Building a Drug Ontology based on RxNorm and Other Sources

Josh Hanna,* Eric Joseph, Mathias Brochhausen, and William R. Hogan
Division of Biomedical Informatics, University of Arkansas for Medical Sciences, Little Rock, Arkansas, USA

ABSTRACT

We built the Drug Ontology (DrOn) to meet the requirements of our comparative-effectiveness research use case, because existing artifacts had flaws too fundamental and numerous to meet them. However, one of the obstacles we faced when creating the Drug Ontology (DrOn) was the difficulty in reusing drug information from existing sources. The primary external source we have used at this stage in DrOn's development is RxNorm, a standard drug terminology curated by the National Library of Medicine (NLM). To build DrOn, we (1) mined data from historical releases of RxNorm and (2) mapped many RxNorm entities to Chemical Entities of Biological Interest (ChEBI) classes, pulling relevant information from ChEBI while doing so.

We built DrOn in a modular fashion to facilitate simpler extension and development of the ontology and to allow reasoning and construction to scale. Classes derived from each source are serialized in separate modules. For example, the classes in DrOn that are programmatically derived from RxNorm stored in a separate module and subsumed by classes in a manually built, realist, upper-level module of DrOn with terms such as 'clinical drug role', 'tablet', 'capsule', etc.

1 INTRODUCTION

An ontology of drugs could be useful for a variety of purposes, such as comparative effectiveness research (Olsen, 2011), clinical decision support (Broverman, 1998; Sperzel, 1998; Kim, 2001), and clinical data warehousing and integration (Broverman, 1998; Nelson, 2011; Palchuk, 2010; Parrish, 2006; Kim, 2001), among others. At present, no existing resource was sufficient for our use cases in these domains (see our sister paper (Hogan, 2013)), and therefore we decided to build the Drug Ontology (DrOn). Minimally, no existing resource contains in its current version a historically comprehensive list of National Drug Codes (NDCs).

RxNorm (Nelson, 2011)—a standard drug terminology maintained by the U.S. National Library of Medicine (NLM)—includes normalized names and relationships extracted from several proprietary drug knowledge bases. Because of (1) the large amount of drug information maintained within RxNorm, (2) the fact that it is freely available, and (3) the fact that much of it is available under a permissive license, RxNorm is a good candidate for a source of information to create a formal drug ontology.

RxNorm is focused primarily on prescription and over-the-counter drugs that are currently available in the United States. It uses Concept Unique Identifiers called RXCUIs to catalog and relate information.

At this stage of DrOn development, we are interested in the ability to query for National Drug Codes (NDCs). The NDC is a unique identifier that the Drug Listing Act of 1972 requires companies to report to the Food and Drug Administration (FDA). RxNorm associates each NDC with a drug product via the RXCUI. Although our requirement is to have a comprehensive, historical list of NDCs, RxNorm maintains only currently active NDCs in its current release. So tracking all NDCs and the RXCUIs with which they have been associated over historical releases of RxNorm is key to building DrOn.

Moreover, NDCs are often lost with no explanation when an RXCUI is retired, especially in releases of RxNorm prior to 2009. This situation necessitates careful tracking to ensure that all valid NDCs (and, indeed, any useful information) associated with a retired RXCUI can be associated with the most recent RXCUI that refers to the same entity.

While other drug information sources exist, none of them was sufficient. Our requirements include (1) a historically comprehensive list of NDCs, (2) correctness with respect to pharmacy and biomedical science, (3) logically consistent, correct axioms that do not entail untrue or inconsistent inferences, and (4) interoperability with other ontologies for translational science. In previous work, we analyzed RxNorm, the National Drug File – Reference Terminology, SNOMED CT, Chemical Entities of Biological Interest (ChEBI), an OWL conversion of the Anatomical and Therapeutic Chemical classification system, DrugBank, PharmGKB, and other sources (Hogan, 2013) and found that none of them met these requirements.

In this paper, we describe how we build DrOn from historical releases of RxNorm, while navigating these pitfalls. In addition, during the build process, we map drug ingredients from RxNorm to the Chemical Entities of Biological Interest (ChEBI) ontology (de Matos, 2010). As a result, we import hundreds of ChEBI classes and their associated URIs, labels, etc. into DrOn.

2 METHODS

The overall workflow of the extraction and translation process has three main steps:

1. Mining RxNorm for relevant entities, including information found only in older releases.
2. Extracting, Transforming, and Loading (ETL) the data mined from RxNorm into a normalized Relational Database Management System (RDBMS).
3. Translating the normalized RDBMS into an OWL 2.0 artifact.

* To whom correspondence should be addressed: jhanna@uams.edu

Each of these three steps is further subdivided into substeps that we explain in detail below.

2.1 Mining RxNorm

We first download the raw RxNorm data files directly from the NLM website, specifically the UMLS (or Unified Medical Language System) Terminology Services (UTS) site (U.S. National Library of Medicine, 2011) and import them into a locally hosted RDBMS using the scripts provided by the NLM. Additionally, to support maintenance of comprehensive information over time, we created and maintain two additional tables that store all the information that we extract from each release of RxNorm (a subset of all the information). We describe these tables in detail below (sections 2.1.3 and 2.1.4).

Currently, we include information from every version of RxNorm released between June, 2008 and February, 2013 in DrOn. The reason is that the June, 2008 release was the first one that includes RxNorm-curated NDCs.

It should be noted that we use only information curated within RxNorm and not any information from its sources directly, and thus our overall process is allowable under the UMLS license (all content reused in DrOn is Level 0).

2.1.1 RxNorm Files

The next step is to extract all relevant information from the files downloaded from the UTS site. RxNorm comes as a set of nine Rich Release Format (RRF) files, each of which contains a specific subset of the total information. However, we do not use all nine files in our build process.

We process RXNSAT.RRF, RXNCONSO.RRF, RXNCUI.RRF, and RXNCUICHANGES.RRF, RXNSAB.RRF. Table 1 shows the information we mined from each file.

File	Extracted Information
RXNSAT.RRF	NDCs and RXCUIs
RXNCONSO.RRF	RXCUI attributes
RXNCUI.RRF	retired RXCUIs with provenance
RXNCUICHANGES.RRF	RXCUI provenance
RXNSAB.RRF	RxNorm version information

Table 1*: The RxNorm files and the information mined from each.*

There are four different term types in RXNCUI.RRF that are relevant to DrOn. They are: (1) Semantic Clinical Drug Forms (SCDFs), (2) Semantic Clinical Drugs (SCDs), (3) Semantic Branded Drugs (SBDs), and (4) Ingredients (IN). RxNorm treats NDCs as attributes of an SCD or SBD rather than a separate term type.

2.1.2 RXCUI Provenance

Tracking entities within RxNorm requires tracking the RXCUIs to which they are attached. This can be a difficult task. Any RXCUIs that have been entered in error are retired. Additionally, if two RXCUIs refer to the same entity, they are consolidated and either one of them is retired while the other remains or a new RXCUI is created and both older RXCUIs are retired. Prior to the April 2009 release of RxNorm, no comprehensive list of retired RXCUIs was provided. Furthermore, the reasons for retirement are not always well-documented, making it difficult to distinguish between RXCUIs that have been retired because they are nonsense and ones that have been replaced or merged. For instance, as of this writing, there are 40 RXCUIs with 210 associated NDCs that are no longer contained in the most recent release of RxNorm, however, there is no record of why these RXCUIs were removed.

2.1.3 Extraction of National Drug Codes (NDCs) and related RXCUIs

To facilitate the tracking of NDCs, we have created an additional table, *NDC_COMP*, that contains a comprehensive list of all RxNorm-curated NDCs from all releases of RxNorm since June 2008 (when they first appeared) and their corresponding RXCUIs. To generate this table, we parse the RXNSAT.RRF data file contained in each release of RxNorm. Any entry in the file whose source is RxNorm and is annotated as being an NDC is extracted from the file, along with its associated RXCUI, and imported into our *NDC_COMP* table. We also store the version from which each NDC was mined, which is parsed from the RXNSAB.RRF data file as mentioned in Section 2.1.1.

2.1.4 Tracking Provenance

The second of the two additional tables is a master conversion table, *DEPRECATED_RXCUIS,* which we use to track the current status of each retired RXCUI. This table contains two fields: *old_rxcui* and *new_rxcui.* The *old_rxcui* field contains a retired RXCUI, and the *new_rxcui* field contains the current RXCUI to which the retired RXCUI's information is now associated. The *new_rxcui* field may also contain a status code if the retired RXCUI's information is unable to be tracked because it was entered in error or split into multiple new RXCUIs. These special status codes are "ERROR" for RXCUIs that have been entered in error and "S_RXNCUI" for RXCUIs which have been split. Because RxNorm does not document why an erroneous RXCUI was entered in error, we are unable to do further processing on them or their associated information. For the RXCUIs which are split, it may be possible to track some of their associated information, but it is not always clear which information belongs to which child RXCUI and this issue requires manual intervention at present.

Our *DEPRECATED_RXCUIS* table is updated with each release of RxNorm through the following procedure:

1. First, we extract any RXCUIs from the comprehensive *NDC_COMP* table, built in Section 2.1.3, that can no longer be found in the *RXNCONSO.RRF* file being imported. We then import these RXCUIs into the *old_rxcui* column of our *DEPRECATED_RXCUIS* table. Because the *RXNCONSO.RRF* file contains all current RXCUIs, any RXCUIs that meet the above criteria must have been retired.

2. Next, using the RxNorm-curated *RXNCUI* table, we update all entries in the *new_rxcui* column. The RXNCUI table contains a *cui1* field containing a retired RXCUI, a *cui2* field containing the RXCUI into which the retired RXCUI's information has been merged, and a *cardinality* column contains the number of RXCUIs into which the information has been merged. Any RXCUI that has been entered in error is indicated by an entry in which the value of the *cui1* field is equal to the value of the *cui2* field. Additionally, any entry with a *cardinality* greater than 1 indicates that the RXCUI has been split. These are indicated in our table by setting the *new_rxcui* entry to "ERROR" and "S_RXNCUI", respectively. As of this writing, 768 RXCUIs and 3,484 associated NDCs are reported by RxNorm to have been entered in error and are therefore not included in DrOn. Additionally, 187 RXCUIs and 3,126 associated NDCs have been split. Both these RXCUIs and NDCs have also been left out of DrOn (for the time being) due to the difficulty of determining which information from the parent RXCUI belongs to which child RXCUI.

3. Finally, we compute the transitive closure, associating each RXCUI with the latest RXCUI that refers to the same entity with no intervening steps in our *DEPRECATED_RXCUIS* table. Because this table is updated with each release of RxNorm, occasionally an RXCUI in the *new_rxcui* field is retired. In such situations, the *new_rxcui* field is updated as described in Step 2, and a new row in the table is created with the newly-retired RXCUI set as the *old_rxcui*, and the *new_rxcui* field is set to match the updated *new_rxcui* from the original entry.

2.2 Mapping to ChEBI

The process maps ingredients (IN entity type) extracted from RxNorm to ChEBI entities where possible. We accomplish this step through a simple Java console application (that we built) that compares the labels of ingredients pulled from RxNorm with annotations in ChEBI. Any exact matches between the names or synonyms of RxNorm IN entities and ChEBI annotations were assumed to be referring to the same entity type and thus the ChEBI URIs were used in DrOn. Three different annotation types from ChEBI are used in the mapping process: *label, related_synonym*, and *exact_synonym*. To date, we import into DrOn ~750 classes (including URI and label and other annotations) from ChEBI: roughly 500 matches were found via *label*, 250 were found via *related_synonym*, and only two were found via *exact_synonym.* Many of the ingredients found in RxNorm are extracts of various plants, *e.g. ginger extract*, which we would not expect to find in ChEBI.

Somatropin (also known as somatotroin or human growth hormone) was erroneously associated with the ChEBI role 'growth hormone'. This error, once noticed, was fixed. The term is now mapped to the Protein Ontology URI that represents the protein molecule somatotropin.

We assigned a DrOn URI to every ingredient that was not found in ChEBI via this process.

2.3 ETL into a Normalized Format

There are five RxNorm entity types we were initially interested in pulling from RxNorm. These are the following: ingredient, clinical drug form, clinical drug, branded drug, and national drug code (NDC). Additionally, we wanted to represent a number of ingredient dispositions. Figure 1 shows these six entity types and the relationships between them. They are described in some detail next. It should be noted that the entities the NDC class represent are not the codes themselves, but instead the packaged drug products that the NDCs represent. Additionally, every DrOn entity that corresponds to a RxNorm entity is annotated with the corresponding RXCUI.

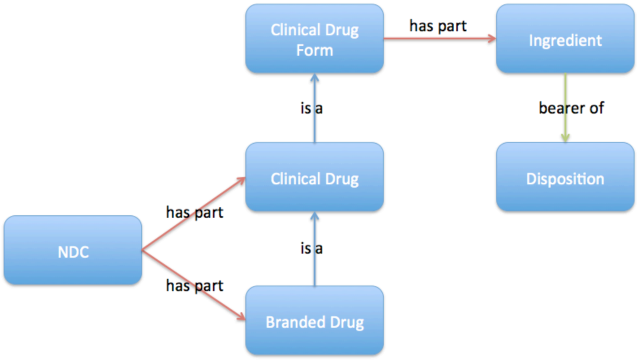

Fig. 1: The entity types of DrOn and their relationships as stored in the normalized format

2.2.1 Entity types

The **ingredient** entities represent the types of molecules that are present in a drug product and have an active biological role. The URIs of ingredients, where possible, are taken from the Chemical Entities of Biological Interest (ChEBI) ontology as described above. Examples of these include acetaminophen, sulfur, and ephedrine. There are 7,848 unique ingredients in DrOn.

The **disposition** entities represent dispositions that molecules bear (see Hogan, 2013). There are, as of now, six molecular dispositions in DrOn. They are:

1. *non-activating competitive beta-adrenergic receptor binding disposition* (i.e., beta-adrenergic blockade)
2. *function-inhibiting hydrogen/potassium adenosine triphosphatase enzyme (H+/K+ ATPase) binding disposition* (i.e., proton-pump inhibition)
3. *function-inhibiting L-type voltage-gated calcium channel binding disposition* (i.e., a subtype of calcium-channel blockade)
4. *function-inhibiting vitamin K epoxide reductase binding disposition* (i.e., a type of Vitamin K antagonism)
5. *function-inhibiting Na-K-Cl cotransporter 2 (NKCC2) binding disposition* (i.e., NKCC2 inhibition)
6. *function-inhibiting T-type calcium channel binding disposition* (i.e., another subtype of calcium-channel blockade)

These six dispositions were chosen based on their biological importance and relevance to ongoing comparative effectiveness research at the University of Arkansas for Medical Sciences. There is no direct correspondence between DrOn dispositions and RxNorm, because by design RxNorm lacks information about mechanism of action. Instead, the relationships between DrOn dispositions and ingredients was mined from ChEBI, although ChEBI treats the same realizable entities that we represent here as roles (see Hogan, 2013 for more details). Table 2 shows the associated ChEBI role from which the ingredient relationships for the three dispositions were mined. The other three dispositions not in the table were curated manually by the authors.

DrOn Disposition	ChEBI Role
non-activating competitive beta-adrenergic receptor binding disposition	beta-adrenergic antagonist
function-inhibiting hydrogen/potassium adenosine triphosphatase enzyme (H+/K+ ATPase) binding disposition	proton pump inhibitor
function-inhibiting L-type voltage-gated calcium channel binding disposition	calcium channel blocker

***Table 2**: The ChEBI roles used to mine DrOn disposition-ingredient relationships*

Function-inhibiting T-type calcium channel binding disposition was included because we erroneously associated *ethosuximide* and *function-inhibiting L-type voltage-gated calcium channel binding disposition*. This error was not due to any particular oversight of ChEBI but an artifact caused by the more specific nature of DrOn's dispositions as compared to ChEBI's more general c*alcium channel blocker*.

The **Clinical Drug Form** (CDF) entities represent a type of drug product, dose form (*e.g.* drug tablet), and, often, the intended route of administration (*e.g.* oral ingestion) without brand or strength information. These correspond to SCDFs in RxNorm. Examples of CDFs include *estradiol transdermal patch, iodine topical solution,* and *menthol crystals.* There are 14,035 unique CDFs in DrOn.

The **Clinical Drug** (CD) entities represent drug products with specific dosage/strength/form information. They are related to the CDF by an is-a relationship. For example, every *aspirin 325 MG enteric coated tablet* (CD) is a *aspirin enteric coated tablet* (CDF). DrOn contains 34,560 CDs.

The **Branded Drug** (BD) entities represent brand-name drug products with specific dosage/strength/form information. The drug products that BDs represent are related to the products that CDs represent by an is-a relationship. There are 21,248 unique BDs in DrOn.

The **National Drug Code** (NDC) entities represent a drug product and its packaging. These entities are distinct from entities represented by BDs or CDs, instead containing some number of instances of drug products represented by CDs/BDs, for example a 100-tablet bottle of aspirin 325 mg tablets. There are 390,813 unique NDC entities in DrOn.

DrOn Entity Type	RxNorm Entity Type
CDF	SCDF
CD	SCD
BD	SBD
Ingredient	IN
NDC	SCD or SBD Attribute

***Table 3**: The associated RxNorm entity type for each DrOn entity type except disposition.*

2.3.2 RDBMS design

The RDBMS design representing the normalized format of the entity types described above is simple. There are 5 core tables, one for each entity type. These are as follows: *clinical_drug_form, clinical_drug, branded_drug, ndc, ingredient,* and *disposition.*

Additionally, there are two tables storing provenance information from RxNorm, such as the version of RxNorm in which each RxCUI was found. These are *rxcui* and *rxnorm*. These are completely separate from the core entity tables to allow for incorporation of other data.

Many-to-many tables representing the relationships between the various entities are omitted in the interest of brevity. However, all of the relationships shown in 1 are also represented in RDBMS.

2.3.2 ETL process

The ETL process is done in four major steps:

1. First, we initialize the rxcui and rxnorm tables. This includes mapping every deprecated RXCUI to the most recent RXCUI that identifies the same object, either to an RXCUI from the current set or another deprecated, but not entered in error, RXCUI.
2. Next, we initialize the ndc table. This primarily involves copying all the NDCs found in the mining process (without the duplication caused by storing NDCs multiple times during the mining process) and associating them with the relevant RXCUI.
3. Next, we create the ingredients, CDFs, CDs, and BDs from the associated RxNorm type. This includes maintaining the proper relationships between the various entities (e.g. associating the correct ingredients with each CDF).
4. Finally, we associate each NDC with the appropriate CD or BD. This primarily involves following the provenance trail of RXCUIs provided in step 1.

2.4 Creating the OWL 2.0 Artifact

We use the OWLAPI 3.4.3 (Horridge, 2011), Scala 2.10 (Odersky, 2004), and Slick 1.0.0 (Typesafe, 2013) to extract the entities from our internal representation and transform them into an OWL artifact. This process is subdivided into the following steps:

1. Extract the ingredients, using ChEBI URIs where appropriate.
2. Extract the dispositions and associated them via the *bearer_of* relation to the one or more ingredients.
3. Extract the clinical drug forms and associate them via the *has_proper_part* relation to the one or more ingredients.
4. Extract the clinical drugs and assert they are a subclass of the appropriate clinical drug form.
5. Extract the branded drugs and assert that they are a subclass of the appropriate clinical drug.
6. Extract the NDCs and assert that they are related to one branded drug or one clinical drug via the *has_proper_part* relation.

This ordering of the steps is deliberate. Each step depends on one or more previous steps.

Since the RDBMS structure defined above represents the entities and their relationships already, this process is fairly straightforward.

2.4.1 Modularization

The ability to incorporate additional sources of information has been a key requirement for the build process. To help facilitate this, we developed DrOn in a modular fashion. Currently, DrOn has five different modules: **dron-full**, **dron-chebi**, **dron-rxnorm**, **dron-pro**, and **dron-upper**.

The **dron-full** module is simply a connector that imports the other modules. It is so named on the assumption that certain subsets of the modules may prove useful enough to warrant lighter versions of the ontology.

The **dron-chebi** module contains all of the annotations for the ingredients mapped to ChEBI (as described in Section 2.2). It contains everything imported from ChEBI.

The **dron-rxnorm** module contains all of the information mined from RxNorm, which, at this point of the ontology's development, is the bulk of DrOn's information. It includes the NDCs, though we plan to split the NDCs from the rest of the RxNorm module in future work.

The **dron-pro** module includes everything imported from the Protein Ontology (PRO). At present, it is very small and only contains the 'protein' and 'somatotropin' classes from PRO. As stated above, we imported these classes to represent somatotropin as a drug ingredient.

The **dron-upper** module contains the hand-created upper level ontology that the other modules are mapped on to (see Hogan, 2013).

This modularization brings two major benefits: development simplicity and increased scalability. By creating logical divisions and well-defined interfaces between the modules, we can more easily maintain each module separately without significantly affecting the other modules. Additionally, as each module grows in size, we can shard the processing and creation of the ontologies to different servers, making scaling the process simpler.

3 DISCUSSION

We developed an ontology, DrOn, that contains information programmatically derived from three different sources (RxNorm, ChEBI, and PRO) during its build process. Because it is derived from general-purpose resources, we believe DrOn can serve many use cases beyond our current ones (although this conjecture requires further research). We plan on adding additional sources in the future to maintain current information in DrOn, with more immediate plans to include information from Structured Product Labels. As such, we built our internal representation to maintain provenance information of the sources separately, ensuring that we can both track the provenance of the various entities as the ontology develops and add new sources without adversely affecting the existing ontology.

DrOn follows OBO Foundry guidelines and is currently listed on the OBO Foundry website as a candidate ontology. In additional to the mining detailed above, DrOn imports BFO 1.1 and includes terms MIREOTed from the Relationship Ontology and BFO 2.

The development site and issue tracker for DrOn can be found at https://bitbucket.org/uamsdbmi/dron. The permanent URL for DrOn is http://purl.obolibrary.org/obo/dron.owl.

Our primary, driving use case was support for Comparative Effective Research. Author WRH was part of a research team wherein a student had to manually identify all drug products that contain acetaminophen historically. We built a web application that uses DrOn to support this use case; users can search for all NDCs that either contain a specific ingredient or contain an ingredient that realizes a specific disposition. This web application is accessible at http://ingarden.uams.edu/ingredients.

Future work includes addressing limitations in the current process. One of the more egregious examples is the lack of a link from the various drug products to their dose forms (*e.g.,* drug capsule). Nearly all of the most common dose forms are already in the upper level of the ontology (**dron-upper** module), but the CDFs are not properly related to them. This is due to (1) time constraints and (2) the dubious ontological nature of some of the dose forms found in RxNorm. For example, 'inhaler' does not refer to the form of the drug but instead to its container (which also serves the role of drug delivery device). But the form of the drug itself is a solution or suspension contained in the inhaler. Note that the presentation form in this case (e.g., solution) differs from the administration form (e.g., aerosol).

Another issue is the lack of a full logical definition for some of the terms. For instance, only a small subset of the parts of each drug product is defined. A *clinical drug form* contains information about its dose form, its route of administration, and its active ingredients. As of the writing of this paper, the only one of these that is represented in the ontology as classes are the active ingredients, though dose forms are mostly represented. Even these, however, are still not fully developed, generally lacking any class restrictions. Additionally, a *clinical drug* contains dosage information and *branded drugs* have brand information. Neither of these is represented in the ontology.

The final issue with the process is the need for manual interaction. Although each step within in the process is automated, they are not tied together in a coherent way. We expect that some manual intervention will always be needed as we continue to mine updated information from these sources, but there is significant room for improvement in connecting the various segments of the overall process flow and fully automating the less ontologically nebulous steps.

Since DrOn is already large, and will likely increase in size as we incorporate more sources and as more drug products are manufactured, we expect that we will run into difficulties managing generation of and reasoning over the ontology. One potential solution we intend to investigate is to reason over the modules individually and combine the results. We also intend to create more manageable subsets of DrOn, which should allow users to work with only the portions of DrOn that they need for a particular use case.

ACKNOWLEDGEMENTS

This work was supported by award number UL1TR000039 from the National Center for Advancing Translational Sciences, award R01GM101151 from the National Institute for General Medical Science, and the Arkansas Biosciences Institute, the major research component of the Arkansas Tobacco Settlement Proceeds Act of 2000. This paper does not represent the views of NCATS, NIGMS, or NIH.

REFERENCES

Nelson, S.J., Zeng, K., Kilbourn, J., Powell, T., &Moore, R. (2011). Normalizied names for clinical drugs: RxNorm at 6 years. Journal of the American Medical Informatics Association: JAMIA, 18(4), 441 - 448

U.S. National Library of Medicine (2011). RxNorm Retrieved April 24, 2013, from http://www.nlm.nih.gov/research/umls/rxnorm/

de Matos, P., Alcántara, R., Dekker, A., Ennis, M., Hastings, J., Haug, K., Spiteri, I., Turner, S., and Steinbeck, C. (2010). Chemical Entities of Biological Interest: an update. *Nucl. Acids Res.*, **38**, D249–D254.

Horridge, M., Bechhofer, S. (2011). The OWL API: A Java API for OWL Ontologies. *Semantic Web Journal*, 2(1), 11 – 21

Odersky, M., Altherr, P., Cremet, V., Dragos, I., Dubochet, G., Emir, B., McDirmid, S., Micheloud, S., Mihaylov, N., Schinz, M., Stenman, E., Spoon, L., Zenger, M. (2004). An Overview of the Scala Programming Language, *Technical Report LAMP-REPORT-2006-001*

Typesafe (2013), Slick Retrieved April 24, 2013, from http://slick.typesafe.com/

Broverman, C., Kapusnik-Uner, J., Shalaby, J., & Sperzel, D. (1998). A concept-based medication vocabulary: an essential requirement for pharmacy decision support. Pharmacy practice management quarterly, 18(1), 1-20.

Palchuk, M. B., Klumpenaar, M., Jatkar, T., Zottola, R. J., Adams, W. G., & Abend, A. H. (2010). Enabling Hierarchical View of RxNorm with NDF-RT Drug Classes. AMIA Annual Symposium proceedings, 2010, 577-581.

Parrish, F., Do, N., Bouhaddou, O., & Warnekar, P. (2006). Implementation of RxNorm as a Terminology Mediation Standard for Exchanging Pharmacy Medication between Federal Agencies. AMIA Annu Symp Proc, 1057.

Olsen, L., Grossman, C., & McGinnis, J. M. (2011). Learning What Works: Infrastructure Required for Comparative Effectiveness Research: Workshop Summary: The National Academies Press.

Sperzel, W. D., Broverman, C. A., Kapusnik-Uner, J. E., & Schlesinger, J. M. (1998). The need for a concept-based medication vocabulary as an enabling infrastructure in health informatics. Proceedings AMIA Annual Symposium. 865-869.

Kim, J. M., & Frosdick, P. (2001). Description of a drug hierarchy in a concept-based reference terminology. Proc AMIA Symp, 314-318.

Hogan, W. R., Hanna, J., Joseph, E., Brochhausen, M. (2013). Towards a Consistent and Scientifically Accurate Drug Ontology. ICBO 2013 Conference Proceedings