edit#gid=1113641407 | | | | | | | | | | | | | |
| General QC of BAO vocabulary files BAO2.x | | | | | | | | | | | | | | | | | |
| specifics to check for this QC? | | | | | | | | | | | | | | | | | |
| Merging files for BAO2.x_complete.owl | | | | | | See details on GoogleSite: | https://sites.google.com/site/baocollaborativedevelopment/home/operational-process/bao-release/bao2-2-2-steps | | | | | | | | | | |
| | bao_vocabulary_assay_dev | | | | | | | | | | | | | | | | |
| | bao_vocabulary_assaykit_dev | | | | | | | | | | | | | | | | |
| | bao_vocabulary_detection_dev | | | | | | | | | | | | | | | | |
| | bao_vocabulary_format_dev | | | | | | | | | | | | | | | | |
| | bao_vocabulary_instrument_dev | | | | | | | | | | | | | | | | |
| | bao_vocabulary_method_dev | | | | | | | | | | | | | | | | |
| | bao_vocabulary_result_dev | | | | | | | | | | | | | | | | |
| | bao_vocabulary_screenedentity_dev | | | | | | | | | | | | | | | | |
| | bao_module_cellline_dev | | | | | | | | | | | | | | | | |
| | bao_module_disease_dev | | | | | | | | | | | | | | | | |
| | bao_module_organism_dev | | | | | | | | | | | | | | | | |
| **QC of merged files for BAO2.x** | | ok | | | | | | | | | | | | | | | |
| specific metrics: # unique labels, # unique IDs; any duplicate IDs/ labels, etc | | | | | | | | | | | | | | | | | |
| Edit Release Notes (I assume there is only ONE location for these) | | | | | | | | | | | | | | | | | |
| | edit version # (2.x') | ok | ok | ok | | 2.3.1 | | | | | | | | | | | |

| Steps \ initials | | Janice (JK) | John (JPT) | Hande (HKM) | File Location(s) | Description | Results (e.g., # new terms; # BAO IDs, # external | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | edit Build # | ok | ok | ok | | NA | | | | | | | | | | |
| | edit release date | ok | ok | ok | | 1/30/17 | | | | | | | | | | |
| | edit Names | ok | ok | ok | | same as 2.2.2 | | | | | | | | | | |
| | edit README/ Release Notes | ok | ok | ok | | edited | | | | | | | | | | |
| Commit new version on github | first as 'pre-release' | ok | ok | ok | | | | | | | | | | | | |
| Change to full release | | | | | | | | | | | | | | | | |
| ? Replace files in 'master BAO' folder on github? | | | | | | | | | | | | | | | | |
| ftp to bioassayontology.org | | | | | | | | | | | | | | | | |
| Check bioassayontology.org that updated files appear | | | | | | | | | | | | | | | | |
| Check if new version appears on Bioportal | | | | | | | | | | | | | | | | |

# Bibliography


[1] C. E. Cook, M. T. Bergman, R. D. Finn, G. Cochrane, E. Birney, and R. Apweiler, "The european bioinformatics institute in 2016: data growth and integration," *Nucleic acids research*, vol. 44, no. D1, pp. D20–D26, 2015.

[2] B. Louie, P. Mork, F. Martin-Sanchez, A. Halevy, and P. Tarczy-Hornoch, "Data integration and genomic medicine," *Journal of biomedical informatics*, vol. 40, no. 1, pp. 5–16, 2007.

[3] G. J. O. R. I. Grauch, "Rare earth element mines, deposits, and occurrences," 2002.

[4] S. Abeyruwan, U. D. Vempati, H. Küçük-McGinty, U. Visser, A. Koleti, A. Mir, K. Sakurai, C. Chung, J. A. Bittker, P. A. Clemons *et al.*, "Evolving bioassay ontology (bao): modularization, integration and applications," *Journal of biomedical semantics*, vol. 5, no. Suppl 1, p. S5, 2014.

[5] U. Visser, S. Abeyruwan, U. Vempati, R. P. Smith, V. Lemmon, and S. C. Schürer, "Bioassay ontology (bao): a semantic description of bioassays and high-throughput screening results," *BMC bioinformatics*, vol. 12, no. 1, p. 1, 2011.

[6] U. of Michigan, "Michigan Institute for Data Science," last Visted on 12/04/2017. [Online]. Available: http://midas.umich.edu/

[7] S. University, "Stanford Data Science Initiative," last Visted on 12/04/2017. [Online]. Available: https://sdsi.stanford.edu/

[8] U. of Virginia, "Data Science Institute," last Visted on 12/04/2017. [Online]. Available: https://dsi.virginia.edu/

[9] V. Marx, "Biology: The big challenges of big data," 2013.

[10] C. P. Chen and C.-Y. Zhang, "Data-intensive applications, challenges, techniques and technologies: A survey on big data," *Information Sciences*, vol. 275, pp. 314–347, 2014.

[11] A. Katal, M. Wazid, and R. Goudar, "Big data: issues, challenges, tools and good practices," in *Contemporary Computing (IC3), 2013 Sixth International Conference on*. IEEE, 2013, pp. 404–409.

[12] V. C. Storey and I.-Y. Song, "Big data technologies and management: What conceptual modeling can do," *Data & Knowledge Engineering*, vol. 108, pp. 50–67, 2017.

[13] Z. Lv, H. Song, P. Basanta-Val, A. Steed, and M. Jo, "Next-generation big data analytics: State of the art, challenges, and future research topics," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 4, pp. 1891–1899, 2017.

[14] N. F. Noy, N. H. Shah, P. L. Whetzel, B. Dai, M. Dorf, N. Griffith, C. Jonquet, D. L. Rubin, M.-A. Storey, C. G. Chute *et al.*, "Bioportal: ontologies and integrated data resources at the click of a mouse," *Nucleic acids research*, p. gkp440, 2009.

[15] L. Hunter, Z. Lu, J. Firby, W. A. Baumgartner, H. L. Johnson, P. V. Ogren, and K. B. Cohen, "Opendmap: an open source, ontology-driven concept analysis engine, with applications to capturing knowledge regarding protein transport, protein interactions and cell-type-specific gene expression," *BMC bioinformatics*, vol. 9, no. 1, p. 1, 2008.

[16] L. Cao, "Data science: Challenges and directions," *Commun. ACM*, vol. 60, no. 8, pp. 59–68, Jul. 2017. [Online]. Available: http://doi.acm.org/10.1145/3015456

[17] V. Marx, "Biology: The big challenges of big data," 2013.

[18] C. J. Bult, H. J. Drabkin, A. Evsikov, D. Natale, C. Arighi, N. Roberts, A. Ruttenberg, P. D'Eustachio, B. Smith, J. A. Blake *et al.*, "The representation of protein complexes in the protein ontology (pro)," *BMC bioinformatics*, vol. 12, no. 1, p. 1, 2011.

[19] W. A. Kibbe, C. Arze, V. Felix, E. Mitraka, E. Bolton, G. Fu, C. J. Mungall, J. X. Binder, J. Malone, D. Vasant, H. Parkinson, and L. M. Schriml, "Disease ontology 2015 update: an expanded and updated database of human diseases for linking biomedical knowledge through disease data," *Nucleic Acids Research*, vol. 43, no. D1, pp. D1071–D1078, 2015. [Online]. Available: http://nar.oxfordjournals.org/content/43/D1/D1071.abstract

[20] S. Sarntivijai, Z. Xiang, T. F. Meehan, A. D. Diehl, U. Vempati, S. C. Schürer, C. Pang, J. Malone, H. E. Parkinson, B. D. Athey *et al.*, "Cell line ontology: Redesigning the cell line knowledgebase to aid integrative translational informatics." *ICBO*, vol. 833, pp. 25–32, 2011.

[21] G. Schreiber, *Knowledge engineering and management: the CommonKADS methodology*. MIT press, 2000.

[22] R. Stevens, C. A. Goble, and S. Bechhofer, "Ontology-based knowledge representation for bioinformatics," *Briefings in bioinformatics*, vol. 1, no. 4, pp. 398–414, 2000.

[23] R. B. Altman, M. Buda, X. J. Chai, M. W. Carillo, R. O. Chen, and N. F. Abernethy, "Riboweb: An ontology-based system for collaborative molecular biology," *IEEE Intelligent Systems and Their Applications*, vol. 14, no. 5, pp. 68–76, 1999.

[24] I. M. Keseler, J. Collado-Vides, S. Gama-Castro, J. Ingraham, S. Paley, I. T. Paulsen, M. Peralta-Gil, and P. D. Karp, "Ecocyc: a comprehensive database resource for escherichia coli," *Nucleic acids research*, vol. 33, no. suppl_1, pp. D334–D337, 2005.

[25] T. G. O. Consortium, "Gene ontology consortium: going forward," *Nucleic Acids Research*, vol. 43, no. D1, pp. D1049–D1056, 2015. [Online]. Available: http://nar.oxfordjournals.org/content/43/D1/D1049.abstract

[26] P. L. Whetzel, N. F. Noy, N. H. Shah, P. R. Alexander, C. Nyulas, T. Tudorache, and M. A. Musen, "Bioportal: enhanced functionality via new web services from the national center for biomedical ontology to access and use ontologies in software applications," *Nucleic acids research*, vol. 39, no. suppl 2, pp. W541–W545, 2011.

[27] H. Wache, T. Voegele, U. Visser, H. Stuckenschmidt, G. Schuster, H. Neumann, and S. Hübner, "Ontology-based integration of information-a survey of existing approaches," in *IJCAI-01 workshop: ontologies and information sharing*, vol. 2001. Citeseer, 2001, pp. 108–117.

[28] P. M. Gray, A. Preece, N. Fiddian, W. Gray, T. J. Bench-Capon, M. J. Shave, N. Azarmi, I. Wiegand, M. Ashwell, M. Beer *et al.*, "Kraft: Knowledge fusion from distributed databases and knowledge bases," in *Database and Expert Systems Applications, 1997. Proceedings., Eighth International Workshop on*. IEEE, 1997, pp. 682–691.

[29] S. Decker, M. Erdmann, D. Fensel, and R. Studer, "Ontobroker: Ontology based access to distributed and semi-structured information," in *Database Semantics*. Springer, 1999, pp. 351–369.

[30] Y. Arens, C.-N. Hsu, and C. A. Knoblock, "Query processing in the sims information mediator," *Advanced Planning Technology*, vol. 32, pp. 78–93, 1996.

[31] S. Luke, L. Spector, D. Rager, and J. Hendler, "Ontology-based web agents," in *Proceedings of the first international conference on Autonomous agents*. ACM, 1997, pp. 59–66.

[32] Y. Sure, M. Erdmann, J. Angele, S. Staab, R. Studer, and D. Wenke, "Ontoedit: Collaborative ontology development for the semantic web," *ISWC 2002*, p. 221235, 2002.

[33] M. V. Blagosklonny and A. B. Pardee, "Conceptual biology: unearthing the gems," *Nature*, vol. 416, no. 6879, pp. 373–373, 2002.

[34] J. C. Barnes, "Conceptual biology: a semantic issue and more," *Nature*, vol. 417, no. 6889, pp. 587–588, 2002.

[35] A. Bandrowski, R. Brinkman, M. Brochhausen, M. H. Brush, B. Bug, M. C. Chibucos, K. Clancy, M. Courtot, D. Derom, M. Dumontier *et al.*, "The ontology for biomedical investigations," *PloS one*, vol. 11, no. 4, p. e0154556, 2016.

[36] Y. He, Z. Xiang, J. Zheng, Y. Lin, J. A. Overton, and E. Ong, "The extensible ontology development (xod) principles and tool implementation to support ontology interoperability," *Journal of biomedical semantics*, vol. 9, no. 1, p. 3, 2018.

[37] O. Corcho, M. Fernández-López, and A. Gómez-Pérez, "Methodologies, tools and languages for building ontologies. where is their meeting point?" *Data & knowledge engineering*, vol. 46, no. 1, pp. 41–64, 2003.

[38] D. B. Lenat and R. V. Guha, *Building large knowledge-based systems; representation and inference in the Cyc project*. Addison-Wesley Longman Publishing Co., Inc., 1989.

[39] M. Uschold and M. King, *Towards a methodology for building ontologies*. Citeseer, 1995.

[40] A. Gómez-Pérez, M. Fernández, and A. d. Vicente, "Towards a method to conceptualize domain ontologies," 1996.

[41] WWW3, "The owl 2 web ontology language document overview," last visited on 11/01/2013. [Online]. Available: http://www.w3.org/TR/owl2-overview/

[42] N. F. Noy, R. W. Fergerson, and M. A. Musen, "The knowledge model of protege-2000: Combining interoperability and flexibility," in *Knowledge Engineering and Knowledge Management Methods, Models, and Tools*. Springer, 2000, pp. 17–32.

[43] K. Wang, P. Tarczy-Hornoch, R. Shaker, P. Mork, J. F. Brinkley *et al.*, "Biomediator data integration: beyond genomics to neuroscience data." in *AMIA*, 2005.

[44] T. E. Klein, J. T. Chang, M. K. Cho, K. L. Easton, R. Fergerson, M. Hewett, Z. Lin, Y. Liu, S. Liu, D. Oliver *et al.*, "Integrating genotype and phenotype information: an overview of the pharmgkb project," *The pharmacogenomics journal*, vol. 1, no. 3, pp. 167–170, 2001.

[45] P. R. Payne, E. A. Mendonça, S. B. Johnson, and J. B. Starren, "Conceptual knowledge acquisition in biomedicine: A methodological review," *Journal of biomedical informatics*, vol. 40, no. 5, pp. 582–602, 2007.

[46] P. Gottgtroy, R. Modaini, N. Kasabov *et al.*, "Building evolving ontology maps for data mining and knowledge discovery in biomedical informatics," in *Proceedings of the third Brazilian symposium on mathematical and computational biology (BIOMATIII), Rio de Janeiro, Brazil*, vol. 1, 2003, pp. 309–328.

[47] P. Gottgtroy, N. Kasabov, and S. Macdonell, "An ontology engineering approach for knowledge discovery from data in evolving domains," 2004.

[48] "Protégé," last visited on 06/10/2015. [Online]. Available: http://protege.stanford.edu/

[49] I. Yeh, P. D. Karp, N. F. Noy, and R. B. Altman, "Knowledge acquisition, consistency checking and concurrency control for gene ontology (go)," *Bioinformatics*, vol. 19, no. 2, pp. 241–248, 2003.

[50] J. Köhler, S. Philippi, and M. Lange, "Semeda: ontology based semantic integration of biological databases," *Bioinformatics*, vol. 19, no. 18, pp. 2420–2427, 2003.

[51] B. Smith, M. Ashburner, C. Rosse, J. Bard, W. Bug, W. Ceusters, L. J. Goldberg, K. Eilbeck, A. Ireland, C. J. Mungall *et al.*, "The obo foundry: coordinated evolution of ontologies to support biomedical data integration," *Nature biotechnology*, vol. 25, no. 11, pp. 1251–1255, 2007.

[52] F. Belleau, M.-A. Nolin, N. Tourigny, P. Rigault, and J. Morissette, "Bio2rdf: towards a mashup to build bioinformatics knowledge systems," *Journal of biomedical informatics*, vol. 41, no. 5, pp. 706–716, 2008.

[53] K. Wolstencroft, S. Owen, O. Krebs, W. Mueller, Q. Nguyen, J. L. Snoep, and C. Goble, "Semantic data and models sharing in systems biology: The just enough results model and the seek platform," in *International Semantic Web Conference*. Springer, 2013, p. 212227.

[54] H. Group, "OntoFox," last visited on 11/01/2013. [Online]. Available: http://ontofox.hegroup.org/

[55] R. P. Hertzberg and A. J. Pope, "High-throughput screening: new technology for the 21st century," *Current Opinion in Chemical Biology*, vol. 4, no. 4, pp. 445 – 451, 2000.

[56] C. P. Austin, L. S. Brady, T. R. Insel, and F. S. Collins, "Nih molecular libraries initiative," *Science*, vol. 306, no. 5699, pp. 1138–1139, 2004.

[57] E. E. Bolton, Y. Wang, P. A. Thiessen, and S. H. Bryant, "Pubchem: integrated platform of small molecules and biological activities," *Annual reports in computational chemistry*, vol. 4, pp. 217–241, 2008.

[58] R. Frank, "Eu-openscreen–a european infrastructure of open screening platforms for chemical biology," *ACS chemical biology*, vol. 9, no. 4, pp. 853–854, 2014.

[59] A. Gaulton, L. J. Bellis, A. P. Bento, J. Chambers, M. Davies, A. Hersey, Y. Light, S. McGlinchey, D. Michalovich, B. Al-Lazikani *et al.*, "Chembl: a large-scale bioactivity database for drug discovery," *Nucleic acids research*, vol. 40, no. D1, pp. D1100–D1107, 2012.

[60] A. Gaulton, L. J. Bellis, A. P. Bento, J. Chambers, M. Davies, A. Hersey, Y. Light, S. McGlinchey, D. Michalovich, B. Al-Lazikani, and J. P. Overington, "ChEMBL: a large-scale bioactivity database for drug discovery," *Nucleic Acids Research*, vol. 40, no. D1, pp. D1100–D1107, jan 2012.

[61] N. H. Jensen and B. L. Roth, "Massively parallel screening of the receptorome," *Combinatorial Chemistry & High Throughput Screening*, vol. 11, pp. 420–426, 2008.

[62] M. Hohman, K. Gregory, K. Chibale, P. J. Smith, S. Ekins, and B. Bunin, "Novel web-based tools combining chemistry informatics, biology and social networks for drug discovery," *Drug Discovery Today*, vol. 14, no. 5–6, pp. 261 – 270, 2009.

[63] S. Lab, "Library of Integrated Network-based Cellular Signatures (LINCS) Information FramEwork," last visited on 11/01/2013. [Online]. Available: http://life.ccs.miami.edu/life/

[64] NIH, "Illuminating the Druggable Genome | NIH Common Fund," last visited on 06/06/2015. [Online]. Available: https://commonfund.nih.gov/idg/index

[65] R. K. G. James Inglese, Caroline E Shamu, "Reporting data from high-throughput screening of small-molecule libraries," *Nature Chemical Biology*, no. 8, pp. 438–441, 2007.

[66] U. D. Vempati, M. J. Przydzial, C. Chung, S. Abeyruwan, A. Mir, K. Sakurai, U. Visser, V. P. Lemmon, and S. C. Schürer, "Formalization, Annotation and Analysis of Diverse Drug and Probe Screening Assay Datasets Using the BioAssay Ontology (BAO)," *PLoS ONE*, vol. 7, no. 11, November 2012.

[67] H. Küçük-Mcginty, S. Metha, Y. Lin, N. Nabizadeh, V. Stathias, D. Vidovic, A. Koleti, C. Mader, J. Duan, U. Visser, and S. Schurer, "It405: Building concordant ontologies for drug discovery," in *International Conference on Biomedical Ontology and BioCreative (ICBO BioCreative 2016)*, ser. Proceedings of the Joint International Conference on Biological Ontology and BioCreative (2016), ICBO and BioCreative. ICBO and BioCreative, 08/01/2016 2016. [Online]. Available: http://icbo.cgrb.oregonstate.edu/

[68] Y. Lin, S. Mehta, H. K. McGinty, J. P. Turner, D. Vidovic, M. Forlin, A. Koleti, D.-T. Nguyen, L. J. Jensen, R. Guha *et al.*, "Drug target ontology to classify and integrate drug discovery data," *bioRxiv*, p. 117564, 2017.

[69] S. Abeyruwan, A. Seekircher, and U. Visser, "Dynamic Role Assignment using General Value Functions," in *Proceedings of Autonomous Agents and Multi-Agent Systems, Workshop on Adaptive Learning Agents*, 2013.

[70] K. Shvachko, H. Kuang, S. Radia, and R. Chansler, "The Hadoop Distributed File System," in *Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST)*. Washington, DC, USA: IEEE Computer Society, 2010, pp. 1–10.

[71] NIH, "Library of Integrated Network-Based Cellular Signatures (NIH LINCS) program," last visited on 06/06/2015. [Online]. Available: http://www.lincsproject.org/

[72] H. M. School, "KiNativ, In Situ Kinase Profiling," last visited on 06/06/2015. [Online]. Available: http://www.kinativ.com/

[73] ——, "KiNativ, In Situ Kinase Profiling," last visited on 06/06/2015. [Online]. Available: http://www.kinativ.com/

[74] D. Peck, E. D. Crawford, K. N. Ross, K. Stegmaier, T. R. Golub, and J. Lamb, "A method for high-throughput gene expression signature analysis," *Genome biology*, vol. 7, no. 7, p. R61, 2006.

[75] B. Institute, "1000 Genomes," last visited on 06/06/2015. [Online]. Available: https://www.broadinstitute.org/science/projects/1000-genomes

[76] H. M. School, "HMS LINCS Database," last visited on 06/09/2015. [Online]. Available: https://lincs.hms.harvard.edu/data/

[77] A. G. McArthur, N. Waglechner, F. Nizam, A. Yan, M. A. Azad, A. J. Baylay, K. Bhullar, M. J. Canova, G. De Pascale, L. Ejim *et al.*, "The comprehensive antibiotic resistance database," *Antimicrobial agents and chemotherapy*, vol. 57, no. 7, pp. 3348–3357, 2013.

[78] G. V. Gkoutos, P. N. Schofield, and R. Hoehndorf, "The units ontology: a tool for integrating units of measurement in science," *Database*, vol. 2012, p. bas033, 2012.

[79] K. Degtyarenko, P. De Matos, M. Ennis, J. Hastings, M. Zbinden, A. McNaught, R. Alcántara, M. Darsow, M. Guedj, and M. Ashburner, "Chebi: a database and ontology for chemical entities of biological interest," *Nucleic acids research*, vol. 36, no. suppl 1, pp. D344–D350, 2008.

[80] T. U. Consortium, "Uniprot: a hub for protein information," *Nucleic Acids Research*, vol. 43, no. D1, pp. D204–D212, 2015. [Online]. Available: http://nar.oxfordjournals.org/content/43/D1/D204.abstract

[81] D. Croft, A. F. Mundo, R. Haw, M. Milacic, J. Weiser, G. Wu, M. Caudy, P. Garapati, M. Gillespie, M. R. Kamdar *et al.*, "The reactome pathway knowledgebase," *Nucleic acids research*, vol. 42, no. D1, pp. D472–D477, 2014.

[82] U. Visser, S. Abeyruwan, U. Vempati, R. Smith, V. Lemmon, and S. Schurer, "BioAssay Ontology (BAO): a semantic description of bioassays and high-throughput screening results," *BMC Bioinformatics*, vol. 12, no. 1, pp. 257+, 2011.

[83] D. Maglott, J. Ostell, K. D. Pruitt, and T. Tatusova, "Entrez gene: gene-centered information at ncbi," *Nucleic acids research*, vol. 33, no. suppl 1, pp. D54–D58, 2005.

[84] S. Pletscher-Frankild, A. Pallejà, K. Tsafou, J. X. Binder, and L. J. Jensen, "Diseases: Text mining and data integration of disease–gene associations," *bioRxiv*, 2014.

[85] E. Bjorling and M. Uhlen, "Antibodypedia, a portal for sharing antibody and antigen validation data," *Molecular & Cellular Proteomics*, vol. 7, no. 10, pp. 2028–2037, 2008.

[86] S. Pletscher-Frankild, A. Palleja, K. Tsafou, J. X. Binder, and L. J. Jensen, "Diseases: Text mining and data integration of disease–gene associations," *Methods*, vol. 74, pp. 83–89, 2015.

[87] A. Santos, K. Tsafou, C. Stolte, S. Pletscher-Frankild, S. I. ODonoghue, and L. J. Jensen, "Comprehensive comparison of large-scale tissue expression datasets," *PeerJ*, vol. 3, p. e1054, 2015.

[88] B. F. Ontology, "Basic Formal Ontology (BFO) Project," last visited on 11/01/2013. [Online]. Available: http://www.ifomis.org/bfo

[89] A. Pease, I. Niles, and J. Li, "The suggested upper merged ontology: A large ontology for the semantic web and its applications," in *Working notes of the AAAI-2002 workshop on ontologies and the semantic web*, vol. 28, 2002.

[90] Y. Kazakov, M. Krötzsch, and F. Simančík, "The incredible elk," *Journal of automated reasoning*, vol. 53, no. 1, pp. 1–61, 2014.

[91] D. Tsarkov and I. Horrocks, "Fact++ description logic reasoner: System description," in *International Joint Conference on Automated Reasoning*. Springer, 2006, pp. 292–297.

[92] D. Tsarkov, "Incremental and persistent reasoning in fact++." in *ORE*, 2014, pp. 16–22.

[93] B. Glimm, I. Horrocks, B. Motik, G. Stoilos, and Z. Wang, "Hermit: an owl 2 reasoner," *Journal of Automated Reasoning*, vol. 53, no. 3, pp. 245–269, 2014.

[94] E. Sirin, B. Parsia, B. C. Grau, A. Kalyanpur, and Y. Katz, "Pellet: A practical owl-dl reasoner," *Web Semantics: science, services and agents on the World Wide Web*, vol. 5, no. 2, pp. 51–53, 2007.

[95] A. Steigmiller, T. Liebig, and B. Glimm, "Konclude: system description," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 27, pp. 78–85, 2014.

[96] B. Motik and U. Sattler, "Practical dl reasoning over large aboxes with kaon2," *Submitted for publication*, 2006.

[97] "BAO GitHub," last visited on 12/04/2017. [Online]. Available: https://github.com/BioAssayOntology/BAO

[98] A. M. Clark, N. K. Litterman, J. E. Kranz, P. Gund, K. Gregory, and B. A. Bunin, "Bioassay templates for the semantic web," *PeerJ Computer Science*, vol. 2, p. e61, 2016.