Published in final edited form as:

Methods Mol Biol. 2019 ; 1939: 49–69. doi:10.1007/978-1-4939-9089-4_4.

How to Develop a Drug Target Ontology – KNowledge Acquisition and Representation Methodology (KNARM)

Hande Küçük-McGinty^{1,4}, Ubbo Visser¹, Stephan Schürer^{2,3}

¹Department of Computer Science, University of Miami, Coral Gables, FL, USA

²Department of Molecular and Cellular Pharmacology, Miller School of Medicine, University of Miami, FL, USA

³Center for Computational Science, University of Miami, Coral Gables, FL, USA

⁴Collaborative Drug Discovery, Inc., Burlingame, CA, USA

Abstract

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. This era of ‘big data’ provides unprecedented opportunities for data driven research and ‘big picture’ models. However, in-depth analyses – making use of various data types and data sources, and extracting knowledge – has become a more daunting task. 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 oversimplifications, 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. Automated methods are still very limited and significant human expertise is required. In this chapter, we describe a methodology to systematize this knowledge acquisition and representation challenge, termed KNowledge Acquisition and Representation Methodology (KNARM). We then describe application of the methodology while implementing the Drug Target Ontology (DTO). We aimed to create an approach, involving domain experts and knowledge engineers, to build useful, comprehensive, consistent ontologies that will enable ‘big data’ approaches in the domain of drug discovery, without the currently common simplifications.

Keywords

knowledge acquisition; ontology; drug target ontology; semantic web; big data; semantic model; KNARM

Introduction

Gruber defines an ontology as a formal and explicit specification of a shared conceptualization for a domain of interest [1]. Almost three decades ago CommonKADS presented a widely accepted methodology for knowledge acquisition and ontology building

which described workflows for manual ontology building [2,3]. Following that, nearly two decades ago, the idea of using semantic web applications for representing life-sciences data and knowledge started gaining more attention in the life-sciences community [4–9]. Wache and colleagues [10] summarized the existing approaches and tools that can help scientists build powerful ontologies. Around the same time, Blagosklonny and colleagues [4,5] described how ontologies could be utilized for bioinformatics and drug discovery research, and that they can be powerful tools for life scientists. Today's well-cited, highly-accessed, well-described, and well-maintained ontologies such as Gene Ontology (GO) [11] and ChEBI [12] are among the first that showed how semantic web technologies could be wielded into creating common vocabularies. However, two decades after the above-mentioned milestones were developed, we lack sophisticated methodologies for knowledge acquisition and data representation using semantic web technologies [2,4–6,9,13–24].

Understanding the bigger picture without oversimplification, by making use of different databases available and extracting knowledge out of data, is becoming a more daunting task in the era of big data [18]. Life-sciences data are not only increasing in numbers but also is fitting more into the description of 'big data' by being too large, too dynamic, and too complex for conventional data tools to handle [20,25]. Screening technologies and computational algorithms have become very powerful, capable of creating diverse types of data increasingly faster and cheaper, such as gene sequencing, RNASeq gene expression, microscopy imaging data. Such large and dynamic data are typically scattered in different databases, in many different formats (i.e. traditional relational databases, NoSQL databases, ontologies, etc.). Additionally, currently available complex life-sciences data is not being efficiently translated into a format that is unambiguously readable and understandable by humans and machines. Furthermore, different types of data from gene expression, small molecule biochemical data to cell phenotyping via imaging, make it harder to manage, consolidate, integrate and analyze these data.

For our purposes, we define 'big data' as data that is high in volume (terabytes and larger), complex (interconnected with over 25 highly accessed databases [18] and over 600 ontologies [23]) with various types of data (varying from gene sequencing to cell imaging), and dynamic (growing exponentially [25,18]) for conventional data tools to store, manage, and analyze.

Related with our research, we have created two major ontologies: BioAssay Ontology (BAO) [20,21,26–28] and Drug Target Ontology (DTO) [20,24,29]. BioAssay Ontology (BAO) [28] aimed at describing and modeling assay data by using formal description logic (DL) and semantic web technologies. Drug Target Ontology (DTO) uses formal description logic to provide a classification of (protein) drug targets based on function and phylogenetics. Rich annotations of (protein) drug targets along with other chemical, biological, and clinical classifications and relations to diseases and tissue expression are also formally described in DTO using DL. Large number of different assays as well as their complexity with data types motivated us to look for a methodology that helps us acquire knowledge and formalize large amounts of data in the development of BAO.

Many different approaches have been presented for handling biological and chemical data for ontologies [1,9,11,15,23,30–40]. Currently, one focus is on combining existing databases and using machine operated data mining tools or relying on complete manual ontology building. However, creating a systematic methodology that effectively combines human and machine capabilities for extracting knowledge and representing it in an ontology is crucial for better understanding of the data. The existing literature lacks a formal methodology or workflow dealing with knowledge acquisition of large amounts of textual data and formalizing that information into a semantic knowledge model.

Confronted with that challenge and as part of our research, we created and implemented a hybrid methodology, KNowledge Acquisition and Representation Methodology (**KNARM**), that handles big data in life sciences in the form of large amounts of textual information, and translates it into axioms by using description logic (DL). In addition, the methodology and tools we built help update the ontologies faster and more accurately by semi-automating the ontology building process (see Figure 1). As our projects grew in size and focus, we also developed a systematically-deepening-modeling (SDM) approach for modeling life sciences data described in detail in the metadata creation and knowledge modeling section of this methodology.

Methods

KNowledge Acquisition and Representation Methodology (**KNARM**), consists of nine steps that allow domain experts and knowledge engineers to build useful, consistent ontologies formalizing biomedical knowledge. This methodology aims at acquiring knowledge from data scattered in different databases and ontologies, combining them in a meaningful fashion that is understandable by humans and machines by effectively combining human and machine capabilities. In this way, we attempt to allow users to understand, query, and analyze their data better by formalizing it using semantic web technologies.

Sub-language Analysis

Sub-language analysis is a technique for discovering units of information or knowledge, and the relationships among these units within existing knowledge sources, including published literature or corpora of narrative text. 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, we recommend an active reading by 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 might be:

- Search for proteins are in the same kinase branch as target X where there were validated chemical hits from external or internal sources.

- I have assay X. What are the other assays that have the same design or technology, but different targets?
- What 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.

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 who work in the same team. This step can be 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 helps identifying the knowledge units and key concepts of the data.

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 ontology in development, 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.

Metadata 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 metadata 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 and Figure 2).

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 [22]).

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 to perform an interview with collaborators who are involved in data creation as well as scientists who are not involved in data creation. The aim of the structured interview is to identify any important points that might have been missed by knowledge engineers and domain experts so far. In this step, the metadata identified is presented in context of the data obtained by knowledge engineers. This data could be dissected based on the metadata identified and dissected information could also be presented to the collaborators.

Knowledge Acquisition Validation

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

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 3 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.

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 previously reported a modular architecture [20,24,26] while building ontologies. The modular architecture allows easier management and sharing of ontology files, standardized vocabularies and axiomatic representations of knowledge. Modularization and ontology development can be performed manually. However, especially while building DTO, we created all vocabulary files and some of the axioms from the database using a java application, OntoJog [24], which will be released soon. This process adds another layer into the modularization to separate axioms that are automatically created by a software from axioms that are manually asserted in the ontology by expert knowledge engineers [20] (Fig. 3).

Ontology Validation

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 “Metadata Creation and Knowledge Modeling” or “Sub-language Recycling” step.

Implementation of the Drug Target Ontology (DTO) using KNARM

As a part of the Illuminating the Druggable Genome (IDG) [41] project, we designed and implemented the Drug Target Ontology (DTO) [29]. The long-term goal of the IDG project [41] is to catalyze the development of novel therapeutics that act on novel drug targets, which are currently poorly understood and poorly annotated, but are likely targetable. The project puts particular emphasis on the most common drug target protein families, G-protein coupled receptors (GPCR), nuclear receptors, ion channels, and protein kinases. Therefore,

we focused initially on formally classifying, annotating, and modeling these specific protein families in their role as drug targets and DTO is focused on proteins known as putative drug targets including many aspects to describe the relevant properties of these proteins in their role as drug target. While creating DTO, we further advanced the methodology and ontology architecture that we used for the BAO [26] and other smaller application ontologies from the LINCS project (LIFE ontology) [20]. A longer-term goal for DTO is to integrate it with the assays (formally described in BAO) that are used to identify and characterize small molecules that modulate these targets. This will result in an integrated drug discovery knowledge framework.

Sub-language Analysis and In-House Unstructured Interview for DTO—The initial interviews and sub-language analysis steps involved determining the different classifications of the drug targets and the properties of them. The IDG project defined “*drug target*” [24,29,42] as “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” [24]. Currently, DTO focuses on protein targets.

The IDG drug targets have been categorized as four major classes 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) [43].
4. **Tdark** 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].

DTO proteins have further been classified based on their structural (sequence/domains) and functional properties. Here we give 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 (PK). These 514 PKs are categorized in ten groups that are further subcategorized in 131 families and 82 subfamilies. The 62 non-protein kinases are categorized in five groups depending upon the substrate that are phosphorylated by these proteins. These five groups are further

sub-categorized in 25 families and seven 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 six 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 our reviews of the free-form text about the data in hand, the domain experts in the group provided help with answering the ontology engineers' questions. At times, the reviews of the free-form text were 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-like environment to better understand the various characteristics of proteins as drug targets – such as protein domains, binding ligands, functions, mutation, binding site, tissue expression, disease association, and many protein family specific concepts – and identify a pattern among the various kinds of molecular entities, their parts, functions, roles, related biomedical concepts, their uses as well as their functions in drug discovery assays and projects.

Above classifications of the proteins were performed by the domain experts and provided to the ontology engineers in excel sheets. Other classification questions were also discussed such as how to best classify mutated and modified proteins. It was decided that the best way to classify them was as a subclass of their wild-type proteins. The different properties identified in the first step are used in subsequent steps to create metadata, model the knowledge, and axiomize in the ontology building process.

Sub-language Recycling for DTO—While designing the ontology, we decided to add the UniProt IDs for the proteins and the ENTREZ IDs [30] for the genes as cross-references. 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.

We aimed at creating the Drug Target Ontology (DTO) as a comprehensive resource by importing existing information about the biological and chemical molecules that DTO contains. In this way, we aim to help the life-scientists query and retrieve information about the different drug targets that they are working on. To do that, we wrote various scripts using Java to retrieve information from different databases. These databases include UniProt and NCBI databases for ENTREZ IDs for the genes.

In addition to several publicly available databases and data including the DISEASES and TISSUES databases [44,47], we also used the collaborators TCRD databases [42] in order retrieve information about proteins, genes and their related target development levels (TDLs), as well as the tissue and disease information.

The DISEASES and TISSUES databases were developed in the Jensen group from several resources including using advanced text mining. They include a scoring system to provide a consensus of the various integrated data sources. We retrieved the proteins, with their tissue and disease relationships and the confidence scores that are given for the relationships. This data was loaded into our database and later used to create the ontology's axioms that refer to the probabilistic values of the relationships.

In addition to the larger scale information derived from the databases mentioned above, a vast amount of manual curation for the proteins and genes is performed in the team by the curators and 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 [48]. Protein kinases not covered by this resource were manually curated and classified mainly based on information in UniProt [49] and also the literature. Non-protein kinases were curated and classified based on their substrate chemotypes. We also added pseudokinases, which are becoming more recognized and relevant drug targets. We continue updating manual annotations and classifications as new data becomes available. Nuclear receptors were organized following the IUPHAR classification. GPCRs were classified based on information from several sources primarily using GPCRDB (<https://gpcrdb.org>) and IUPHAR as we have previously implemented in our GPCR ontology [50]. However, not all GPCRs were covered and we are aligning GPCR ontology with other resources to complete classification for several understudied receptors. We are also incorporating ligand chemotype-based classification. A basic classification of ion channels is available in IUPHAR [51]. Manual classification is in progress for 342 ion channels in order to provide a better classification is required including domain functions, subunit topology, and heteromer and homomer formation.

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. Ion channel domains were annotated and trans-membrane domains were identified; additional ion channel characteristics – such as regulatory and, gating mechanism, transported ion – were curated for ion channel drug targets. Additional sub-classification and annotation are in progress and will further improve that module.

In addition to the curated drug target family function-specific domain annotations, we generated comprehensive Pfam domain annotations for the kinase module [42]. The domain sequences were compared to the PDB chain sequences by BLAST and e-values were calculated. For significant hits, domain identities were computed using the EMBOSS software suite. These results were used to align and identify critical selectivity residues, such as gatekeeper and the hinge binding motif (publication in preparation). These annotations also allowed the integration with KINOMEScan assays from the LINCS project [52]. These domains are classified manually based on curated annotations to generate meaningful interpretable assertions in DTO.

Metadata Creation for DTO and Knowledge Modeling—Based on the sub-language analysis, the in house unstructured interview and sub-language recycling, the next step in formalizing descriptions is creating a set of metadata.

The metadata 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 to add the following axioms for the different protein classes (not a complete list):

- 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: Tclin, Tchem, Tbio, Tdark) from TCRD [42] were assigned using *has target development level* relationship and based on the criteria set by the IDG project. Each protein has an axiom annotating a target development level (TDL), i.e., Tclin, Tchem, Tbio and Tdark. The protein is linked to gene by *has gene template* relation.

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[44], 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 (not shown in TCRD)),
- *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).

Structured Interview for DTO—Based on the metadata created, we 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 [49].

With this step, the aim is to finalize names and types of concepts used in the metadata. 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.

Knowledge Acquisition Validation (KA Validation) for DTO—In this case after the metadata creation, various interviews, and reviews of the data, ontology engineer runs several scripts to check the consistency of the data. In addition, domain expert performs a thorough manual expert review of the extracted data. Before the database formation, metadata is also reviewed. Domain experts use metadata for grouping the extracted data. Modeling of the knowledge is confirmed with ontology engineer and domain expert reviews. Structured data then shared with the research scientists inside and outside of the team, especially with the scientists in the IDG project to make sure that the information contained was valid. Corrections, if necessary, were made on the data and metadata provided.

Database Formation for DTO—Previously we had engineered the BAO in a rigorous way using version control and Protégé. However, the modularization approach, although it has many benefits, requires tracking of many vocabulary files and ID ranges (to avoid conflicts). In addition to the vocabulary files, BAO has mostly expert constructed, manual

axioms defining assays and various related BAO concepts. For DTO on the other hand, much information was extracted from third party databases and then consolidated by curators and domain experts. Use of external resources also requires a mechanism for frequent updates of the ontology. To facilitate that process and to better track DTO modules, vocabularies, and ID ranges, a more efficient and less error-prone method to manage all information was required.

For the DTO, a new MySQL database was built to handle all data and metadata. Drug Target Ontology (DTO) uses various external databases and ontologies as information sources. Data from these databases is retrieved via web-based applications and in-house-built scripts. All data is stored in a relational database. The database schema for the DTO is provided in Figure 3.

Semi-Automated Ontology Building for DTO—The ontology is then built from this database in an automated way using a java application, OntoJog [24], which will be released and described separately soon. This process builds all vocabulary files, modules, and the axioms that can be automatically constructed given information in the database. In addition, all the external modules are built. The various vocabularies and modules are organized hierarchically via direct and indirect imports leading to DTO_core. DTO_core is then imported, along with the expert asserted axioms and the external modules, to DTO_complete (see Figure 4).

Knowledge Modeling of the Drug Target Ontology

In BAO, the formal descriptions of assays are manually axiomized. DTO, which is created for the IDG project, focuses on the bio-molecules and their binding partners such as the specific ions for ion-channeling proteins, or small molecule ligands for GPCRs, as well as their relationships to the specific diseases and tissues.

We use several tools, including Java, OWL API and Jena to build the ontology in a semi-automated way leveraging our local database and implementing a new modularization architecture given in detail below.

A New Modular Architecture for the Drug Target Ontology

The modular design of the DTO adds an additional layer on top of our previously reported modular architecture developed for BAO [26]. Specifically, we separate the module with auto-generated simple axioms, which are created using native-DTO concepts and/or various pieces of data imported from external databases after internal preprocessing. Following the auto-generated axioms, complex axioms are formed by ontologists or knowledge engineers. This way, auto-updates do not affect expert-formalized knowledge. The modular design is illustrated in Figure 4. The new approach is detailed below.

First, we determine an abstract horizon between TBox and ABox. TBox contains vocabularies and modules. Vocabularies which define the conceptualization without dependencies. The vocabularies are self-contained and well-defined with respect to the domain and they contain concepts, relations, and data properties (i.e. native-DTO concepts). We can have n of these vocabularies and modules which are combined into DTO_core.

Second, once the n number of native vocabularies and modules are defined, we can design modules that import modules from our domain of discourse, and also from third party ontologies. 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. We combine these modules 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 a different user group or area of research (e.g. kinases, GPCRs).

At the third level, the modules with axioms can be generated automatically are created. The auto-generated modules have interdependent axioms, i.e. these axioms can be generated using native-DTO concepts and/or concepts imported from external ontologies. At this level one could create any number of gluing modules, which import other modules without dependencies or with dependencies.

Fourth level contains axioms created manually. The manual modules are an optional level and they inherit the axioms created automatically. An example of axioms that may be seen in this level are axioms for protein modifications and mutations, which are knowledge modeling questions relevant to protein drug interactions.

Fifth level contains the TBox released based on the modules created from the fourth 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.

At the sixth and last level, the necessary modules ABoxes (i.e. instances of concepts created in the TBoxes) can be created. ABoxes can be loaded to a triple store or to a distributed file system (Hadoop DFS [53]) in a way that one could achieve pseudo-parallel reasoning. In another layer, using modules, we can 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 (see Figure 4).

Ontology Validation for DTO—Several check points for data validation are performed throughout the methodology. For example, after the data is extracted in “Sub-language recycling” step, the ontology engineer runs several scripts to check the consistency of the data. In addition, the domain expert performs a thorough manual expert review of the extracted data. The second check point in the methodology is during the “Database formation” step. Several scripts are used to check if the extracted data is properly imported to the database under the appropriate metadata categories as a unit of information along with the metadata. Once the “Semi-Automated Ontology Building” step is complete, the ontology engineer runs available reasoners to check the consistency of information. Furthermore, several SPARQL queries are run to flag any discrepancies. If there are any issues in the ontology, the ontology engineer and domain experts could decide to step back and re-perform the previous steps in the methodology. Another ontology validation script for DTO

is designed to read DTO vocabulary and module files and compare them to the previous version of the ontology. This script generates reports with all new (i.e. not present in the previous version), deleted (i.e. not present in the current version), and changed classes or properties based on their URIs and labels.

Any issues resulting from the tests are then discussed among ontology engineers and domain experts. GitHub is used to store the different versions of the ontology to help audit the quality control (QC) and ontology validation (OV) process. Once all the QC and OV procedures are completed with no errors then DTO is released on the public GitHub and its web page[29].

Notes

Complex life sciences data is fitting the ‘big data’ description due to the large volume (terabytes and larger), complexity (interconnected with over 25 highly accessed databases [18] and over 600 ontologies [23]), variety (many technologies generating different data types, such as gene sequencing, RNASeq gene expression, microscopy imaging data), and dynamic nature (growing exponentially and changing fast [25,18]). New tools are required to store, manage, integrate, and analyze such data while avoiding oversimplification. It is a challenge to design applications involving such ‘big data’ sets aimed at advancing human knowledge. One approach is to develop a knowledge-based integrative semantic framework, such as an ontology that formalizes how the different data types fit together given the current understanding of the domain of investigation. Building ontologies is time consuming and limited by the knowledge acquisition process, which is typically done manually via domain experts and knowledge engineers.

In this chapter, we described a methodology, KNowledge Acquisition and Representation Methodology (KNARM), as a guided approach, involving domain experts and knowledge engineers, to build useful, comprehensive, consistent ontologies that will enable ‘big data’ approaches in the domain of drug discovery, without the currently common simplifications. It is designed to help with the challenge of acquiring and representing knowledge in a systematic, semi-automated way. We applied this methodology in the implementation of the Drug Target Ontology (DTO).

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 above described methodology is still work in progress, it provided a systematic process for building concordant ontologies such as BioAssay Ontology (BAO) and Drug Target Ontology (DTO) [20]. 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 ontology generation and in particular knowledge acquisition, we are also developing new tools to improve the interaction between ontology developers and users given the reality of rapidly advancing knowledge and the need for more dynamic environment in which user requests can be incorporated in real time via direct information exchange with ontology developers. 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.

Acknowledgements and funding

This work was supported by 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’). The IDG-KMC and IDG-RDOC (<https://druggablegenome.net/>) are components of the Illuminating the Druggable Genome (IDG) project (<https://commonfund.nih.gov/idg>) awarded by the National Cancer Institute (NCI) and National Center for Advancing Translational Sciences (NCATS), respectively. The BD2K LINC DCIC is awarded by the National Heart, Lung, and Blood Institute through funds provided by the trans-NIH Library of Integrated Network-based Cellular Signatures (LINCS) Program (<http://www.lincsproject.org/>) and the trans-NIH Big Data to Knowledge (BD2K) initiative (<https://datascience.nih.gov/bd2k>). IDG, LINCS, and BD2K are NIH Common Fund projects.

References

1. Gruber TR (1993) Towards Principles for the Design of Ontologies Used for Knowledge Sharing. *International journal of human-computer studies* 43 (5–6):907–928
2. CommonKADS CommonKADS. <http://commonkads.org/>.
3. Schreiber GaBob W and de Hoog Robert and Akkermans Hans and Van de Velde Walter (1994) CommonKADS: A comprehensive methodology for KBS development. *IEEE expert* 9 (6):28–37
4. Barnes JC (2002) Conceptual biology: a semantic issue and more. *Nature* 417 (6889):587–588
5. Blagosklonny MV, Pardee AB (2002) Conceptual biology: unearthing the gems. *Nature* 416 (6879):373–373 [PubMed: 11919607]
6. Benson DA, Karsch-Mizrachi I, Lipman DJ, Ostell J, Rapp BA, Wheeler DL (2000) GenBank. *Nucleic acids research* 28 (1):15–18 [PubMed: 10592170]
7. Heflin J, Hendler J (2000) Semantic interoperability on the web. Maryland University, College Park, Department of Computer Science, Maryland, USA
8. Noy NF, Fergerson RW, Musen MA (2000) The knowledge model of Protege-2000: Combining interoperability and flexibility In: *Knowledge Engineering and Knowledge Management Methods, Models, and Tools*. Springer, pp 17–32
9. Stevens R, Goble CA, Bechhofer S (2000) Ontology-based knowledge representation for bioinformatics. *Briefings in bioinformatics* 1 (4):398–414 [PubMed: 11465057]
10. Wache H, Voegelé T, Visser U, Stuckenschmidt H, Schuster G, Neumann H, Hübner S Ontology-based integration of information-a survey of existing approaches In, 2001 IJCAI-01 workshop: ontologies and information sharing. Citeseer, pp 108–117
11. Yeh I, Karp PD, Noy NF, Altman RB (2003) Knowledge acquisition, consistency checking and concurrency control for Gene Ontology (GO). *Bioinformatics* 19 (2):241–248 [PubMed: 12538245]
12. Degtyarenko K, De Matos P, Ennis M, Hastings J, Zbinden M, McNaught A, Alcántara R, Darsow M, Guedj M, Ashburner M (2008) ChEBI: a database and ontology for chemical entities of biological interest. *Nucleic acids research* 36 (suppl 1):D344–D350 [PubMed: 17932057]
13. Baader F, Calvanese D, McGuinness DL, Nardi D, Patel-Schneider PF (2010) *The Description Logic Handbook: Theory, Implementation and Applications*. 2nd edn Cambridge University Press, New York, NY, USA

14. Buchanan BG, Barstow D, Bechtal R, Bennett J, Clancey W, Kulikowski C, Mitchell T, Waterman DA (1983) Constructing an expert system. *Building expert systems* 50:127–167
15. Natale DA, Arighi CN, Blake JA, Bona J, Chen C, Chen S-C, Christie KR, Cowart J, D'Eustachio P, Diehl AD, Drabkin HJ, Duncan WD, Huang H, Ren J, Ross K, Ruttenberg A, Shamovsky V, Smith B, Wang Q, Zhang J, El-Sayed A, Wu CH (2011) The representation of protein complexes in the Protein Ontology (PRO). *BMC bioinformatics* 12 (1):1 [PubMed: 21199577]
16. Clark AM, Litterman NK, Kranz JE, Gund P, Gregory K, Bunin BA, Cao L (2016) BioAssay templates for the semantic web Data science: Challenges and directions. *PeerJ Computer Science* 2 (8):e61
17. Belleau F, Nolin M-A, Tourigny N, Rigault P, Morissette J (2008) Bio2RDF: towards a mashup to build bioinformatics knowledge systems. *Journal of biomedical informatics* 41 (5):706–716 [PubMed: 18472304]
18. Cook CE, Bergman MT, Finn RD, Cochrane G, Birney E, Apweiler R (2015) The European Bioinformatics Institute in 2016: data growth and integration. *Nucleic acids research* 44 (D1):D20–D26 [PubMed: 26673705]
19. Hitzler P, Krötzsch M, Rudolph S (2009) *Foundations of Semantic Web Technologies*. Chapman and Hall (CRC), USA
20. Küçük-McGinty H, Metha S, Lin Y, Nabizadeh N, Stathias V, Vidovic D, Koleti A, Mader C, Duan J, Visser U, Schurer S IT405: Building Concordant Ontologies for Drug Discovery. In: International Conference on Biomedical Ontology and BioCreative (ICBO BioCreative 2016), Oregon, USA, 2016.
21. Schurer SC, Vempati U, Smith R, Southern M, Lemmon V (2011) BioAssay Ontology Annotations Facilitate Cross-Analysis of Diverse High-Throughput Screening Data Sets. *Journal of Biomolecular Screening* 16 (4):415–426 [PubMed: 21471461]
22. Smith B, Ashburner M, Rosse C, Bard J, Bug W, Ceusters W, Goldberg LJ, Eilbeck K, Ireland A, Mungall CJ, others (2007) The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature biotechnology* 25 (11):1251–1255
23. Whetzel PL, Noy NF, Shah NH, Alexander PR, Nyulas C, Tudorache T, Musen MA (2011) 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* 39 (suppl 2):W541–W545 [PubMed: 21672956]
24. Lin Y, Mehta S, Küçük-McGinty H, Turner JP, Vidovic D, Forlin M, Koleti A, Nguyen D-T, Jensen LJ, Guha R, Mathias SL, Ursu O, Stathias V, Duan J, Nabizadeh N, Chung C, Mader C, Visser U, Yang JJ, Bologna CG, Oprea TI, Schürer SC (2017) Drug Target Ontology to Classify and Integrate Drug Discovery Data. *Journal of Biomedical Semantics* 8 (1):50 [PubMed: 29122012]
25. Ma'ayan A (2017) Complex systems biology. *Journal of The Royal Society Interface* 14 (134):1742–5689
26. Abeyruwan S, Vempati UD, Küçük-McGinty H, Visser U, Koleti A, Mir A, Sakurai K, Chung C, Bittker JA, Clemons PA, Chung C, Bittker JA, Clemons PA, Brudz S, Siripala A, Morales AJ, Romacker M, Twomey D, Bureeva S, Lemmon V, Schürer SC (2014) Evolving BioAssay Ontology (BAO): modularization, integration and applications. *Journal of biomedical semantics* 5 (Suppl 1):S5 [PubMed: 25093074]
27. BAOsearch. <http://baosearch.ccs.miami.edu/>.
28. Visser U, Abeyruwan S, Vempati U, Smith R, Lemmon V, Schurer S (2011) BioAssay Ontology (BAO): a semantic description of bioassays and high-throughput screening results. *BMC Bioinformatics* 12 (1)
29. Drug Target Ontology. <http://drugtargetontology.org/>.
30. Brinkman RR, Courtot M, Derom D, Fostel JM, He Y, Lord P, Malone J, Parkinson H, Peters B, Rocca-Serra P, Ruttenberg A, Sansone SA, Soldatova LN, Stoeckert CJ Jr., Turner JA, Zheng J (2010) Modeling biomedical experimental processes with OBI. *J Biomed Semantics* 1 (Suppl 1):S7 [PubMed: 20626927]
31. Callahan A, Cruz-Toledo J, Dumontier M (2013) Ontology-Based Querying with Bio2RDF's Linked Open Data. *Journal of Biomedical Semantics* 4 (Suppl 1):S1 [PubMed: 23735196]

32. Ceusters W, Smith B (2006) A realism-based approach to the evolution of biomedical ontologies. Annual Symposium proceedings/AMIA Symposium AMIA Symposium:121–125
33. Consortium TGO (2015) Gene Ontology Consortium: going forward. Nucleic Acids Research 43 (D1):D1049–D1056. doi:10.1093/nar/gku1179 [PubMed: 25428369]
34. Decker S, Erdmann M, Fensel D, Studer R (1999) Ontobroker: Ontology based access to distributed and semi-structured information In: Database Semantics. Springer, pp 351–369
35. Gruber TR (1993) A translation approach to portable ontology specifications. Knowledge acquisition 5 (2):199–220
36. Köhler J, Philippi S, Lange M (2003) SEMEDA: ontology based semantic integration of biological databases. Bioinformatics 19 (18):2420–2427 [PubMed: 14668226]
37. Ontology BF Basic Formal Ontology (BFO) Project. <http://www.ifomis.org/bfo>.
38. Pease A, Niles I, Li J The suggested upper merged ontology: A large ontology for the semantic web and its applications In, 2002 Working notes of the AAAI-2002 workshop on ontologies and the semantic web.
39. Sure Y, Erdmann M, Angele J, Staab S, Studer R, Wenke D (2002) OntoEdit: Collaborative ontology development for the semantic web. Springer, USA
40. Welty CA, Fikes R A Reusable Ontology for Fluents in OWL. In, 2006 Formal Ontology in Information Systems Frontiers in Artificial Intel. and Apps. IOS, pp 226–236
41. NIH Illuminating the Druggable Genome | NIH Common Fund. <https://commonfund.nih.gov/idg/index>.
42. TCRD Database. <http://habanero.health.unm.edu/tcrd/>.
43. Hamosh AaS, Alan F and Amberger Joanna S and Bocchini Carol A and McKusick Victor A (2005) Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders. Nucleic acids research 33:D514–D517 [PubMed: 15608251]
44. Pletscher-Frankild SaP, Albert and Tsafou Kalliopi and Binder Janos X and Jensen Lars Juhl (2015) DISEASES: Text mining and data integration of disease–gene associations. Methods 74:83–89 [PubMed: 25484339]
45. NCBI. <https://www.ncbi.nlm.nih.gov/gene/about-generif>. 2017
46. Kiermer V (2008) Antibodypedia. Nature Methods 5 (10):860–860
47. Santos A, Tsafou K, Stolte C, Pletscher-Frankild S, O'Donoghue SI, Jensen LJ (2015) Comprehensive comparison of large-scale tissue expression datasets. PeerJ 3:e1054 [PubMed: 26157623]
48. Sugan and the Salk Institute. (2012). <http://kinase.com/human/kinome/phylogeny.html>.
49. Consortium TU (2015) UniProt: a hub for protein information. Nucleic Acids Research 43 (D1):D204–D212. doi:10.1093/nar/gku989 [PubMed: 25348405]
50. Przydzial MJ, Bhatarai B, Koleti A, Vempati U, Schürer SC (2013) GPCR ontology: development and application of a G protein-coupled receptor pharmacology knowledge framework. Bioinformatics 29 (24):3211–3219 [PubMed: 24078711]
51. Pawson AJaS, Joanna L and Benson Helen E and Faccenda Elena and Alexander Stephen PH and Buneman O Peter and Davenport Anthony P and McGrath John C and Peters John A and Southan Christopher (2013) The IUPHAR/BPS Guide to PHARMACOLOGY: an expert-driven knowledgebase of drug targets and their ligands. Nucleic acids research 42 (D1):D1098–D1106 [PubMed: 24234439]
52. Vidovi D, Koleti A, Schürer SC (2014) Large-scale integration of small molecule-induced genome-wide transcriptional responses, Kinome-wide binding affinities and cell-growth inhibition profiles reveal global trends characterizing systems-level drug action. Frontiers in Genetics 5
53. Shvachko K, Kuang H, Radia S, Chansler R The Hadoop Distributed File System In, Washington, DC, USA, 2010 Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST). IEEE Computer Society, pp 1–10

Figure 1. Steps of KNowledge Acquisition and Representation Methodology (KNARM). This figure shows the nine steps and flow of KNARM. Following agile principles, there are feedback loops present before finalizing ontologies. The circular flow also represents that ontology building process is a continuous effort, allowing ontology engineers to iteratively add more concepts and knowledge.

Figure 2.

Conceptual modeling example, showing modeling of an example kinase (ABL1) and how some of its axioms relate to the different ontologies created using KNARM.

Figure 3.
Excerpt of the database schema used to create DTO.

Figure 4. Modular architecture of DTO showing the core principles and levels of DTO's architecture with direct and indirect imports.