TRANSFoRm

Translational Research and Patient Safety in Europe

The TRANSFoRm Project is partially funded by the European Commission under the 7th Framework Programme

Grant Agreement Number FP7-247787

D6.3
Data Integration Models

Work Package Number:	6
Work Package Title:	Models: Core ICT Concepts for system and data integration
Nature of Deliverable:	Other
Dissemination Level:	Public
Version:	1.0
Delivery Date From Annex 1:	M36

Principal Authors:	Ethier JF, McGilchrist MM
Contributing Authors:	Burgun A, Sullivan FS
Partner Institutions:	University of Dundee INSERM/UR1

7th Framework Programme http://cordis.europa.eu/fp7/ict/
European Commission http://ec.europa.eu/information_society/index_en.htm

1. Table of contents

List of Tables

List of Figures

2. Executive Summary

This deliverable describes a set of three models, which in conjunction with the semantic mediator (WT7.1) enables the execution of queries formulated through the eligibility representation of the Clinical research information model (WT6.4). An ontology-driven mechanism was developed to enable linkage and integration of phenotypic and genotypic data from multiple distributed data sources. It makes use of the Clinical Data Integration Model (CDIM, WT6.5), the Data Source Model (DSM, WT6.6) and the CDIM-DSM mapping model (WT6.6). Queries formulated through the CDIM and vocabulary service (WT7.2) are translated to local queries by the mediator using the individual source instances of the DSM and CDIM-DSM models.

CDIM is a global mediation model expressed as an ontology for use in the primary care domain. It uses a realist approach employing Basic Formal Ontology (BFO v1.1) as an upper ontology. Other ontologies were imported or specialized to give deeper definition to the concepts in the domain. These included OGMS, IAO and VSO. The CDIM ontology includes concepts that are especially important to primary care (e.g. episode of care or reason for encounter), but also others to handle temporality in queries (e.g. the start and beginning of processes).

The specific requirements for primary care data were first gathered through discussions with experts in the field in order to get a broad view of the domain. The resulting ontology was fine-tuned using two TRANSFoRm use cases covering RCTs and epidemiology (WT1.1, D1.1). Existing models from other projects were also investigated for their usefulness, but these did not satisfy TRANSFoRm's needs in regard to approaches to interoperability (unifying goal), modelling (ontology), domain of interest and content (primary care research).

Following on from a detailed survey and characterisation in year-1 of EHRs and data repositories (WT6.1, WT6.2), three data sources were selected for further investigation for their modes of access (such as SQL and HL7 messaging), their data

model and content to better understand the issue and degree of heterogeneity. These sources included EHR and routine clinical data and research genetic data. From this understanding a data source model (DSM) was developed covering structural aspects of data representation and at all levels of granularity. The model supports dynamic structure, which is a common feature in databases of clinical data (where the structure of data held in a given element is dependent on the content of another element).

Using the DSM, the two clinical repositories and one genetic repository were instantiated and a CDIM-DSM mapping model was developed to align the database structures to the concepts of the ontology. It is this mapping model that identifies the semantics in the source data structures. For the chosen data sources and use-cases a preliminary instantiation of the mapping models proved to be straightforward.

Guidance was developed to assist those charged with writing the TRANSFoRm semantic mediator and to illustrate by specific examples from the use-cases how the three models and terminologies are used to translate eligibility queries to local database queries for execution.

3. Introduction

In this report we describe the infrastructure and supporting models for the integration of heterogeneous data from both clinical and research data sources such as genetic sources. Specifically, we describe the development of three models which drive the infrastructure components, as laid out in DoW tasks 6.6 and 6.5. The architecture for this infrastructure and the position of the models within it is shown in figure 3.1.

In the field of biomedical data interoperability, multiple approaches are proposed in the literature such as data warehousing, data federation, and data mediation. In all cases, a key challenge is that structural and terminological aspects are often dealt with separately, yet are interdependent. We chose to address this challenge as part of our work. A major feature of the TRANSFoRm approach to data interoperability is the provisioning of terminological and structural resources in a uniform way through LexEVS v6 [1].

The workbench (and eCRF) uses LexEVS services to help formulate a query. The query formulator (WT5.3) uses the eligibility representation of the Clinical Research information Model (CRIM, WT6.4), terminology searches using the vocabulary service (VS, WT7.2), and concept retrieval from the Clinical Data Integration Model (CDIM, WT6.5) to generate an executable representation of the formulated queries and uses the distributed middleware (WT 7.5) to invoke the executable queries over the linked repositories or EHRs.

Section 4 presents an article published in JAMIA by the current authors addressing this key challenge. It illustrates our general approach to data interoperability: a mediation system based on the local-as-view paradigm. The mediation approach mandates that a general model be related to specific source models in order to achieve data mediation. Developing our general model as an ontology (CDIM, WT 6.5) allowed us to address this challenge, and also brought other operational advantages. Section 5 presents an article shortly to be submitted to Methods of Information in Medicine where choices regarding the CDIM are discussed. Existing projects using general models for database integration were identified and

characterised to assess their usefulness to the TRANSFoRm project, which itself focuses on integration of data from routine primary care activity and research, whether epidemiology, trials or genomics studies. Seven projects (I2B2, ePCRN, BIRN, ACGT, epSOS, caBIG, FURTHeR) were identified and features relevant to TRANSFoRm were reviewed for each, but none satisfied our requirements.

At the data nodes, the semantic mediator (SM, WT7.1) uses LexEVS services to translate the query to the required local representation for execution. To drive this translation the mediator uses a model of the data source (DSM, WT 6.6) and a model of its mapping to the CDIM (CDIM-DSM, WT 6.6), along with mappings from the query terminologies to the local terminologies (VS, WT 7.2).

The DSM describes the internal structure of data sources such as RDBMS, XML documents and HL7 messages. It is designed to be a target for mapping CDIM concepts to structural elements within the sources and is used by the mediator for query translation. In section 6 we review the underlying structure for RDBMS, XML documents and HL7 message formats and generate a very simple UML model that can represent all of these. A range of existing data sources were investigated for their data models and user-defined data types covering both clinical data repositories and genetic data sources. This was done through a review of specifications of the data models and also with the assistance of staff familiar with the data sources. Models were instantiated for the data sources at NIVEL, CPRD and Biomina. The instantiated source models are represented as XML files, and they were generated semi-automatically using a tool for the purpose. These model files are deployed via the LexEVS 6 platform using a custom loader. The data source model for NIVEL (NPCD DSM v1.0.xml) can be found on the project wiki at:
https://transform.kcl.ac.uk/groups/models/wiki/c1d01/Models_for_D64.html
(Contact info@transformproject.eu to access these files.)

The CDIM-DSM mapping model expresses how concepts from the CDIM are related to one or more structural elements of the DSM using a variety of operators. Operators can be assembled in parallel or in series and support numerical and string operations as well as built in functions. Knowledgeable staff at the data sources must

comprehend the meaning of a CDIM concept and establish the best match to the structural elements in the data source model and combine these in the appropriate way. The concepts of the CDIM are expressed independently of the data sources so such mappings are not guaranteed to exist. In section 7, mappings were instantiated for the data sources at NIVEL, GPRD and Biomina. The mapping models are represented as XML files, and were generated semi-automatically using a tool for this purpose and are deployed using the LexEVS 6 platform using a custom loader. The CDIM-DSM mapping model for NIVEL (CDIM-NPCD v1.0.xml) can be found on the project wiki at:

https://transform.kcl.ac.uk/groups/models/wiki/c1d01/Models_for_D64.html

(Contact info@transformproject.eu to access these files.)

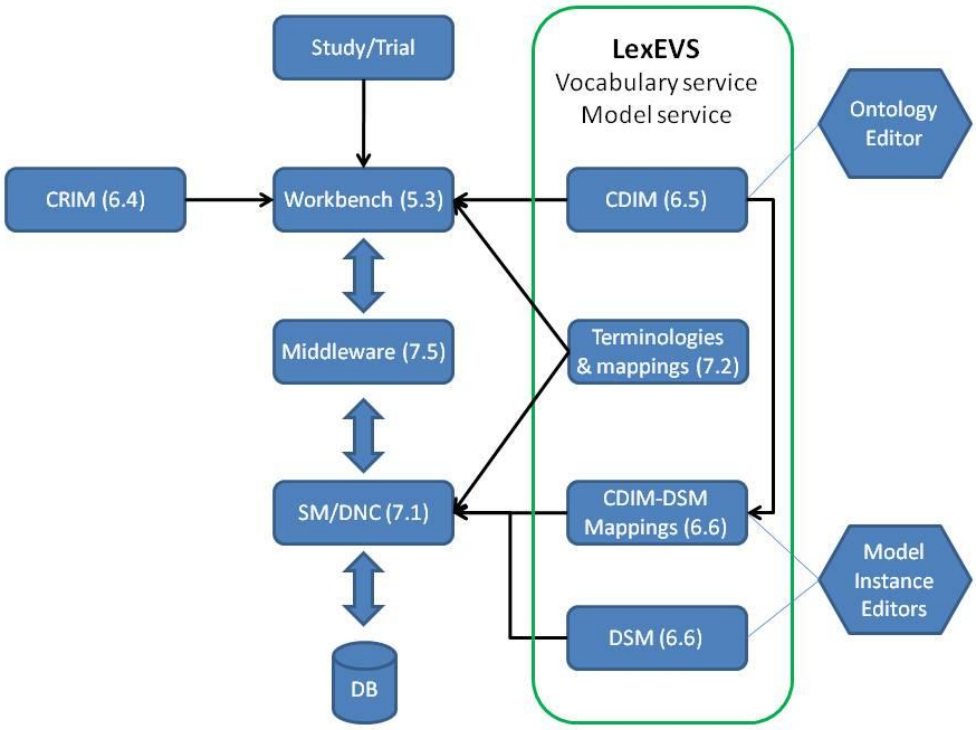


Figure 3.1: TRANSFoRm architecture and supporting models.

The workbench (WT5.3) draws on the eligibility representation of the CRIM (Clinical Research Information Model, WT6.4), vocabulary service (WT7.2) and the CDIM (Clinical Data Integration Model, WT6.5) to formulate queries. These queries are distributed to the individual data nodes via the middleware (WT7.5). The data node console (DNC) authorises the semantic mediator (SM, WT7.1) to translate the queries to local equivalents (SQL, HL7 messages) using the DSM (Data Source Model, WT6.6), the CDIM-DSM mappings (WT6.6), and the terminology mappings (WT7.2) before authorising its execution. All models, terminologies and mappings are stored and manipulated within an instance of LexEVS [1].

4. A unified structural/terminological interoperability framework based on LexEVS

Note: This chapter is reproduced from **Ethier J-F, et al. J Am Med Inform Assoc 2013;0:1–9**. doi:10.1136/amiajnl-2012-001312. All figures and tables are numbered relative to this chapter. All references can be found at the end of this chapter.

4.1. INTRODUCTION

Biomedical research increasingly relies on the integration of information from multiple data sources, obtained either primarily for the purposes of research, such as trial data and genetic samples, or through secondary use of routinely collected data, e.g. electronic health records (EHRs). However, the heterogeneity of these data sources represents a major challenge to the research task.[1–3] Two levels of heterogeneity can be distinguished: structural and terminological. Firstly, *Information models* are used to represent the organization of data structures in information systems.[4–6] Variation in their forms and approaches generates *structural heterogeneity* of the data models. Secondly, numerous medical coding systems (terminologies) are used to represent diagnoses, procedures, and treatments in health databases,[7] frequently with many-to-many mappings between them, creating *semantic heterogeneity,* sometimes also referred to as *terminological heterogeneity*.[8]

Alan Rector mentions that these two types of heterogeneity, structural and semantic, are not independent as there are mutual constraints between the information models and coding systems.[9] This interdependence corresponds to what Rector calls the “binding” between an information model and a coding system, and presents a notorious source of ambiguity in clinical systems.[4] At the time of coding, implicit knowledge is sometimes used but not formally represented in the information model. Some models function under the closed world assumption, whereby omission implies falsehood, while others support the open world assumption in which omission merely states that the information is not available. Further complexity is caused by

differences in granularity, depth, coverage and composition (single term versus expressions) between models.

This article proposes a unified framework for integration of heterogeneous information models and terminologies to construct a single solution for structural and semantic interoperability. This approach is currently being adopted in TRANSFoRm, an EU FP7 project that aims to comprehensively support the integration of clinical and translational research data in the primary care domain.[10,11]

4.2. BACKGROUND AND SIGNIFICANCE

Structural and semantic interoperability in biomedical data have been explored in a number of initiatives. Given our interest in translational medicine and data reusability, we focus here on those allowing federated queries from multiple clinical repositories and EHRs.

There have been attempts to create generic information models to serve as standards, including the OpenEHR reference model, the Informatics for Integrating Biology and the Bedside (i2b2) model, the HL7 Reference Information Model (RIM) and the Clinical Data Acquisition Standards Harmonization (CDASH).[12–15] An ongoing international collaboration between standards organizations and industry partners, the Clinical Information Modeling Initiative (CIMI), aims at bringing together a variety of approaches to clinical data modeling (HL7 templates, openEHR archetypes, etc.) as a series of underlying reference models.[16] A similar endeavor is ongoing with the Biomedical Research Integrated Domain Group (BRIDG) in the research area.[17] Nevertheless, many existing data sources are not designed according to these initiatives.

Approaches to structural heterogeneity can be grouped in two categories: Extract-Transform-Load (ETL) systems and mediators systems. In the former, the different data sources to be integrated (e.g. data warehouses) are all expected to conform to some structural model. This is achieved by carrying out an Extract-Transform-Load process on an existing relational database to transfer the data into a single target

model. Multiple projects have been built on this approach. The Shared Health Research Information Network (SHRINE) aims at bringing together various “Informatics for Integrating Biology and the Bedside” (i2b2) clinical data repositories.[13,18,19] The i2b2 model is also used by other projects like TRANSMART.[20] The Stanford Translational Research Integrated Database Environment (STRIDE), an initiative from Stanford, uses the HL7 RIM as a foundation for their model while EU-ADR developed its own common model.[21,22] Finally, the electronic Primary Care Research Network (ePCRN) project, focusing on the primary care domain, based its structure on the American Society for Testing and Materials (ASTM) Continuity of Care Record (CCR) information model.[23,24]

Other systems use a mediator approach to address structural heterogeneity. Some central schema is mapped to the local schemas of individual data sources, which retain their original structure. These central schemas were initially described as ontologies.[25] Projects like Advancing Clinico-Genomic Trials (ACGT) in the cancer domain leveraged this approach.[26] Other projects implemented mediators in different ways. The Biomedical Informatics Research Network (BIRN) and its follow-up initiative the Neuroscience Information Framework (NIF) are using an XML approach.[27–29] The caBIG (cancer Biomedical Informatics Grid) is a longstanding National Cancer Institute (NCI) driven initiative to federate healthcare data with sources represented as Unified Modeling Language (UML) models. [30–32] A similar modeling approach is used by the Federated Utah Research and Translational Health eRepository (FURTHeR) and Electronic Health Record for Clinical Research (EHR4CR).[33,34] None of these implementations use vocabulary services to support their structural aspects.

The terminological needs of various projects are handled internally. The SHRINE project uses a pivot terminology and BIRN stores term mappings in a relational database.[35,36] The Smart Open Services for European Patients (epSOS) project is developing an ontology to address the multilingual and mapping needs of its community.[37,38] Nevertheless, terminology servers are often involved like Apelon DTS in FURTHeR and Bioportal in ONCO-I2B2.[39,40]

The LexEVS terminology server, having originally been developed in the context of the caBIG initiative, is being used by several projects (e.g. ePCRN, National Cancer Institute Thesaurus browser).[24,41,42] The web-based server Bioportal also uses it as part of its infrastructure.[43] LexEVS permits unification of all loaded terminologies under the LexGrid format (including ontologies expressed as Ontology Web Language - OWL).[44] It allows a range of deployment options, from a local installation to a grid service, and is available under an open source license. Version 6 of LexEVS implements the HL7 Common Terminology Services 2 (CTS 2) Service Functional Model (SFM) – although it does not conform to the HL7 CTS 2 OMG specification since the specification was finalized after version 6 was released.[45,46] Prior to our efforts, LexEVS implementations have mostly been used to support terminological information

Binding between information models and terminologies presents a challenge in its own right. A number of projects mentioned above have developed their own solutions, nevertheless, standards for metadata registries have been created to address this question (e.g. ISO 11179).[47] Projects such as eMERGE and caBIG use the cancer Data Standard Repository (caDSR).[48] It stores data elements described by a definition of what is represented as well as the list of valid values. caBIG binds its UML models with the terminologies through use of these data elements. eMERGE also uses the caDSR to harmonize local genotype and phenotype data elements. The binding of structure and terminology has also been addressed in the context of HL7 with the TermInfo initiative currently focusing on the use of SNOMED CT in HL7 V3.[49]

All of these projects consider structural and semantic aspects of interoperability to be distinct, leading them to be managed separately, although the separation between structure and terminology is drawn differently in different projects. Recognizing their dependencies and that terminological and structural operations share a common set of requirements (through binding and mappings), we hypothesized that a unified ontology-based knowledge framework can facilitate interoperability between heterogeneous sources, without having to create a separation and different tools for

management. Based on our analysis of terminological solutions, we investigated if LexEVS was a functional tool to implement this approach.

In the next section, we present the framework and describe the generic approach for each of its components. We then test this method on a clinical study example from the TRANSFoRm project, focusing on integrating two primary care data repositories, the NIVEL Primary Care Database (NPCD)[50] of the Netherlands institute for health services research (NIVEL)[51] and the General Practice Research Database (GPRD)[52] of the United Kingdom's Medicines and Healthcare products Regulatory Agency (MHRA).[53]

4.3. MATERIAL AND METHODS

The main aims of our work are to simplify the handling of heterogeneous data sources for the users and to minimize the interoperability implementation workload for the data sources. We believe the mediation paradigm best meets these goals.[25] Instead of using ETL to enforce a uniform information model, our framework uses mappings to relate local models to a general information model (GIM). This also facilitates user operations as they only need to interact with the general model and do not need to be familiar with each data source's information model.

The mediation framework has been constructed according to the local-as-view (LAV) principle.[54] In this approach, each source schema is defined as a set of views on the global schema, as opposed to the global-as-view principle where the global schema is defined in terms of the sources. So the GIM does not have to be derived directly from any source. Rather, it should be built to construct a sound and logical view of the domain of interest in order to make sure all required concepts are present. This ensures scalability as adding a new source does not necessitate a modification of the GIM. It also presents a more stable model to the user.

In our framework, GIM is represented as an ontology, allowing it to be stored in the LexEVS terminology server together with the data sources models (DSM) and the terminologies. Mappings between GIM and data sources (DSmG) can then be uniformly created, stored and leveraged as described below. In parallel, similar methods can be used to handle terminological operations.

4.3.1. Architecture Overview

The modelling infrastructure resides entirely within a terminology server, enabling unification of structural and semantic modelling and operations within this server. Several types of models are present:

1. The general information model (GIM)
2. Models describing each data source (DSM)
3. Mapping sets between the sources and the GIM - one set per source (DSmG)
4. Terminologies used to code the data elements (e.g. International Classification of Diseases – ICD - 10 codes…)
5. Mappings between terminologies (TmT)

An overview of how the different models interact together is presented in Figure 4.1, which shows a user query being sent to mulitple data sources. Security and other administrative issues have been intentionally left out of this list in order to focus on the relevant steps for this demonstration.

1. The query is expressed using GIM concepts
2. The mediation engine generates a specific query for each data source
3. The data sources fulfill the requests
4. The returned dataset has its structure aligned with the GIM
    a. Data Source Models to extract which terminology was used to code a given concept in the source
5. If possible and desired, the system can semantically align resulting coded values based on the terminologies used by one of the sources or a separate terminology. This operation uses:
    a. Terminologies and mappings between terminologies to transcode the values

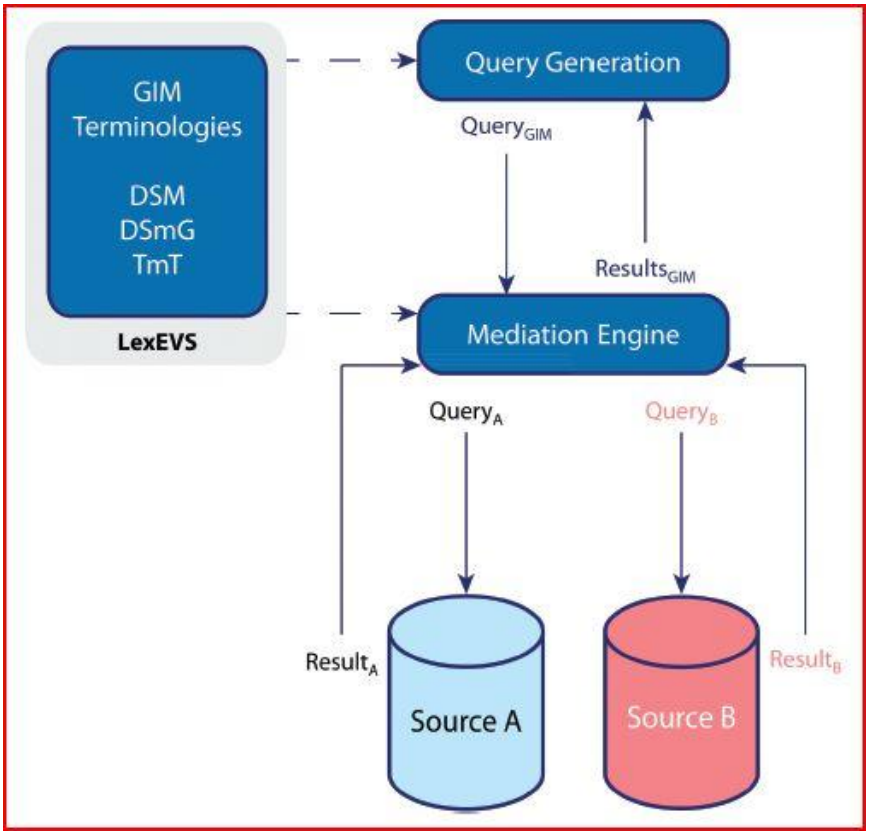


Figure 4.1: Architecture supporting model interactions based on LexEVS for query meditation-based query resolution.

4.3.2. General Information Model (GIM)

The General Information Model is used to represent a unified view of the domain concepts and their relationships. For example, date of birth, diagnosis and patient are all relevant concepts in a clinical care context. Each concept also has intrinsic properties. Given the data integration function of the ontology and its role as a mediation schema, we chose a realist approach using BFO 1.1 as the foundation of the model.[55,56] The implementation of BFO as a formal, description logics ontology allows easier interaction with projects using semantic web technologies (like epSOS), or other parts of projects implementing the framework. For example, the provenance service and the decision support service from TRANSFoRm both rely on ontologies and will need to interact closely with the unified integration framework.

Figure 4.2 illustrates how “gender” and its relevant attributes represented in GIM are rendered once loaded in LexEVS.

The “codedWith” properties of the concept supports binding between the information model and the relevant terminology (or value set) and contributes to its semantics representation. In this case, it indicates that values for this concept are to be represented with the terminology named “gim_gender” stored in LexEVS. Multilingual capabilities are handled natively within LexEVS by combining property values with a language descriptor. When a translation is provided, this allows the model to also propose a multilingual solution without resorting to another system.

Figure 4.2: GIM - partial representation of "gender" attributes in LexEVS.

4.3.3. Data Source Models (DSM)

A new DSM is defined for every data source to be supported. The goal of this stage is to provide enough information to the system in order to translate a query based on the GIM into the local language used to query the source. The exact nature of the properties and relations will be related to the underlying type of source to be modeled.

For example, a SQL data source “SA” would have hierarchical relations such as hasTable and hasField with other relations representing the relations between the tables (oneToMany, OneToOne…) with the keys on each side. Another data source “SB” could be an XML document, with XPath as its query language. A model fulfilling the same goal can be created describing nodes, elements and attributes.

A DSM fragment is illustrated in Figure 4.3, representing a field. In terms of concept properties, we have some similarities with the GIM but also specific properties for a SQL source concept.

The objectType property gives the nature of the concept (field) while the name of the object is in the description. Multiple textual presentations (here Dutch and English) can be created to provide translations in order to facilitate the use of the information in multiple contexts. As with the GIM, “codedWith” properties hold the name and versions of the terminology (or local value set) used to code data for this concept (a field in this example). Note that this does not need to be the same terminology in all DSMs and GIM. This allows a DSM to register the specific terminology (or value set) used to code the information locally, irrespective of what is registered with GIM.

Figure 4.3: DSM - partial representation in LexEVS of a field named "GESLACHT" from a sources SQL database.

4.3.4. Mappings between a source and the GIM (DSmG)

A mapping set does not need to duplicate the concepts from the model but simply reference them via their code and coding scheme name. A relation is then created for each correspondence between a GIM concept and a DSM concept.

We developed a generic mapping model defining data transformation operations to align source data values with the GIM, supporting not only one-to-one mappings but also more complex cases. One-to-one operations include simple mappings such as

a date corresponding to a date/time value, while a more complex case would consist of two distinct but related fields. For example, a symptom (a code from a terminology) can possibly denote multiple entity types (in GIM). For example, “abdominal pain” can be used to code a “presenting complaint”, a “symptom” or even sometimes a “final diagnosis” if no clear diagnosis emerges during the consultation. Some data sources, instead of having three fields representing the three possible entity types, will have two fields: one storing the actual symptom code and one for the entity type. For example, field A would store the value “abdominal pain”, while field B would store the entity type “presenting complaint” in the same record, to distinguish it from someone with a diagnosis of abdominal pain as part of their medical history.

In this case, instead of linking directly from the source to the GIM, an intermediate concept is created in the mapping set. This intermediate concept will hold the condition for this relation to be true. So, if our example maps to some concept AP154 in GIM, the mapping would proceed as *Field A → Condition 1 (Field B=”Value 1”) →GIM AP154, i.e.* Field A represents GIM concept AP154 only if Field B = “Value 1”. Intermediate concepts can also be chained in order to combine different operations.

The model also supports the creation of a virtual element to capture implicit knowledge. For example, it could represent a laboratory unit which might not be physically present in the data source because it is always the same in the context of that source. Similarly, the mapping model can support yes/no fields (e.g. a column denoting presence or absence of diabetes) which combines both the structural and terminological elements.

4.3.5. Terminologies

The Unified Medical Language System (UMLS) presents a unified view of a large number of relevant biomedical terminologies.[57] It includes over 2 million concepts from various vocabularies and millions of relationships. By using concept unique identifiers (CUI) – used to relate codes in different terminologies but with a similar meaning – and semantic groups, it facilitates terminology alignment. The UMLS can be loaded directly in LexEVS 6 which supports all its features.

Additional LexEVS loaders are easily created to load terminologies that are not yet supported. This was exemplified by the creation of a loader for the Anatomical Therapeutic Chemical classification system (ATC 2011) in collaboration with the LexEVS developers.

4.3.6. Mappings between terminologies (TmT)

Once terminologies are loaded in LexEVS, mappings between them can be created in a similar way as for the data models. For some of them, relationships are readily available and can be simply loaded into LexEVS. This is typically the case for terminologies integrated in the UMLS.

For others, local mappings have to be created. For example, if a hospital uses a local coding set to identify its laboratory tests, it could be loaded into LexEVS. Subsequently, mappings between this local set and LOINC could be created. This would allow translations from the local site to a more standard terminology, thereby facilitating interoperability with other groups without having to recode data locally or create a duplicate data warehouse.

When more than two terminologies are used, mapping sets can be created between each of them or only to some selected central (pivot) terminology, which then acts as a hub for translating concepts. A pivot terminology is optional in the GIM framework and left for the users to decide on. In the absence of a designated terminology, the user can choose one of the terminologies supported in the selected sources to which the others will attempt to map.

4.4. RESULTS

The first implementation of GIM was realized as part of the EU FP7 TRANSFoRm project, which aims at supporting patient safety through integration of clinical and research settings, workflows and data.[11] The technology developed can facilitate the interactions with individual EHR systems for trial recruitment and follow-up, as well as diagnostic support. The TRANSFoRm project also relies on a workbench to explore clinical and research data repositories. To achieve this, significant

challenges need to be overcome in the areas of interoperability and methods for data integration.

4.4.1. Clinical Data Integration Model: GIM instantiation in TRANSFoRm

The Clinical Data Integration Model (CDIM) is the GIM instantiation in TRANSFoRm, and covers concepts relevant to data integration in primary care research like medication, diagnosis, and laboratory tests. It is implemented as an OWL ontology based on the Basic Formal Ontology (BFO) 1.1.[56] It imports the General Medical Science (OGMS),[58] the Vital Sign Ontology (VSO)[59] and the Information Artifact Ontology (IAO).[60] The ontology also integrates concepts from existing ontologies such as the Ontology for Biomedical Investigations (OBI),[61] the Gene Ontology (GO)[62] and the Translational Medicine Ontology[63] when possible.

The resulting ontology has 457 classes (102 unique to CDIM) and 73 object properties (1 sub-property unique to CDIM). Twenty-one novel CDIM classes had to be introduced to represent and manage temporal aspects necessary in TRANSFoRm. All required concepts, as defined by use cases, could be modeled in CDIM. Figure 4.4 presents a subset of CDIM adapted to illustrate a subset of queries related to the diagnosis of diabetes.

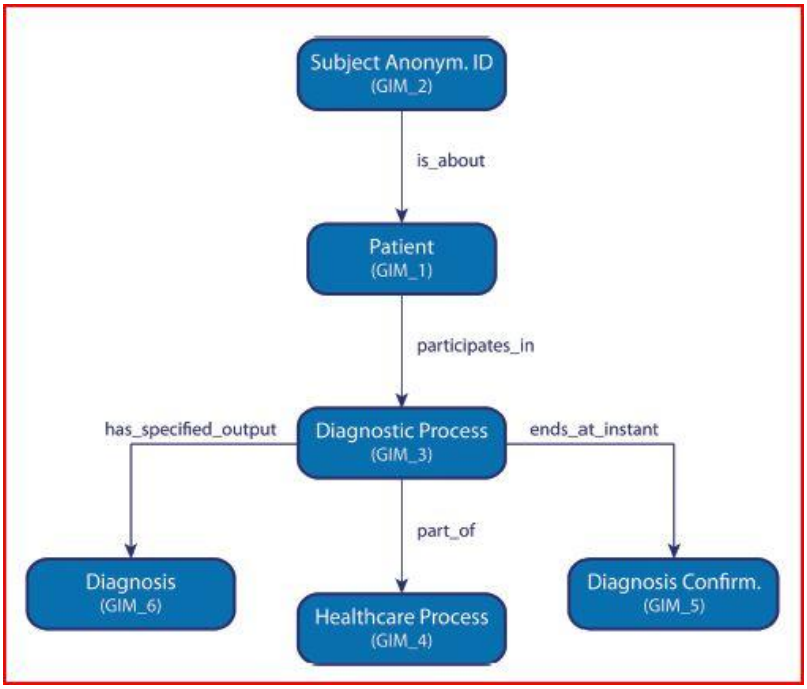


Figure 4.4: CDIM subset focused on diagnosis. Identifies are in parentheses.

4.4.2. Instantiation of structural models, terminologies and mappings

Two clinical data repositories were used to evaluate the suitability of the framework for the project: NPCD from the Netherlands and GPRD from United Kingdom. Both their structures and the terminologies used to code information are different. For example, medication is coded with the British National Formulary (BNF) codes in GPRD but the ATC classification is used in NPCD, with diagnoses coded with Read Codes version 2 in GPRD and ICPC version 1 in NPCD.

Structural models in XML were created for both sources using a semi-automated tool and then loaded into LexEVS. The NPCD database extract we used contained 60521 anonymised patient records whereas the GPRD extract made available for the project contained 5000 patient entries. Eight tables (181 fields) in NPCD and ten tables (107 fields) in GPRD were considered in the structural models.

CDIM was mapped with 44 elements in NPCD and 47 in GPRD. High level classes like “processual entity” are part of CDIM and are essential to knowledge modeling but are not expected to be used as mapping targets as they are too generic. Twenty-nine mappings (32%) were one-to-one direct relations between CDIM concepts and a data source structural element. The other mappings included concatenation operations and conditional mappings (including related tables). No virtual elements were necessary for the current data source mappings. Figure 4.5 illustrates an example of a conditional mapping. Precise and comprehensive knowledge of each data source and its real-life usage was essential to achieve satisfactory mappings and query results. Not all fields of the data sources are targets for mappings, nor are all concepts in CDIM mapped to each data sources; their coverage typically differs from CDIM. Nevertheless, all the relevant entities for the use cases were successfully mapped. Figure 4.5 presents those mappings necessary to illustrate the examples in Figure 4.6.

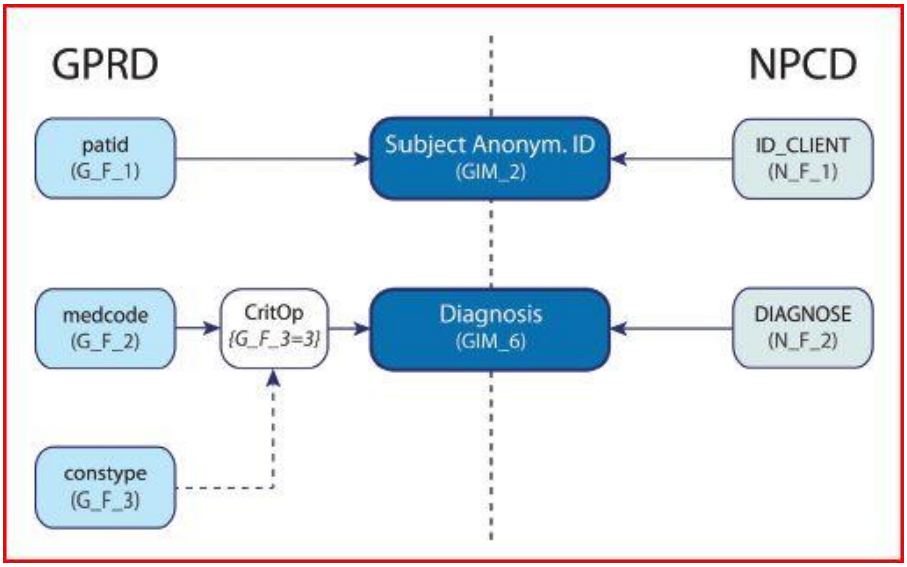


Figure 4.5: Mapping examples between GIM and DSMs (GPRD and NPCD). Identifiers are in parentheses.

Based on our use case and available data sources, we focused on ICD 9 and 10 codes, International Classification of Primary Care version 1 codes, Read Codes version 2 for diagnoses, the Anatomical Therapeutic Chemical classification (ATC), the British National Formulary (BNF) for drugs, as well as on Logical Observation Identifiers Names and Codes (LOINC) for laboratory tests.

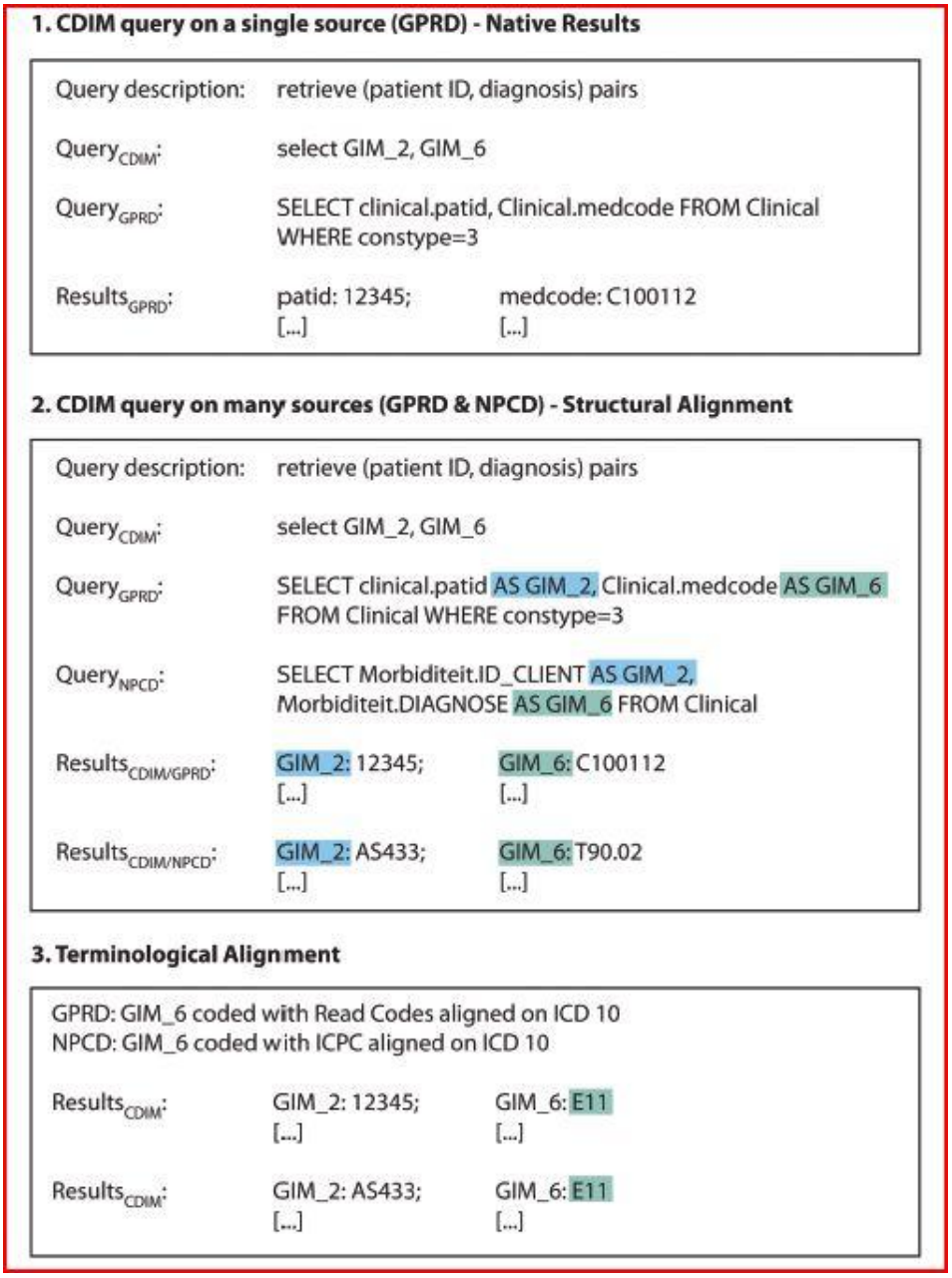


Figure 4.6: Examples of query resolutions as applied to TRANSFoRm using CDIM (Figure 4.4), its mappings to the DSMs (Figure 4.5) and terminologies. Highlighted segments represent each level-specific addition based on information from the models served by LexEVS.

4.4.3. Evaluation

We evaluated the applicability of the GIM approach to TRANSFoRm's clinical trial use cases. We focused on the retrospective diabetes cohort study.[64] This use case aims at identifying eventual associations between single nucleotide polymorphism (SNPs) and diabetes complications or responses to oral antidiabetic drugs. Twenty-

six relevant queries were identified and were all successfully implemented, in conjunction with appropriate terminological values. For example:

- Patients >= 35 years old
  AND
  ((with a diagnosis of diabetes accompanying a prescription or an episode of care)
  OR (taking metformin OR a sulfonylurea medication in last 5 years)
  OR (having a laboratory test of glycosylated hemoglobin > 6.5%
  OR a random glucose > 11.0 mmol/L
  OR a fasting glucose > 7.0 mmol/L))

Figure 4.6 demonstrates different features of the LexEVS implementation of the framework. The first example illustrates how to create the local source query based on information contained within CDIM and the data source model (DSM). The latter would contain field and table relations required to derive the SQL statement. By utilizing the mappings shown in Figure 4.5, the query is translated in the local source query format.

Similar principles can be applied for multiple sources but as shown in the first example of Figure 4.6, the resulting dataset structure is based on the local source. In the example, it is not clear that “DIAGNOSE” and “medcode” carry a similar meaning, especially since this equivalence is only true if a condition on the field “constype” is applied. By adjusting the local query to maintain a reference to CDIM, the resulting datasets from two data sources (NIVEL and GPRD) can be assembled in a coherent structure as in example 2.

Although both result sets now share an identical structure, the terminologies used to code the information are different. In some situations, alignment might not even be possible, at least not in a completely automated fashion as with ATC and BNF for medication types. In this diabetes example, we consider the “coded with” properties in the local DSMs, as previously described. For GPRD, “Non-insulin dependent diabetes mellitus” in Read Codes version 2 can be related to an ICD 10 code (E11) by following mappings in LexEVS. The same can be done for NIVEL with ICPC-1 code T90.02 to ICD 10 code E11. The final unified dataset is homogenous and consistent semantically as in example 3.

4.5. DISCUSSION

Achieving interoperability between health data sources such as EHRs and registries is a challenging but crucial endeavor for both designers and users of health IT. The structural and terminological aspects of data source interoperability, while intrinsically linked, have traditionally been handled separately.[65,66] From a structural perspective, a number of projects have adopted a common model to which each source is expected to comply, whether when inputting data (e.g. CDASH in the clinical research domain)[15] or when data is being extracted (e.g. EU-ADR focusing on adverse event analysis).[22]

Other projects have opted for a mediation approach, with a centralized knowledge model, often represented as an ontology. XML and UML designs are also possibilities, as utilized in the BIRN and FURTHeR projects, respectively. Our framework is built around GIM as the central knowledge model, expressed as an ontology with a realist approach based on BFO 1.1.

The semantic challenges are addressed either through dedicated project-specific tools or through terminological servers, such as the one used in the ePCRN project. The GIM framework is novel in that it uses a terminological server not only for handling semantic interoperability, but for structural aspects as well.

Binding both terminological and structural aspects, when they are managed separately, is a challenge that has previously been handled through the use of metadata registries like caDSR, as used in the caBIG and eMERGE projects.[30,67] The registries allow data elements to be created where a definition and a list of permissible values is attached. Our framework avoids this situation by handling the binding in the mediation structure, where both sets of models are located already. It allows data elements present in existing data sources to be described and integrated readily in the context of GIM and allows the use of local code value sets easily as they are stored in the framework.

Our approach represents a step beyond the traditional interoperability paradigm involving a different set of tools for dealing with structural, terminological and binding challenges, in that we present a unified framework that provides an integration solution for these facets inside a single structure. Our LexEVS implementation of GIM, as demonstrated in the TRANSFoRm project, allows a query to be expressed

using clinical concepts from a single generic model that is represented as an ontology, and allows its translation into source-specific queries which then return the results from each source, simplifying and standardizing the interoperability task.

4.5.1. Strengths and Limitations

One of the biggest barriers to the usage of federated data sources is the resource and effort expected from the data sources to participate in a collaborative structure.[3] In order to mend heterogeneity between two data sources, related elements must be mapped to each other. Whether structural models, such as database schemas, or terminologies are to be aligned, the processes share a common subset of requirements.[68] Multiple approaches have been developed to address the issue.[57,69] Our infrastructure does not necessitate *a priori* substantial changes to the structure of the data source. If desired, ETL may be used to transform the initial data schema into a derived schema closer to GIM and this could facilitate the use of direct mappings. If an organization already has a data warehouse, it might be used as-is, thereby reducing integration effort and avoiding data duplication.

The architecture presented decouples the interoperability modeling aspects from the application itself. For some data sources, especially EHRs, exposing the structure of their databases might not be possible or desirable. In this case, an instance of LexEVS can be installed on a local server, allowing query translation to happen at the local level.

From the maintenance perspective, the addition of a new piece of information to a source will necessitate mappings to the relevant GIM terms before becoming usable.[9] Note that our approach can leverage the GIM semantic richness to make this mapping step easier.[70] This occurred with the CDIM implementation of GIM in the TRANSFoRm project, where we use “codedWith” properties to suggest concepts which might share similar semantics. Similarly, distance between concepts in the graph can be used to suggest related concepts. Mappings within TRANSFoRm are currently created manually but should it be expanded, mapping tools will be required in order to support its development. Our LexEVS implementation supports most attributes necessary to allow such work.[70] This has recently been identified as a

core challenge to the field by Shvaiko and Euzenat,[68] and we believe that our approach can contribute to an alignment infrastructure, fostering collaboration.

There are a number of advantages to using LexEVS as the implementation technology. The GIM ontology is stored in the LexEVS terminology server, allowing us to leverage its two optimization axioms: “fully restrict then query” and “lazy loads”. The former minimizes resource requirements by allowing the system to fully restrict any query, including operations on sets (e.g. intersections, unions or differences) before running it against the data source. The latter technique preferentially loads only certain types of information in the first pass while retaining a pointer to dynamically load more information should this be needed. Together, these facilitate efficient query mediation on heterogeneous data sources.

Our approach also benefits directly from LexEVS capabilities for handling versioning. Multiple versions of the models, terminologies and mappings can coexist in the system, and be maintained independently from our framework, removing the need for a separate implementation of versioning. Similarly, multilingual capabilities supported by LexEVS can be used for many operations without resorting to an ancillary tool.

Once loaded and functional, the framework can leverage intrinsic capabilities of LexEVS to create value sets (i.e. subsets of related concepts), which can then be used to handle terminological needs (e.g. codes used to represent drugs to treat diabetes) and manage GIM concept groups. For example, relevant concepts related to laboratory tests can be grouped in order to facilitate searching and browsing. This is different from other efforts where structural models are stored in project-specific structures. Using LexEVS to manage GIM and DSMs automatically provides the methods that implement the HL7 CTS 2 SFM, and ultimately HL7 CTS 2 OMG, ensuring that the implementation remains maintainable and reusable.[71]

The level of automation for query translation and results aggregation depend on the possibility of creating meaningful mappings between relevant terms.[72,73] We showed in our example that mappings between different terminologies can be utilized to fully automate the process for some situations. Nevertheless, some terminology pairs to not lend themselves to such an exercise. These include the ATC and BNF terminologies for therapeutic substances.[74,75] Their approach to classification varies in granularity, depth and coverage, leading for some terms to

one-to-many mappings or absence of related concept. In such scenario, the infrastructure can readily support a user interface where similar, but not necessarily equivalent, terms in different terminologies used by different sources could be suggested, edited and finally approved by the user instead of being automatically chosen.

4.5.2. Applicability

The infrastructure is currently being deployed in the pan-European TRANSFoRm project, with a view to deploying it in other EU and US translational research projects in academia and industry. Specific TRANSFoRm activities that require combined semantic and structural integration include:

- Support for dynamic and persistent linkage between data sources for widely scalable epidemiological studies.
- Support for clinical decision support embedded in the EHR, enabling capture and recording of clinical diagnostic cues in a controlled form.
- Support for real time linkage to a variety of different EHR systems for extraction of clinical data elements into an electronic case report form and write-back of controlled data elements to the EHR to serve as an eSource for regulated clinical trials.

Deploying CDIM as a unified framework in this setting allows the project tools to have full control over the content and structure of queries data sources, and demonstrated its applicability to multiple deployment scenarios, including distributed installations. This study showed that this unified framework, supported by LexEVS, is a suitable platform in which to achieve these tasks in the context of two exemplar databases. The tool chosen in TRANSFoRm was LexEVS. Nevertheless, in a different context, other tools like Bioportal might also have the potential to support the framework.

4.6. CONCLUSION

In this paper, we presented a novel, unifying approach to address interoperability challenges in heterogeneous data sources, by representing structural and semantic models in a single framework. This represents a significant departure from the previous strategies for addressing interoperability in translational research, and it has been successfully demonstrated within the context of the clinical research studies of the EU TRANSFoRm project.

The advantage of this approach is that the systems using the architecture can rely solely on GIM concepts, abstracting over both the structure and coding specificities of the data sources. Information models, terminologies and mappings are all stored in LexEVS and can be accessed using the same methods (implementing the HL7 CTS2 SFM). The system is flexible, and should reduce the integration effort required from the data sources, thereby lowering the cost of entry of this type of research for smaller institutions, and removing the need for larger institutions to invest in additional data warehousing.

4.7. REFERENCES

1. Cimino JJ. Desiderata for controlled medical vocabularies in the twenty-first century. *Methods Inf Med* 1998;**37**:394–403.

2. Cimino JJ. In defense of the Desiderata. *J Biomed Inform* 2006;**39**:299–306.

3. Sujansky W. Heterogeneous Database Integration in Biomedicine. *J Biomed Inform* 2001;**34**:285–98.

4. Rector AL, Qamar R, Marley T. Binding ontologies and coding systems to electronic health records and messages. *Appl Ontol* 2009;**4**:51–69.

5. Eichelberg M, Aden T, Riesmeier J, *et al.* A survey and analysis of Electronic Healthcare Record standards. *ACM Comput Surv* 2005;**37**:277–315.

6. Kalra D. Electronic health record standards. *Yearb Med Inform* 2006:136–44.

7. Bodenreider O. Biomedical ontologies in action: role in knowledge management, data integration and decision support. *Yearb Med Inform* 2008:67–79.

8. Rector AL. Clinical terminology: why is it so hard? *Methods Inf Med* 1999;**38**:239–52.

9. Qamar R, Kola JS, Rector AL. Unambiguous data modeling to ensure higher accuracy term binding to clinical terminologies. *AMIA Annu Symp Proc* 2007;**2007**:608–13.

10. Delaney B. TRANSFoRm: Translational Medicine and Patient Safety in Europe. In: Grossman C, Powers B, McGinnis JM, eds. *Digital Infrastructure for the Learning Health System: The Foundation for Continuous Improvement in Health and Health Care: Workshop Series Summary*. Washington, DC: National Academies Press, 2011:198–202.

11. TRANSFoRm Project. http://www.transformproject.eu (accessed 11 Apr 2012).

12. Beale T, Heard S, Kalra D, *et al.* The openEHR Reference Model - EHR Information Model - Release 1.0.2. http://www.openehr.org/releases/1.0.2 (accessed 29 Jun 2012).

13. Murphy SN, Mendis M, Hackett K, *et al.* Architecture of the Open-source Clinical Research Chart from Informatics for Integrating Biology and the Bedside. *AMIA Annu Symp Proc* 2007:548–52.

14. Schadow G, Mead CN, Walker DM. The HL7 reference information model under scrutiny. *Stud Health Technol Inform* 2006;**124**:151–6.

15. CDASH - Basic Recommended Data Collection Fields for Medical Research. http://www.cdisc.org/cdash (accessed 8 Dec 2012).

16. Clinical Information Modelling Initiative... http://www.openehr.org/326-OE.html?branch=1&language=1 (accessed 8 Dec 2012).

17. Fridsma DB, Evans J, Hastak S, *et al.* The BRIDG Project: A Technical Report. *J Am Med Inform Assoc* 2008;**15**:130–7.

18. Weber GM, Murphy SN, McMurry AJ, *et al.* The Shared Health Research Information Network (SHRINE): A Prototype Federated Query Tool for Clinical Data Repositories. *J Am Med Inform Assoc* 2009;**16**:624–30.

19. Murphy SN, Weber G, Mendis M, *et al.* Serving the Enterprise and Beyond with Informatics for Integrating Biology and the Bedside (i2b2). *J Am Med Inform Assoc* 2010;**17**:124–30.

20. Szalma S, Koka V, Khasanova T, *et al.* Effective knowledge management in translational medicine. *Journal of Translational Medicine* 2010;**8**:68.

21. Lowe HJ, Ferris TA, Hernandez PM, *et al.* STRIDE – An Integrated Standards-Based Translational Research Informatics Platform. *AMIA Annu Symp Proc* 2009;**2009**:391–5.

22. Avillach P, Dufour J-C, Diallo G, *et al.* Design and validation of an automated method to detect known adverse drug reactions in MEDLINE: a contribution from the EU-ADR project. *J Am Med Inform Assoc* 2012

23. Delaney BC, Peterson KA, Speedie S, *et al.* Envisioning a Learning Health Care System: The Electronic Primary Care Research Network, A Case Study. *Ann Fam Med* 2012;**10**:54–9.

24. Peterson KA, Fontaine P, Speedie S. The Electronic Primary Care Research Network (ePCRN): A New Era in Practice-based Research. *J Am Board Fam Med* 2006;**19**:93–7.

25. Wiederhold G. Mediators in the architecture of future information systems. *Computer Journal* 1992;**25**:38 –49.

26. Martin L, Anguita A, Graf N, *et al.* ACGT: advancing clinico-genomic trials on cancer - four years of experience. *Stud Health Technol Inform* 2011;**169**:734–8.

27. Gupta A, Ludascher B, Martone ME. Knowledge-based integration of neuroscience data sources. In: *Scientific and Statistical Database Management, 2000. Proceedings. 12th International Conference on*; 2000:39 –52.

28. Astakhov V, Gupta A, Grethe JS, *et al.* Semantically Based Data Integration Environment for Biomedical Research. In: *Proc of the 19th IEEE Symp Comput Based Med Syst*; 22-23 June 2006, Washington, DC: IEEE Computer Society, 2006:171–6.

29. Ashish N, Ambite JL, Muslea M, *et al.* Neuroscience Data Integration through Mediation: An (F)BIRN Case Study. *Front Neuroinform* 2010;**4**:118.

30. Stanford J, Mikula R. A model for online collaborative cancer research: report of the NCI caBIG project. *International Journal of Healthcare Technology and Management* 2008;**9**:231–46.

31. González-Beltrán A, Tagger B, Finkelstein A. Federated ontology-based queries over cancer data. *BMC Bioinformatics* 2011;**13**(Suppl 1):S9.

32. Saltz J, Oster S, Hastings S, *et al.* caGrid: design and implementation of the core architecture of the cancer biomedical informatics grid. *Bioinformatics* 2006;**22**:1910–6.

33. Livne O, Schultz N, Narus S. Federated Querying Architecture with Clinical & Translational Health IT Application. *J Med Syst* 2011;**35**:1211–24.

34. Ouagne D, Hussain S, Sadou E, *et al.* The Electronic Healthcare Record for Clinical Research (EHR4CR) information model and terminology. *Stud Health Technol Inform* 2012;**180**:534–8.

35. Core Ontology - SHRINE. https://open.med.harvard.edu/display/SHRINE/Core+Ontology (accessed 18 Apr 2012).

36. Bug W, Ascoli G, Grethe J, *et al.* The NIFSTD and BIRNLex Vocabularies: Building Comprehensive Ontologies for Neuroscience. *Neuroinformatics* 2008;**6**:175–94.

37. D3.5.2_Appendix_E_Ontology_Specifications_01.pdf. http://www.epsos.eu/uploads/tx_epsosfileshare/D3.5.2_Appendix_E_Ontology_Specifications_01.pdf (accessed 8 Dec 2012).

38. epSOS: About epSOS. http://www.epsos.eu/home/about-epsos.html (accessed 11 Apr 2012).

39. Matney S, Bradshaw R, Livne O, *et al.* Developing a Semantic Framework for Clinical and Translational Research. In: *AMIA Summit on Translational Bioinformatics*; 7-9 March 2011, Bethesda, MD: AMIA, 2011:24.

40. Segagni D, Tibollo V, Dagliati A, *et al.* An ICT infrastructure to integrate clinical and molecular data in oncology research. *BMC Bioinformatics* 2012;**13**(Suppl 4):S5.

41. NCI Thesaurus Browser —. https://cabig-stage.nci.nih.gov/community/tools/NCI_Thesaurus (accessed 8 Dec 2012).

42. LexEVS 6.0 Architecture. https://cabig-kc.nci.nih.gov/Vocab/KC/index.php/LexEVS_6.0_Architecture (accessed 30 May 2011).

43. Noy NF, Shah NH, Whetzel PL, *et al.* BioPortal: ontologies and integrated data resources at the click of a mouse. *Nucleic Acids Res* 2009;**37**(Web Server issue):W170–3.

44. Pathak J, Solbrig HR, Buntrock JD, *et al.* LexGrid: A Framework for Representing, Storing, and Querying Biomedical Terminologies from Simple to Sublime. *J Am Med Inform Assoc* 2009;**16**:305–15.

45. CTS2. http://informatics.mayo.edu/cts2/index.php/Main_Page (accessed 14 Jun 2011).

46. LexEVS 6.0 CTS2 Guide - EVS - LexEVS - National Cancer Institute - Confluence Wiki. https://wiki.nci.nih.gov/display/LexEVS/LexEVS+6.0+CTS2+Guide (accessed 2 Jul 2012).

47. caDSR and ISO 11179 - caDSR - National Cancer Institute - Confluence Wiki. https://wiki.nci.nih.gov/display/caDSR/caDSR+and+ISO+11179 (accessed 8 Dec 2012).

48. Warzel DB, Andonyadis C, McCurry B, *et al.* Common Data Element (CDE) Management and Deployment in Clinical Trials. *AMIA Annu Symp Proc* 2003;**2003**:1048.

49. Terminfo Project - Overview. http://www.hl7.org/Special/committees/terminfo/overview.cfm (accessed 9 Dec 2012).

50. NIVEL | LINH. http://www.nivel.nl/en/netherlands-information-network-general-practice-linh (accessed 28 Jul 2012).

51. NIVEL | Netherlands institute for health services research. http://www.nivel.nl/en (accessed 11 Apr 2012).

52. Clinical Practice Research Datalink - CPRD. http://www.cprd.com/intro.asp (accessed 28 Jul 2012).

53. Medicines and Healthcare products Regulatory Agency. http://www.mhra.gov.uk/#page=DynamicListMedicines (accessed 28 Jul 2012).

54. Lenzerini M. Data integration: a theoretical perspective. In: *Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*; 3-6 June 2002, New York, NY: ACM, 2002:233–46.

55. Smith B, Ceusters W. Ontological realism: A methodology for coordinated evolution of scientific ontologies. *Applied Ontology* 2010;**5**:139–88.

56. Grenon P, Smith B. SNAP and SPAN: Towards Dynamic Spatial Ontology. *Spatial Cognition & Computation* 2004;**4**:69–104.

57. Bodenreider O. The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Res* 2004;**32**(Database issue):D267–70.

58. Scheuermann RH, Ceusters W, Smith B. Toward an Ontological Treatment of Disease and Diagnosis. *AMIA Summit on Translational Bioinformatics* 2009:116–20.

59. Goldfain A, Smith B, Arabandi S, *et al.* Vital Sign Ontology. In: *Proceedings of the Workshop on Bio-Ontologies*; ISMB, Vienna: 2011:71–4. http://ontology.buffalo.edu/smith/articles/Vital_Sign_Ontology.pdf

60. information-artifact-ontology - The Information Artifact Ontology (IAO) is an ontology of information entities based on the BFO - Google Project Hosting. http://code.google.com/p/information-artifact-ontology/ (accessed 9 Dec 2012).

61. Brinkman RR, Courtot M, Derom D, *et al.* Modeling biomedical experimental processes with OBI. *J Biomed Semantics* 2010;**1**(Suppl 1):S7.

62. Ashburner M, Ball CA, Blake JA, *et al.* Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet* 2000;**25**:25–9.

63. Luciano JS, Andersson B, Batchelor C, *et al.* The Translational Medicine Ontology and Knowledge Base: driving personalized medicine by bridging the gap between bench and bedside. *J Biomed Semantics* 2011;**2 Suppl 2**:S1.

64. Leysen P, Bastiaens H, Van Royen P. TRANSFoRm : Development of Use Cases. http://transformproject.eu/Deliverable_List_files/D1.1%20Detailed%20Use%20Cases_V2.1-2.pdf (accessed 28 Feb 2013).

65. Qamar R, Rector A. Semantic issues in integrating data from different models to achieve data interoperability. *Stud Health Technol Inform* 2007;**129**(1 Pt):674–8.

66. Park J, Ram S. Information systems interoperability: What lies beneath? *ACM Transactions on Information Systems* 2004;**22**:595–632.

67. Pathak J, Wang J, Kashyap S, *et al.* Mapping Clinical Phenotype Data Elements to Standardized Metadata Repositories and Controlled Terminologies: The eMERGE Network Experience. *J Am Med Inform Assoc* 2011;**18**:376–86.

68. Pavel S, Euzenat J. Ontology Matching: State of the Art and Future Challenges. *IEEE Trans Knowl Data Eng* 2011;**PP**:1.

69. Choi N, Song I-Y, Han H. A survey on ontology mapping. *ACM SIGMOD Record* 2006;**35**:34–41.

70. Shvaiko P, Euzenat J. A Survey of Schema-Based Matching Approaches. In: Spaccapietra S, ed. *Journal on Data Semantics IV*. Springer Berlin / Heidelberg, 2005:146–71.

71. CTS2 - HL7Wiki. http://wiki.hl7.org/index.php?title=CTS2 (accessed 28 Jul 2012).

72. Cimino JJ, Clayton PD, Hripcsak G, *et al.* Knowledge-based approaches to the maintenance of a large controlled medical terminology. *J Am Med Inform Assoc* 1994;**1**:35–50.

73. Cimino JJ. Terminology tools: state of the art and practical lessons. *Methods Inf Med* 2001;**40**:298–306.

74. Miller GC, Britt H. A new drug classification for computer systems: the ATC extension code. *Int J Biomed Comput* 1995;**40**:121–4.

75. BNF.org. http://www.bnf.org/bnf/index.htm (accessed 10 Dec 2012).

5. The Clinical Data Integration Model: Core Ontology of the TRANSFoRm Unified Interoperability Framework in Primary Care.

Note: This chapter forms the basis of a submission to a call for papers for a special edition of Methods of Information in Medicine (http://informatics.mayo.edu/CNTRO/index.php/Events/Special-Issue_MIXHS-Methods_2013). There is therefore some repetition with parts of chapter 4. Numbering of tables and figures are relative to this chapter.

5.1. Introduction

Primary care data is the single richest source of routine health care data, yet its use, both in research and clinical work, often requires combining data from multiple practices. Data integration and interoperability are therefore of upmost importance. They rely on a set of models and mappings, whether explicit or implicit and their formulations, instantiations and maintenance are consequently of considerable importance.

One solution relies on creation and maintenance of *data warehouses*. Data from each local data source is transferred in another structure. If the local source does not share the structure of the data warehouse, an Extract-Transform-Load (ETL) process is used to transfer and transform the data into the target structure. The “Informatics for Integrating Biology and the Bedside” (i2b2) [1] is an example of such an approach. A uniform and unique structure can then be used for queries.

When local sources share a similar structure, *data federation* can be used. The ePCRN project [2] explored this approach for primary care research. It requires all sources to be structured with the American Society for Testing and Materials Continuity of Care Record (CCR) information model. But instead of transferring

data, queries are executed locally and the results aggregated. The Shared Health Research Information Network (SHRINE) [3] uses a similar approach to federate i2b2 sources.

The *mediation* approach allows local sources which are structured differently to be used in context of a distributed infrastructure. A central information model is related to each local model via mappings. Queries are first expressed according to the central model and then “translated” by the system for each local source. Each source therefore retains its structure and control over its data. Central models were initially designed as ontologies and the Advancing Clinico-Genomic Trials (ACGT) [4] was developed according to such a view in the cancer research domain. The central models have also been expressed in other ways like XML (in BIRN project [5] and its follow-up project NIF) or the Unified Modelling Language (UML) for cancer Biomedical Informatics Grid (caBIG) [6].

Nevertheless, no mediation project focused on supporting the primary care research field. TRANSFoRm, an EU FP7 project that aims to comprehensively support the integration of clinical and translational research data in the primary care domain, fills that gap. It will foster patient safety through two types of interventions (see Figure 5.1). The first includes developing a foundation for an efficient decision support system framework, while the second relies on research to enhance care delivery. In this context, both retrospective integrations (through a workbench query tool) and prospective integrations (through electronic case report forms) will be supported. With a European focus, TRANSFoRm will enable interoperability between different types of sources and different countries.

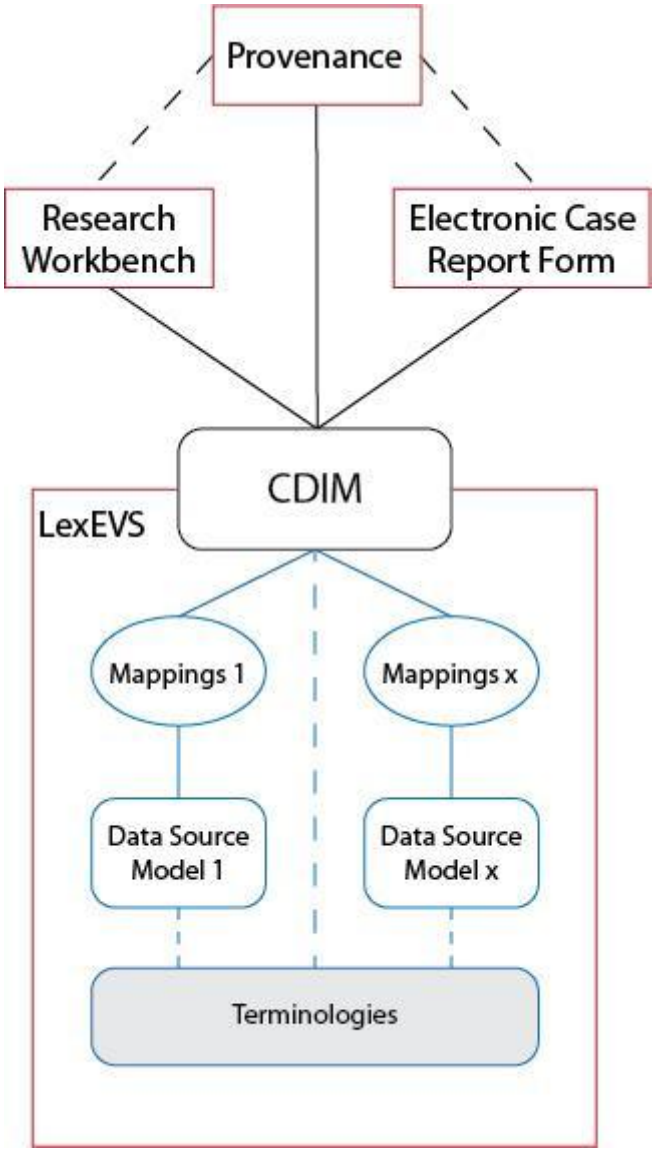


Figure 5.1: CDIM interactions

5.2. Objectives

We developed a unified structural terminological interoperability framework for data mediation based on the Clinical Data Integration Model (CDIM) ontology. We present here specific requirements and characteristics to support primary care in the context of the TRANSFoRm European project.

5.3. Methods

The clinical data integration model (CDIM) was designed to represent clinical concepts relevant to primary care and serve as a basis for data integration in the TRANSFoRm project. Data integration often relies on a combination of two types of models: (1) information models (also called structural models) and (2) terminological models (also referred to as semantic models). These two models need to be "bound" together to be able to fully assert their content. For example, a field in a database might be named "dx" and contain the value "E11.9". By binding the information model (dx represents diagnosis) with the terminological model used (the International Classification of Disease 10 - ICD 10), we can assert that this represents a diagnosis of diabetes type 2. TRANSFoRm utilizes a unified structural/terminological interoperability framework based on the local-as-view paradigm (bringing together information models, terminologies and binding information) (see chapter 4). Such an approach mandates a global information model to describe the domain of interest, irrespectively of the data sources to be explored.

Both types of models are required (information and terminological) since they each carry unique types of information. Terminologies tend to include generic information with the aim of re-usability in different contexts. We call it generic as the same terminological item can be used to represent a possible disease that needs to be explored, a newly confirmed diagnosis, or co-morbidity. It may also been used in the history section to represent an episode that occurred years before or even in the patient's family. Moreover, a terminology like ICD 10 is meant to be used by various systems (e.g. public health surveillance, electronic health records, and billing systems).

On the other hand, information models are developed to address the specific needs of a system. Their depth can vary but they focus on core concepts (e.g. "diagnosis") and omit particular representations of data (e.g. "adenocarcinoma of the prostate") in order to be flexible and they support binding with multiple terminologies (which might vary in depth and coverage).

Nevertheless, there is a grey zone where certain concepts might be found both in the information models and terminologies. For example, should an information model contain concepts like diabetes type 1 and diabetes type 2? Or should it only contain the concept diagnosis, and rely on terminologies to support the relationships between these two diabetes concepts as they can also be found in ICD 10 for example (codes E10 and E11).? This underlines the importance of recognizing that there is a continuum between information models and terminologies.

Given this continuum, we chose to design CDIM with this guiding principle in mind: if information could be found in a recognized terminology used in primary care research (like diabetes and its sub-concepts, which are present in the International Classification of Primary Care - ICPC), then we only included the parent concept in CDIM (in this case “Disease”).

However, exceptions were made if a specific concept was to be used extremely frequently as part of queries. The cardinal example would be systolic and diastolic blood pressures. A systolic blood pressure of 100 mmHg could be expressed with three triplets linked together : physical examination = systolic blood pressure (expressed as a code); physical examination value = 100; physical examination value unit = mmHg (expressed as a code). Yet, if included in CDIM, its expression only require the assignments “systolic BP examination = 100” and systolic “BP unit = mmHg”.

Given the extensive use of such measurements, including it in CDIM facilitates query expressions. As an additional benefit, it can also facilitate mapping to data sources where, for the same reason, these measurements are often stored separately.

5.3.1. Content Development

The specific requirements for primary care data were first gathered through discussions with experts in the fields as well as a through a sampling of various research criteria in order to get a broad view of the domain. Although many concepts like those related to diagnosis, medication or demographics are not unique to

primary care, two are specifically important in primary care: reason for encounter and episode of care. The former is reflection of the fact that many presentations are undifferentiated and varied in primary care (e.g. symptoms like abdominal pain to diabetes control follow-up). The latter represents the fact that, although multiple visits might be coded with a diagnosis of “major depression”, they might all be related to one episode of “major depression”. Recognizing this is of upmost importance to properly assess incidence and related measures.

We also added concepts like physician practice since primary care is often based on smaller organizational units (as opposed to large organizations in tertiary care). Although not directly used to create queries (i.e. not part of inclusion/exclusion criteria), they allow data operations such as restrictions based on selected practices. This is especially important since TRANSFoRm will include a data source quality tool which will allow users to select a subset of practices based on certain quality criteria (completeness, correctness, accuracy, consistency). Supporting organizational units in CDIM also allows a more fined-grain control over data access security as policies can be applied distinctly to different subsets of data.

Requirements were not as clearly delineated for genetic data. The technology is evolving very quickly and data availability is increasing, driving increasingly complex queries. Masys et al. expose a desideratum for integration of genomic data into Electronic Health Records. Based on their layered vision, current requirements for CDIM are interpretive codes (easily actionable) for single nucleotide polymorphisms (SNP), but not the sequence (ACGT) itself.

5.3.2. Ontology

CDIM is part of our goal to unify structural, terminological and binding information. As discussed in chapter 4, we developed our information model as an ontology as it allowed us to design a framework bringing all three model types into LexEVS [8]. As a mediation schema, in the framework, CDIM needs to support data integration from multiple types of data sources, ranging from varied relational databases to xml files, both standard (like HL7 CDA) and non-standard. The current interoperability framework allows this.

We also want to be as forward-looking as possible and compatibility with the semantic web approach is important in that regard and the use of a formal ontology supports this. Initiatives such as R2RML [9] and R2DQ [10] are being developed to bridge relational and RDF data models through mappings. Although not present in TRANSFoRm yet, SPARQL end points might well become interesting storage approaches for bioinformatics repositories and projects relying on data integration will need to take this into account. Using an ontology will facilitate such a development.

Finally, a long standing goal in the bioinformatics community involves fostering interoperability of systems and projects, and developing CDIM as an ontology allows an easier way forward. Our formal ontology facilitates external usage by presenting concepts and relations through formal logic. It allows other parts of the project using semantic web approaches (like the provenance and the decision support systems in TRANSFoRm) to interact with the model natively.

A basic and driving assumption about CDIM is that it follows the realist approach. Two general approaches exist in terms of formal ontologies: the realist and the cognitivist approaches. A cognitivist (or conceptualist) ontology aims at formalizing the concepts we use to categorize the world, as revealed by our common sense and our language: such an ontology has a cognitive and linguistic bias (for example, the DOLCE ontology [11] accepts the distinction between abstract and concrete entities, cf.). Alternatively, a realist ontology aims at formalizing the real entities of the world, which we know through our best scientific theories. Should our best scientific theories change, then a realist-oriented ontology would of course also change (that is, realist ontologist are fallibilist: they reflect our best knowledge of what the world is constituted by).

Since realist ontologies focus on formalizing the real world as described by sciences, they are very well-fitted to the biological domain. Indeed, there has been a significant effort to develop a set of realist orthogonal ontologies covering different parts of the biomedical domain, namely the OBO foundry, whose developers have agreed to the

adoption of a set of principles specifying best practices in ontology development. The Basic Formal Ontology (BFO) [12] aims at being a foundational ontology for the OBO foundry [13], formalizing the top-level entities covering all the different fields.

The medical domain, however, does not reduce to its biological component: it includes also informational objects or apparently mental constructs, like diagnoses, as well as objects whose exact nature is not clearly agreed upon, like diseases. Such objects may seem to be easier to formalize in a cognitivist ontology. However, they can also be efficiently formalized in a realist ontology, as illustrated by the OBO foundry candidate Ontology for General Medical Science (OGMS). For example, OGMS formalizes a diagnosis as an informational content entity about the health status of a patient, that is as a generically dependent continuant (that can be encoded in different digital or physical entities: ink marks on a paper, a computer file, etc.)

A long-standing question in philosophy of medicine concerns the status of diseases. According to objectivists like Boorse (1975), disease is a harmful departure from a body's natural function, and the determination of malfunction and the judgments of harmfulness are an objective matter. On the other hand, according to constructivists, diseases are social artifacts involving normative or value-laden judgments. A realist ontology naturally invites to an objectivist conception of disease; and indeed OGMS defines diseases as dispositions to undergo pathological processes that exists in an organism because of one or more disorders in the organism – in a typical objectivist fashion.

Given the large spectrum of OBO foundry's promising formalization of the biomedical field, we decided to take a realist approach, and to adopt BFO 1.1 as a foundational ontology.

5.4. Results

CDIM is based on a number of popular ontologies, which are fully or partially included. As its foundation is BFO 1.1, all of its classes were imported. Given the

focus and relevance of many concepts found in the Ontology of General Medical Science (OGMS) [14], the Vital Sign Ontology (VSO) [15] and the Information Artifact Ontology (IAO) [16] (all candidates to the OBO foundry), these ontology were also directly imported directly into CDIM. They contributed 313 classes in total (which is less than the sum of classes for each as they share classes). CDIM also integrates terms from other ontologies such as the Ontology for Biomedical Investigations (OBI) [17], and the Gene Ontology (GO) [18]. Since only partial extracts were necessary, they were included through the MIREOT (Minimum Information to Reference an External Ontology Term) method [19], totalling 40 classes.

The resulting ontology has 457 classes (102 unique to CDIM, 353 imported) and 73 object properties (1 sub-property unique to CDIM). Twenty-one novel CDIM classes had to be introduced to represent and manage temporal aspects necessary in TRANSFoRm. We had to explicitly define classes like ‘human birth instant’ through equivalent classes definition (e.g. temporal_instant and ('has temporal occupant' some 'human birth')) in order to provide a mapping target with a URI to serve as an anchor since the unified interoperability framework has to support operations that are not based on semantic web (and reasoning). Internal validity and consistency was checked using HermiT 1.3.7 [20].

In order to address specific requirements of primary care, a class “health care episode” was created, such that “’health care process’ has_part ‘health care episode’”. Of note, an encounter can be part of multiple episodes as many problems can be dealt with during a single visit. Given the variety of possibilities for “reason for encounter”, it is more a role which “inheres in” a symptom, a sign, a disease, etc. This provides flexibility while fully conveying the concept’s semantic.

5.4.1. Use Case Evaluation

CDIM was evaluated in terms of its capacity to support query definitions required by two use cases in TRANSFoRm. The first one is an epidemiological study on genetic risk markers for diabetes mellitus type 2. The main question is “Are well selected single nucleotide polymorphisms (SNPs) in type 2 diabetic patients associated with

variations in drug response to oral antidiabetics (Sulfonylureas)?" The second use case is a randomized controlled trial on proton pump inhibitors (PPI - antacid medication) in gastroesophageal reflux disease (GORD). It will attempt to compare "on-demand" versus continuous use of PPI.

One of the major requirements for both use cases in Transform is the ability to identify eligible patients in the eHR and the primary care data sources. Previous research found that two thirds of all information needed to assess the eligibility of a patient for a trial are related to disease history (mainly: disease, symptoms, signs and diagnostic or lab tests) and treatment history. (Köpcke 2013) This is similar for the Transform use cases. One of the crucial aspects is to minimize misclassification while identifying eligible patients. Therefore it is important to not solely rely on diagnostic codes to identify diagnoses but to also use other patient characteristics like laboratory tests or medication to verify the diagnosis.

The data elements needed for these studies were described in detail to ensure concept coverage in CDIM. The main clinical concepts were: diagnosis (recent and medical history), laboratory test, technical investigations (upper endoscopy), medication, symptoms and signs (difficulties swallowing, signs of gastrointestinal bleeding, unintentional weight loss), and physical examination data (blood pressure, weight, height). The genetic concepts needed for the diabetes use case could be limited to SNPs. The following information also needed to be provided: moment of diagnosis; dates, values and units for all measurements; dates, number and dose for medication. All required concepts could be used and expressed with CDIM.

For example, formulated pharmaceutical can be characterized through several data item entities, including 'active ingredient data item', 'dose form data item' and 'strength data item'. Such formalization can be made compatible with preexisting norms – for example, RxNorm's category 'semantic clinical drug form' could be formalized as the mereological sum of 'active ingredient data item' and 'dose form data item'. Additionally, the instruction given by a prescription can be formalized as a subclass of OBI's 'directive information entity', composed of several 'directive information entity parts'. For example, the prescription 'take Zyrtec 5 mg 7 times a

week during two weeks' is composed of '7 times a week' (which is an instance of 'administration frequency item') and 'during two weeks' (which is an instance of a 'duration of treatment period item').

On the other hand, other "desired" (as opposed to required) items are not present in CDIM. They mostly focused on habits (e.g. level of physical activity/sedentarism, dietary habits) or behavioural interventions (e.g. status of self-management education or performance of self-measurement of blood glucose). Although very important concepts, they are too specific to a research area or very rarely encountered in current data sources. Nevertheless, we will ultimately review those to evaluate which ones are to be integrated in CDIM as the system develops. They will be included if they are relevant to a substantial portion of our target researchers.

5.4.2. Query Formulation Workbench

The TRANSFoRm Query Formulation Workbench provides a user interface for clinical researchers to create clinical studies, design eligibility criteria, initiate distributed queries, monitor query progress, and report query results. Through the Workbench, researchers can browse the TRANSFoRm Integrated Vocabulary Service, search for appropriate criteria concepts and bind clinical codes to these criteria concepts. In addition, researchers can use the federated query infrastructure to remotely search eHR sources and data repositories.

The Workbench implements the eligibility criteria design, query formulation and the query submission for patient counts. It captures eligibility criteria in a computable representation, which is based on the CDIM ontology so the criteria can be translated into executable query statements at the data sources using CDIM to data source model mappings (see chapters 6 and 7).

Researchers use the Workbench to build study protocols, composed of sets of inclusion and exclusion criteria. The specification of each individual criterion is based on CDIM concepts and a constraint model. They are then grouped to form application friendly reusable units called CDIM-artefacts.

Let us consider the example of an inclusion criterion for patients who had an HbA1c test result of ≥ 6.5% on or before the 16/04/2013 date. The Laboratory Measurement artefact aggregates relevant concepts closely related to the laboratory test concept extracted from CDIM, such as Test Type, Date of Test and Test Value. It is one of seven artefacts (such as demographics, medications etc.) currently used within the Workbench. The structure allows new artefacts to be easily added as per user requirements.

Figure 5.2: Specifying attributes of time and values

The example criterion is specified by a user of the Query Formulation Workbench as shown in Figure 5.2. The Laboratory Test artefact is presented to the user in the form of a template for entering values for concepts, operators and values. These correspond to the CDIM triplets in Table 5.1. Logical Observation Identifiers Names and Codes (LOINC) is a universal code system for identifying laboratory and clinical observations [21]. The type of test is HbA1c, with LOINC code 4548-4. Units are represented as Ontology of Units of Measurements (UO) [22]. The unit for HbA1c is % (ratio), with UO code value 0000187.

Ontology Label	Operator	Value
Laboratory_Test_Type_ID	=	([LOINC;4548-4])
laboratory_measurement_datum	≥	6.5
laboratory_measurement_unit_label	=	([UO;0000187])
lab_result_confirmation_instant	≤	16/04/2013

Table 5.1: Example of Laboratory Test Measurement criteria (terminology version omitted for simplicity)

5.5. Discussion

As mentioned previously, some “desired” concepts were not initially present in CDIM. It is to be expected that not every single point evaluated in a research project will make its way in CDIM. Some niche concepts might never be included in order to keep the ontology manageable and relevant to most users. Nevertheless, we clearly chose a local-as-view approach and the decision to include a concept or not must be based on relevance to the users and not to its availability in data sources. In this context, a high number of queries using a concept but returning no data can be highly informative. As incentives are put in place to foster use of electronic health records, such information might help guide such incentives, at least in terms of research needs.

Genetic (and eventually proteomic) primary observations will be more easily leveraged with time and as their availability increases. At some point, personal molecular differences (represented as offsets from a standard genome) might be relevant to the researcher and such concepts will also need to be included in CDIM. Nevertheless it is unclear, given the high heterogeneity inherent to the field of translational research, as to which level of precision the models will need to abide by.

5.5.1. Integration with TRANSFoRm Components

As a proof of concept, the current version of TRANSFoRm query workbench demonstrates an intuitive linkage between eligibility criteria and CDIM concepts and implements a simple constraint model, which satisfies the key query requirements of TRANSFoRm use cases.

TRANSFoRm uses data and process provenance as means to achieve traceability and audit in a heterogeneous system, with individual tasks executed by multiple software tools. The novelty of the TRANSFoRm provenance framework is that it links the provenance model, represented with the Open Provenance Model standard, to

the medical domain models, by means of bridging ontologies, thereby enabling verification with respect to established concepts.

CDIM is a key element in this approach, since it allows a uniform conceptualization of annotations in provenance traces that are produced by multiple tools and across national boundaries. This has direct impact on the ability of the system to be audited in a consistent manner regardless of whether it is a clinical trial design conducted in Germany, or a record of data extraction for an epidemiological study in France.

5.5.2. Collaboration

Although TRANSFoRm develops various services and tools to foster more efficient and creative research in the primary care research community, other initiatives will always co-exist. Similarly, although primary care data sources contain extremely valuable information, even greater potential lies in aligning primary and secondary/tertiary care data in order to truly have a picture of a patient's clinical evolution. Therefore, facilitating interoperability between TRANSFoRm and other initiatives is also very important. When adding specific concepts like "reason for encounter" or "episode of care", we made sure to add them in a way which would not impact other more common concepts. For example, "reason for encounter" being a role, the classes of diagnosis, symptom and sign did not need to be modified at all in order to include it.

So while part of the specificity of primary care is reflected in the addition of a few classes like those previously mentioned, the framework leverages existing terminologies as much as possible (e.g. ICPC).

The reusability of the ontology is thereby enhanced since a large part of the specific requirements can be handled by the binding of terminologies providing sufficient precision and coverage for the desired context. The ontology can then focus mostly on common concepts in translational research in large part found in existing initiatives (e.g. OGMS, IAO and VSO). This facilitates ontology alignment and interoperability with other projects. This could be achieved with projects like epSOS [23] (aiming at developing cross-border electronic health information transfer) as it is also using a core ontology based on BFO.

5.6. Conclusion

Using a core ontology service to define system-wide semantic interoperability enables simplicity and consistency of design. The operation of systems such as query or form generation is maintained separately from their content. The TRANSFoRm ontology-based approach to dealing with heterogeneity has the potential to provide a flexible and highly extensible framework for all types of interaction between health record systems and research systems. CDIM is highly extensible with additional domain ontologies readily added to the service. As only one ontology service runs within a given instance of the TRANSFoRm platform, only updating that ontology is necessary for the whole system to be extended. As the requirements of translational medicine extend, new domains will be needed on a regular basis. Proper versioning of the ontology service, combined with a provenance service to track versions becomes a core requirement for such systems.

TRANSFoRm requires source databases or EHR systems to provide mappings for their data source model to CDIM, but these mapping allow operation of all TRANSFoRm functions, including feasibility, recruitment, data extraction and linkage, follow up and embedded semantically controlled eCRFs. When source data schemas change, only the mappings need to follow suit. An additional benefit is that the necessary references to ontologies and terminologies can be created and 'set within' existing standards such as the CDISC Operational Data Model, which do not necessarily support ontologies, least of all the native creation of complex data elements constrained against both a clinical and research ontology. Systems that support standards can thus be rapidly extended to support CDIM without abandoning the existing standard.

- The full CDIM as an OWL file (CDIM_Merged_v3.owl) can be found on the project wiki at: https://transform.kcl.ac.uk/groups/models/wiki/c1d01/Models_for_D64.html (Contact info@transformproject.eu to access these files.)

5.7. References

[1] I2B2, https://www.i2b2.org/

[2] ePCRN, http://www.epcrn.bham.ac.uk/

[3] SHRINE, https://www.i2b2.org/work/shrine.html

[4] ACGT Project, http://www.ncbi.nlm.nih.gov/pubmed/18694140

[5] BIRN, http://www.birncommunity.org/

[6] caBIG, https://wiki.nci.nih.gov/display/caEHR

[7] Masys et.al., J Biomed Inform. 2012 June; 45(3): 419–422. Published online 2011 December 27. doi: 10.1016/j.jbi.2011.12.005

[8] LexEVS 6, https://wiki.nci.nih.gov/display/LexEVS/LexEVS

[9] R2RML, http://www.w3.org/TR/r2rml/

[10] R2DQ, http://d2rq.org/

[11] DOLCE, http://www.loa.istc.cnr.it/DOLCE.html

[12] BFO, http://www.ifomis.org/bfo

[13] OBO Foundry, http://www.obofoundry.org/

[14] OGMS, http://code.google.com/p/ogms/

[15] VSO, http://code.google.com/p/vital-sign-ontology/

[16] IAO, http://code.google.com/p/information-artifact-ontology/

[17] OBI, http://obi-ontology.org/page/Main_Page

[18] GO, http://www.geneontology.org/

[19] MIREOT, http://obi-ontology.org/page/MIREOT

[20] HermiT, http://www.cs.ox.ac.uk/isg/tools/HermiT//

[21] LOINC, http://loinc.org/

[22] UO, http://code.google.com/p/unit-ontology/

[23] epSOS, http://www.epsos.eu/

6. A Generic Data Source Model for the unified interoperability framework

6.1. Introduction

Clinical and research sources of data present a range of methods of access, data representation, content and support. It is imperative that a software interface is available from the TRANSFoRm platform that allows sources to 'plug in' to the platform with the minimum of effort. In TRANSFoRm this interface is configured and driven by a set of models: the clinical data integration model (CDIM, WT 6.5), a data source model (DSM, WT 6.6) and a mapping model connecting the two (CDIM-DSM, WT 6.6). Platform software falls into two categories: (1) user workbench, middleware for routing queries to data nodes, semantic mediator (SM) which is configured by instantiated models and a data node console (DNC) for authorising queries; and (2) model editors which allow local experts to create and maintain the DSM and CDIM-DSM models. This arrangement is shown in Figure 3.1 of section 3.

The DSM permits a definition of the source data structures both in terms of how they are represented to the semantic mediator and how they are represented locally. For example, the native source data structures may be relational (SQL), and this is captured by the DSM in more abstract terms useable by the mediator. Similarly, sources representing their data as XML, HL7v2 messaging and HL7v3 messaging can be captured in this way. The mapping model then captures the relationship between the structural elements within the DSM and the concepts of the CDIM.

We start by proposing an abstract model for data source structures using an analysis of the structure of the types of sources of data the TRANSFoRm platform must support, i.e. relational (SQL), XML and messages (HL7v2/v3).

This model is then tested again two prominent clinical data repositories by looking at their data models, clinical content and associated documentation. An important objective was to establish the degree of software support required in constructing

source and mapping models, and the local expertise required to complete the process.

6.2. The Data Source Model (DSM)

During the first year of the TRANSFoRm project surveys were performed to locate data sources relevant to the primary care and research domain (WT 1.2, D10.1). These consisted of repositories of clinical data obtained from EHRs; vendor-specific EHR systems; and research data repositories, particularly those related to biobanks. From this review only two types of data interface were found: relational database management systems (RDBMS) for repositories, and HL7v2/3 message interfaces for EHR systems.

Figure 6.1 shows the structure of data available through these interfaces. In all cases the structure can be expressed in an hierarchical manner. In addition, positions within the hierarchy can be specified, and referenced from elsewhere within the hierarchy. This suggests that a general model can be defined which allows all three types of data structure to be expressed. This model is shown in Figure 6.2 and offers a means for configuring the semantic mediator.

Components of the source data structures are represented by the class *Entity* in the data source model, with mediator-facing abstractions provided by the attributes *representation type* and *representation value*. The technology-specific aspects of the source data structures are provided by the attributes *structure type* and *system type*. The ability of one entity to reference another is provided through the entity-relationship class (ER). When representing the model in XML this is often implicitly defined through the nesting of entities, but each entity has a unique reference which permits explicit reference.

By taking into account data types and formatting the data source model allows substructure within the data sources to be taken into account. For example, consider a string field within a relational database that is used to represent a date. This date may be formatted as ‘ddmmyyyy’ according to a specified collating sequence. This

can be considered as substructure to the field itself which requires to be understood. However, this substructure is not provided directly by the relational technology. As a result, such substructure is not provided by the *structure type* and *system type* attributes, but is handled by introducing extra options for the *representation type*. Thus, we may introduce representation types of DTddmmyyyy or DTmm-dd-yy, etc. Possible values for all the entity attributes are given in Table 6.1 and an example application is given in Table 6.2.

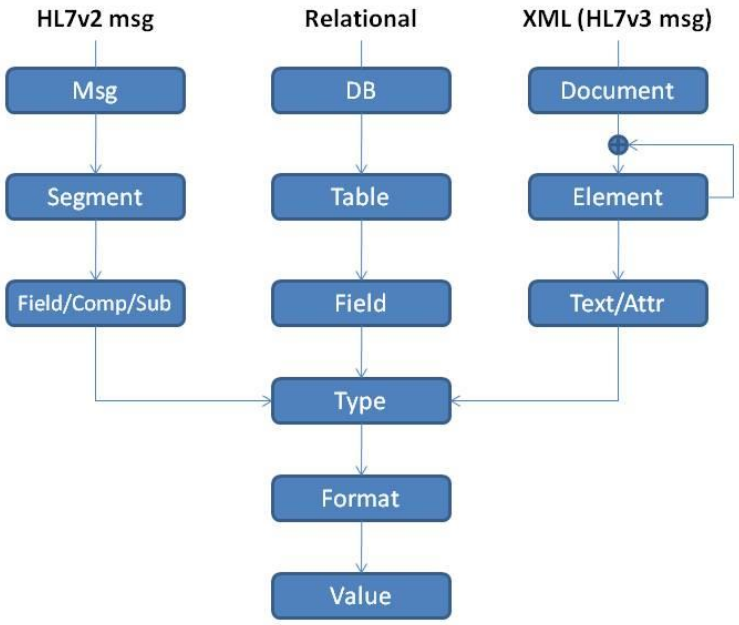


Figure 6.1: The structure of three typical types of data source expressed as hierarchies.

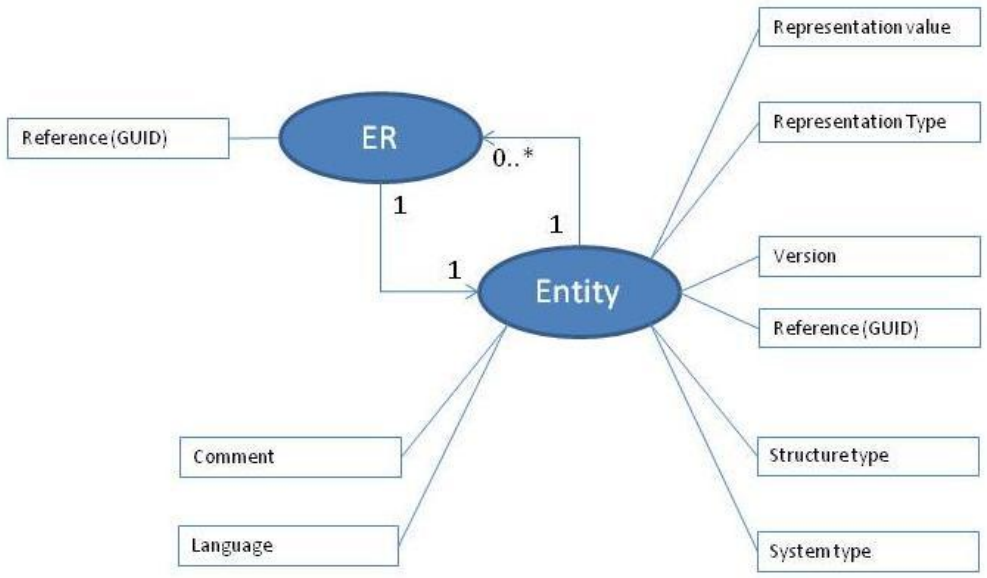


Figure 6.2: TRANSFoRm data source model (DSM)

Representation types	Structure types	(Relational) System types
Collection Domain (Collection) Item Identifier (ID*) Coded value (CV*) Coded ordinal (CO) Physical quantity (PQ) DateTime (DT*) Structured text (ST) Free text (FT) Value Count Dependency	RelDb RelTable RelField XMLDocument XMLElement XMLText XMLAttribute HL7v2Msg HL7v2Segment HL7v2Field HL7v2Component HL7v2Subcomponent LexEVS	[Unsigned] (BigInt, Integer, SmallInt, TinyInt) Single, Double Boolean [Long][Var] (Char, Wchar) [DB] (Date, Time, TimeStamp)

Table 6.1: Representation types and structural types recognised by the semantic mediator.

*Representation types ending in * include a set of possible representation types. For example, CV* includes CV_ICD10, CV_LOINC, etc, while DT* includes DTddmmyyyy, DTmm-dd-yyyy, etc.*

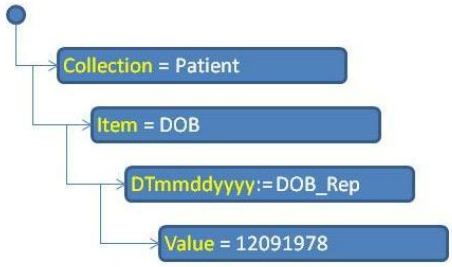
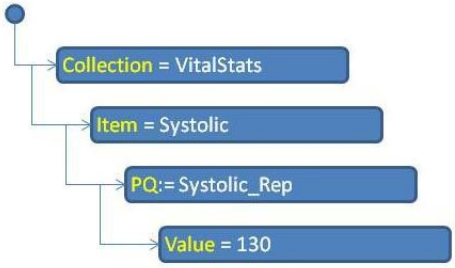
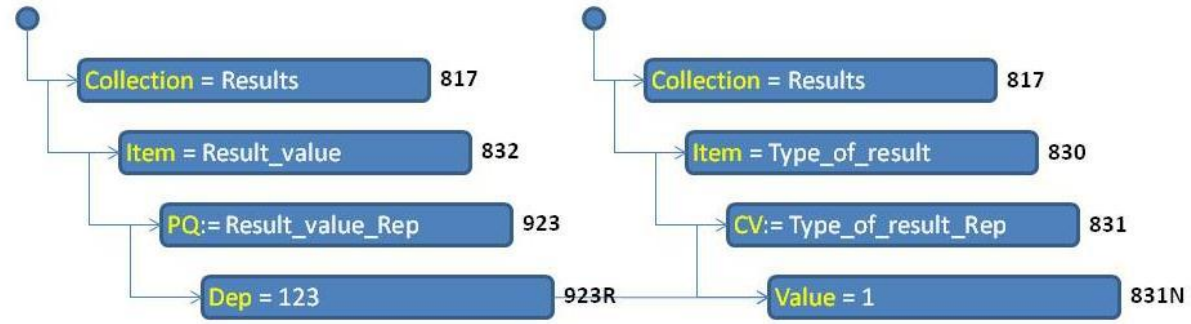
(1) String field holding patient's date of birth			
			
Representation type	**Representation value**	**Structure type**	**System type**
Collection	Patient	RelTable	UserTable
Item	**DOB**	**RelField**	**String**
DTddmmyyyy	DOB_Rep	-	-
Value	12091978	-	-
(2) Integer field holding patient systolic blood pressure			
			
Representation type	**Representation value**	**Structure type**	**System type**
Collection	VitalStats	RelTable	UserTable
Item	**Systolic**	**RelField**	**Integer**
PQ	Systolic_Rep	-	-
Value	130	-	-

Table 6.2: Example entities and their attributes for two cases:

(1) string field holding patient's data of birth, and (2) Integer field holding patient systolic blood pressure.

At first sight the *value* representation type is not essential for instantiating a source model. However, its use permits dynamic structure to be incorporated. For example, very often in a relational database the representation of data within a field may depend on the value of another field. If the source model is instantiated to include unique values for these controlling fields they can be referenced as constraints on another field. The controlled field can have multiple representation types and these can have child entities that represent a dependency and which reference the value entity of the other field. An example of this dynamic structure is given in Table 6.3.



String field holding physical quantity under certain circumstances			
Representation type	**Representation value**	**Structure type**	**System type**
Collection	Results	RelTable	UserTable
Item	Result_Value	RelField	String
PQ (physical quantity)	Result_Value_Rep	-	-
Dependency	Entity 123	-	-
Integer field holding code for type of result			
Representation type	**Representation value**	**Structure type**	**System type**
Collection	Results	RelTable	UserTable
Item	Type_of_result	RelField	Integer
CV (coded value)	Type_of_result_rep	-	-
Value (Entity 123)	1	-	-

Table 6.3: Two structures within a DSM, one specifying a dependency on the other.

The model can be extended further with the introduction of the representation type *count*, indicating the number of times a particular representation value occurs. The two representation types – *value* and *count* – allow for the aggregation of an entire database. A software tool used at the source can extend an initial model by evaluating these two entities. The resulting model extension can be used to define dependencies; assist with determination and upload of local terminology to the platform terminology server (WT7.2); and assist the data quality tool (WT 5.1, D5.1) since the fundamental operations that are performed to assess completeness, correctness and accuracy of data are enumeration and counting.

6.3. Instantiating the NIVEL source model

To test the model a copy of one year's data was provided by NIVEL [3] in conjunction with documentation on the database schema provided by their informatics personnel. Database tables and fields and their descriptions were

designated in Dutch, which enhanced the test further. The data from the National Primary Care Database (NPCD) [4] was provided as a backup copy of the actual MS-SQL RDBMS database used by NIVEL to perform queries for users. The NPCD schema is shown in Figure 12.1 in Appendix A. As can be seen this model is centred on the concept of a *problem* with *diagnosis*, *treatment* and *prevention* as major related concepts. Notice that the results of investigations are not linked to problems. This is distinct to the other clinical repository which we instantiated, that of CPRD (Figure 12.2 in Appendix A). There the model is *consultation* focussed with diagnosis (clinical details), investigations (test results) and treatment (therapy) as major related concepts. This difference in design is a useful test of TRANSFoRm's data source model. Further detail on both these schemas can be found in Appendix A.

The NPCD data source model was instantiated automatically from the source database using a software tool specifically for instantiating a model for a relational database. It parses the relational schema and enumerates and counts all values stored in the fields of the database. The automatically generated model was then further developed by updating the representation types, and defining dependencies between entities of the model. This initial model consisted of entities for collections (tables), items (fields) and representations of the item (coded value, physical quantity, etc.). Initial representations of the item could be guessed by the software from the field type, but could be changed later to reflect the correct substructure. A defect in the database provided was the absence of primary and foreign keys. No explanation was given for this; however it meant that internal references within the model could not be generated automatically and had to be entered into the source model. Also, although not essential for the data source model, it was found convenient to incorporate documented descriptions of the model entities using the *comment* attribute. For example the comment for DOB_Rep in Table 6.2 above is 'Date of birth of patient'.

A portion of the NPCD data source model is shown in Figure 6.3 where an automatically generated fragment (left) is augmented by expert input to achieve the final result (right).

The NPCD data source model (NPCD DSM v1.0.xml) can be found on the project wiki at: https://transform.kcl.ac.uk/groups/models/wiki/c1d01/Models_for_D64.html (Contact info@transformproject.eu to access these files.)

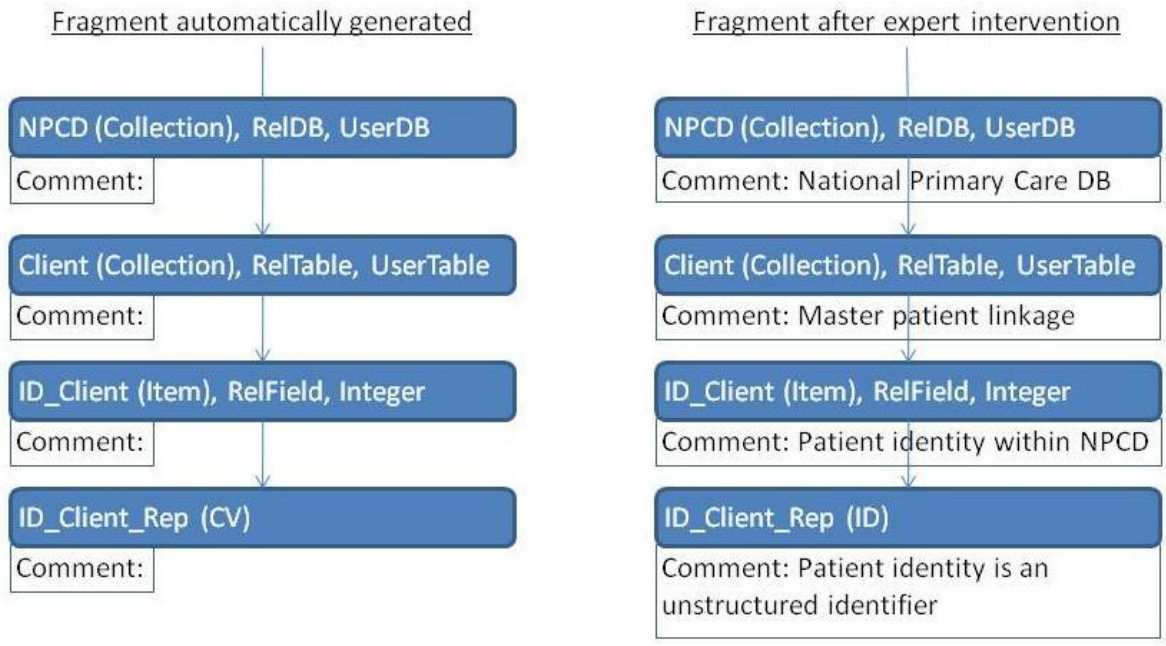


Figure 6.3: Fragments of the NPCD data source model after automatic generation (left) and expert intervention (right).

6.4. Instantiating the Biomina genetic repository

Genetic data sources were investigated early in the project in 2010 and more recently. A consistent theme was the lack of a managed resource for genetic data. Data were invariably available as flat files in various formats, some proprietary, and these were not stable over time. The notion of a genetic repository of known structure, which could be accessed through the TRANSFoRm platform, was not viable. More recently, repositories using relational technologies have become available to the project. However, while schema information has been provided, data has not yet so far been made available.

The Biomina repository is a store of genetic variation data from patient blood samples. The repository is a relational database consisting of 5 principal tables shown in Figure 6.4 below. Unique known population variants are maintained in the

Variants table independently of any patient samples. These variants are annotated from public resources such as dbSNP v135 and RefGene (formerly UCSC) and these annotations are provided in tables Variant_x_ANNOVAR_snp135 and Variant_x_ANNOVAR_RefGene respectively. Patient characteristics relevant to the repository are held in the Sample table with pointers to external patient details. There may be multiple samples per patient. The variants found in these samples are specified through the table Variants_x_sample. This model is easily instantiated as a TRANSFoRm data source model.

Appropriate mappings for this source might include:

CDIM concept id	CDIM concept name	Table	Field
TMO_0038	Genotype with SNP	Variant_x_ANNOVAR_snp135	rsID
CDIM_000003	Patient CRID symbol	Sample	XID

Table 6.4: Possible mappings for some CDIM concepts to the Biomina data source model.

If used in a CDIM Artefact, the translation procedure offered in section 8 would automatically provide the correct joins to bring these fields into the same relation and permit their extraction by the workbench.

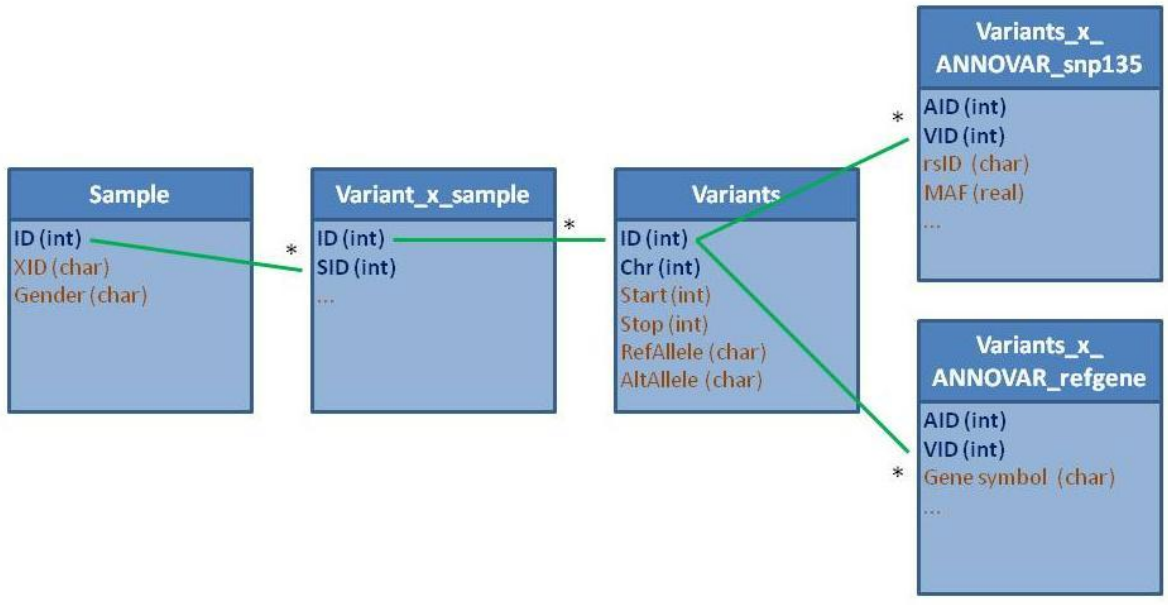


Figure 6.4: Partial schema for Biomina genetic repository.

Unique known variants are maintained in the Variants table and are annotated from public resources such as dbSNP v135 and RefGene (formerly UCSC). The presence of these variants in patient samples is provided by the Variant_x_Sample table. Fields for primary keys are marked in dark blue and field types are in parenthesis. ForeignKey-PrimaryKey relationships are shown by solid green lines.

A more complete definition of the tables is given in Table 6.5 below. At the time of writing this report, we await access to data from Biomina and other genetic data sources to complete this part of the work.

	Variants
Id	Unique number for this variant
Start	Genomic position
Stop	Genomic position
Chr	Chromosome
RefAllele	Allele listed as the 'normal' reference in public databases
AltAllele	The observed allele. Multiple observed alleles at the same position are stored as separate variants (e.g. : A/G and A/T in 2 samples at same position)
	Samples
Id	Unique sample ID
Name	Arbitrary string to identify the sample and provided by the user (e.g. DNA sample ID, link to hospital system ID, ...)
Gender	Male/Female
	Variant_x_sample
Id	Variant id
Sid	Sample id
Multiallelic	0/1: 1 if 3 or more alleles found at same position in same sample, meaning there are 2 variants with same position linked to this sample.
Validation	arbitrary information on experimental validation of the variant
PhredPolymorphism	QC-value from the variant-calling algorithm
AltCount	0/1/2 : Number of alternative alleles present, corresponds to homozygousReference/Heterozygous/HomozygousAlternative
RefDepth	for NextGen Sequencing: number of reads containing reference allele
AltDepth	for NextGen Sequencing: number of reads containing alternative allele
Other columns	QC-values from the variant-calling algorithm
	Variants_x_ANNOVAR_snp135 (annotation table)
Aid	annotation-id
Vid	variant-ID (Allows us to store annotation once for variants seen in multiple samples)
rsID	dbSNP id if the variant exists in dbSNP, -1 otherwise
MAF	Minor allele frequency if available in dbSNP, 0 otherwise
PopSize	Number of people the MAF was based on
Clinical	0/1, 1 for SNPs with known clinical associations
	Variants_x_ANNOVAR_refgene (annotation table)
aid	annotation-id
vid	variant-ID (links to the Variants-table. Allows us to store annotation once for variants seen in multiple samples)
GeneSymbol	RefSeq Gene Symbol (eg GAPDH, YwHAZ, etc)
Transcript	Affected transcript varaint: a single gene can have different transcript variants, and thus multiple entries in this table

	per variant
Exon	affected exon number
GeneLocation	exon/intron/upstream/downstream/UTR/...
VariantType	synonymous/nonsynonymous/stopgain/stoploss/frameshift/...
cPointXX	notation of the variant following cPoint conventions, based on cDNA (NT) or protein (AA) position

Table 6.5: Table detail for Biomina genetic repository.

6.5. Preparing a repository source

Figure 6.5 shows the process of preparing a relational data source, such as NPCD, for inclusion in the TRANSFoRm platform. The data source can be conceptually separated into four parts: (1) fields holding data, (2) fields holding control data for dynamic structure, (3) fields holding semantic definitions for locally coded fields, and (4) annotation data for the database as a whole. The latter can be held external to the database if convenient. TRANSFoRm tools created to support WT 6.6 (but not part of the deliverable) allow for the extraction of semantics into the LexEVS server (WT7.2), and the source data structure and dynamic structure into a source model which is also placed on the LexEVS server. Within TRANSFoRm, LexEVS is used for the purpose of storing both terminologies and structural models along with their mappings to the platform ontology CDIM. This is done as part of WP7 (WT7.1, 7.2, 7.5).

The full NPCD data source model (NPCD DSM v1.0.xml) can be found as an XML file on the project wiki at:
https://transform.kcl.ac.uk/groups/models/wiki/c1d01/Models_for_D64.html
(Contact info@transformproject.eu to access these files.)

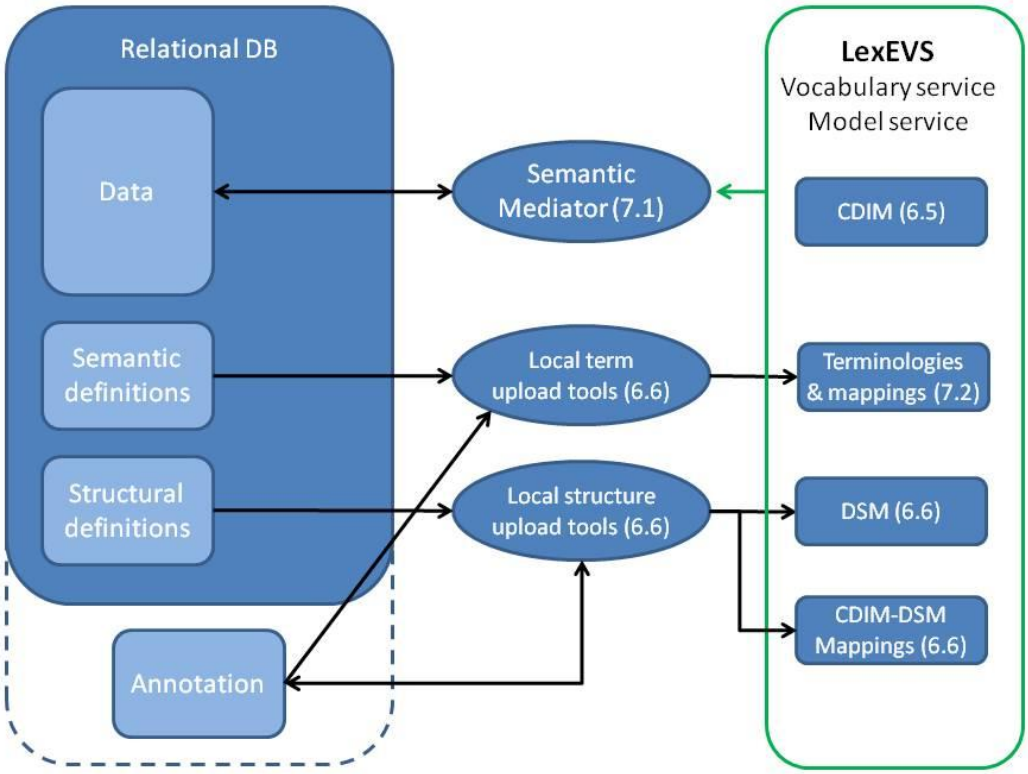


Figure 6.5: Process for preparing a data source for the TRANSFoRm platform.

The semantic mediator (WT7.1) makes use of all models at runtime to translate workbench queries (WT5.3) into local queries (SQL) for execution. The data source model (DSM) is generated automatically using the local structure upload tool (WT6.6) and is further refined through annotation, e.g. defining an integer field as a coded value, or encoding dynamic structure (see main text). This is then loaded into the LexEVS server. The structure upload tool also maps entities in the DSM to the CDIM and these too are loaded into the server. The local terminology mapping and upload tool (WT6.6) extracts local terminologies and semi-automatically creates matches to standard terminologies within the server. Both the local terminologies and generated mappings are loaded into the server. LexEVS is chosen as the server for all the models.

6.6. Internal and External references

Databases invariably define entities that provide reference values for another entity. For example, in a relational database a field may have a foreign key constraint from another table. This table represents the domain from which key values can be chosen. A representation type of *domain* is provided in the data source model for this purpose and this representation type is used for both the source and destination of the reference.

When this reference is internal to the model/database it usually relates to structural dependencies or dynamic structure. However, it can relate to semantic content for which translation will require to be performed by the semantic mediator. For example, a coded value may be a foreign key and refer to gender in another table where definitions are provided. This table must be extracted from the database and loaded into the platform's terminology server along with mappings to standard terminologies. To recognise this fact in the source model the structure type of foreign key (FK) is replaced by a reference to the terminology server and the original table in the local database has no further use to the platform. While this now introduces semantic aspects into the data source model it is natural to leave this reference where it is.

For example, a simple case is the table HULP_GESLACHT (English: Help Sex). This provides a dictionary of administrative genders holding both alphabetic (M, V, O) and integer codes (1,2, 0) for various gender definitions (male, female, unknown). This table must be extracted with the TRANSFoRm tools and loaded into the terminology server along with mappings to the standard terminologies (e.g. V (vrouw) maps to 248152002 (female) within SNOMED-CT, OID 2.16.840.1.113883.6.96). Within the data source model, the *domain* references to this table are changed to references contained within the terminology server. In other words, the domain for these gender codes can now be found on the terminology server rather than the local database.

Another example is the table HULP_UITSLAGHIS (English: Help Results) where the field NHGNummer (English: NHG Number) must be mapped to international standard terminologies such as LOINC [5] or SNOMED-CT [6]. (NHG is the Dutch College of General Practitioners and they are responsible for a classification that provides codes for things such as laboratory results and vital statistics). The eenheid (English: unit) field is similar, mapping to an external standard of units of measure such as UCUM [7] or UO [8].

6.7. Other access interfaces (HL7v2, native APIs)

For EHR systems the most promising mode of access is through HL7 v2 messaging as v3 messaging is not widely used and native APIs are not available to TRANSFoRm at the time of writing. EHR vendors have been approached and will contribute to TRANSFoRm in due course when native APIs will be reviewed. As shown in Figure 3.1 the HL7 v2 messaging structure follows a hierarchy and can be accommodated by the data source model. No work has been done at this time to instantiate a set of messages covering the necessary CDIM concepts. This work will commence when an EHR vendor is available to the project.

7. Mapping model

The TRANSFoRm CDIM-DSM mapping model provides a mapping between a concept within the CDIM model and one or more structures within the data source model. The source model has recognised *entry points* for this purpose. For example, the CDIM concept 'human birth instant' should map to the NPCD structural entity (DT, GEBOORTEDATUM_Rep, 'Date of birth represented as an internal date and time ', Ref=11), although other possibilities exist elsewhere in the model (e.g. Ref=569).
See the NPCD DSM (NPCD DSM v1.0.xml) at
https://transform.kcl.ac.uk/groups/models/wiki/c1d01/Models_for_D64.html
(Contact info@transformproject.eu to access these files.)

In the case of NPCD from NIVEL most CDIM concepts that are needed map directly to entry points within the source model, i.e. only the *equals* operator is required within the mapping model. However, in general, a CDIM concept may require more than one source structure to be accessed and combined in various ways. A general mapping model for this purpose is shown in Figure 7.1. This combination can be as simple as equivalence, or some arithmetical combination such as calculating BMI from height and weight. It might also include various built-in functions providing transformations of various kinds.

The mapping model for the NPCD data source (CDIM-NPCD v1.0.xml) can be found on the project wiki at:
https://transform.kcl.ac.uk/groups/models/wiki/c1d01/Models_for_D64.html
(Contact info@transformproject.eu to access these files.)

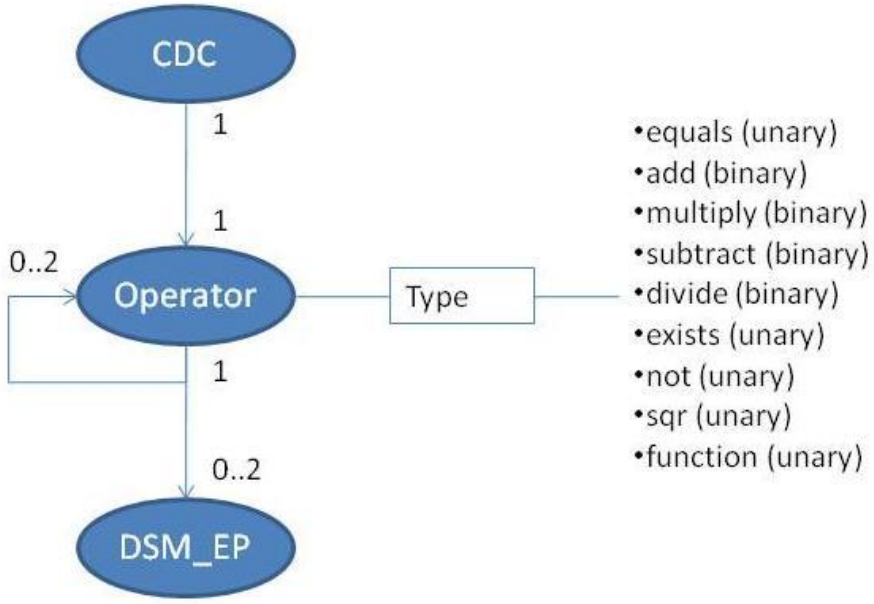


Figure 7.1: CDIM-DSM Mapping model showing CDIM concept (CDC) and Data Source Model entry points (DSM_EP), connected through a series of binary or unary operators.

8. Guidance on application of the models

In this section we offer advice to developers within TRANSFoRm on how the models may be used. In particular, we shall show how to construct CDIM-oriented queries (for use at the workbench) and translate these to queries on the local database (for use by the console at the data node). This discussion is restricted to relational data sources. We will use the example of extracting HbA1c measurements by patient and pharmacy from the NPCD data. To demonstrate this we have used a slightly extended CDIM Artefact-based eligibility criteria model (p60, D5.3). A portion of this model is shown in Figure 8.1 and groups CDIM concepts required to extract related fields within the NPCD data source. These concepts together translate to a single SQL query at the data node and return related fields. The role values are designed to demonstrate the purpose of each concept within the group. However, these roles can normally be inferred from the CDIM ontology itself.

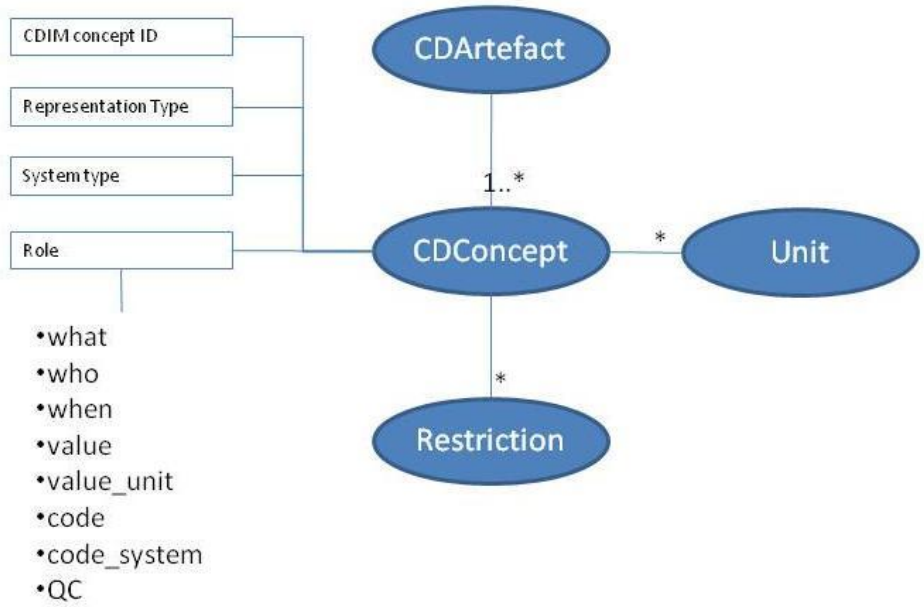


Figure 8.1: A portion of the CDIM-Artefact eligibility criteria model, which groups CDIM concepts for querying data sources.

These concepts together translate to a single SQL query at the data node and return related fields. The role values are designed to demonstrate the purpose of each concept within the group. However, these roles can be inferred from the CDIM ontology itself.

The CDIM-Artefact model illustrates that CDIM concepts can be operationalised in groups. For example, at the workbench there is little point in accessing a laboratory value for eligibility criteria without knowing who it relates to and when it was valid. In addition, a laboratory value may be difficult to process without unit labels as different sources may use different units of measurement. The artefact model permits mapping of multiple CDIM concepts at the same time. Finally, the extended artefact model presented here allows the final representation and system type of the individual CDIM concepts to be set along with the role each concept plays within the group. An example of a CDIM artefact is given in Table 8.1.

CDIM concept id	CDIM concept name	Role	Representation	System type
OGMS_0000056	Laboratory test	What	CV	String
CDIM_000032	scalar measurement datum laboratory finding	Value	PQ	Single
IAO_0000003	Measurement unit label	Value unit	CV	String
CDIM_000029	laboratory test result confirmation instant	When	DT	DateTime
CDIM_000003	Patient CRID symbol	Who	ID	String
OMRSE_00000017	Physician practice role	QC	ID	String

Table 8.1: Example of a CDIM artefact designed to retrieve laboratory test values.

The restriction required to specify which laboratory test to search is not shown, but can be found in the XML representation below.

This specification might have the XML representation below (see Artefact HbA1c.xml at https://transform.kcl.ac.uk/groups/models/wiki/c1d01/Models_for_D64.html (Contact info@transformproject.eu to access these files.)), where the laboratory test is specified as a restriction by a single LOINC code.

```
<CDA Name="Laboratory result HbA1c DCCT" SysProduct="MS-SQL">
<!-- Consists of CDIM elements -->
<CDC cdim="OGMS_0000056"> <!-- laboratory test -->
 <role>what</role>
 <rep>CV</rep>
 <type>string</type>
 <restrict codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC">
  <include>4548-4</include>
</restrict>
</CDC>
<CDC cdim="CDIM_000032"> <!-- scalar measurement datum laboratory finding -->
 <role>value</role>
 <rep>PQ</rep>
 <type>single</type>
</CDC>
<CDC cdim="IAO_0000003"> <!-- Measurement unit label -->
```

```
 <role>value_unit</role>
 <rep>CV</rep>
 <type>string</type>
 <unit codeSystem="2.16.840.1.113883.6.8" codeSystemName="UCUM">
  <report>%</report>
  <translate>mmol/mol</translate>
 </unit>
</CDC>
<CDC cdim="CDIM_000029"> <!-- laboratory test result confirmation instant -->
 <role>when</role>
 <rep>DT</rep>
 <type>datetime</type>
</CDC>
<CDC cdim="CDIM_000003"> <!-- Patient CRID symbol -->
 <role>who</role>
 <rep>ID</rep>
 <type>string</type>
</CDC>
<CDC cdim="OMRSE_00000017"> <!-- Physician practice role -->
 <role>QC</role>
 <rep>ID</rep>
 <type>string</type>
 <restrict localCodeSystemName="NIVEL-Practices">
  <include>1234</include>
  <include>5678</include>
 </restrict>
 <restrict localCodeSystemName="CPRD-Practices">
  <include>5678</include>
  <include>1232</include>
 </restrict>
</CDC>
</CDA>
```

This specification will ultimately be translated to a single SQL statement by the semantic mediator (WT 7.1) at the node for the local database to execute. A subset of the structural mappings for the NPCD data source relevant to the specification above is given below. The semantic mediator will also translate the LOINC codes to the local codes of the data source using the terminology service (WT 7.2). This mapping will have been uploaded to LexEVS by the terminology upload tool (see CDIM-NPCD v1.0.xml at https://transform.kcl.ac.uk/groups/models/wiki/c1d01/Models_for_D64.html (Contact info@transformproject.eu to access these files.)).

```
<!-- CDIM-NIVEL mapping -->
<Map SysProduct="MS-SQL">
<Mapping cdim="CDIM_000003"> <!-- Patient CRID symbol -->
 <operator arg="a" type="equals">
  <dsm_ep>3</dsm_ep>
 </operator>
</Mapping>
<Mapping cdim="OGMS_0000056"> <!-- Laboratory test -->
 <operator arg="a" type="equals">
  <dsm_ep>827</dsm_ep>
 </operator>
</Mapping>
<Mapping cdim="CDIM_000032"> <!-- scalar measurement datum laboratory finding -->
 <operator arg="a" type="equals">
  <dsm_ep>923</dsm_ep>
```

```
 </operator>
</Mapping>
<Mapping cdim="IAO_0000003"> <!-- Measurement unit label -->
 <operator arg="a" type="equals">
  <dsm_ep>4721</dsm_ep>
 </operator>
</Mapping>
<Mapping cdim="CDIM_000029"> <!-- laboratory test result confirmation instant -->
 <operator arg="a" type="equals">
  <dsm_ep>825</dsm_ep>
 </operator>
</Mapping>
<Mapping cdim="OMRSE_00000017"> <!-- Physician practice role -->
 <operator arg="a" type="equals">
  <dsm_ep>602</dsm_ep>
 </operator>
</Mapping>
</Map>
```

In this example XML, only the operation of *equality* is required, i.e. each CDIM concept maps to a single entry point in the data source model.

Take as an example the concept of 'scalar measurement datum laboratory finding' (CDIM_000032). Figure 6.8 shows the structure of the data source model at entry point 923 (NPCD DSM v1.0.xml at https://transform.kcl.ac.uk/groups/models/wiki/c1d01/Models_for_D64.html (Contact info@transformproject.eu to access these files.)). By parsing parent entities in the model we are able to produce the relational fragment `NPCD.Results.Result`. The PQ representation indicates that the value should be cast as a real value (from a string), so the final fragment expression is:

```
Cast(NPCD.Results.Result) as Real
```

By parsing child nodes we find a dependency (Dep) at entry point 831N. Again, by parsing the parent entities for this entry point we obtain the relational fragment `NPCD.Results.Type_of_result` and we do not need to cast this as the system type integer is compatible with the representation coded value (CV). Because the entry point was at a *value* entity the final fragment is:

```
NPCD.Results.Type_of_result=1
```

In a similar way all the other CDIM concepts in the group yield fragments. For example, 'patient CRID symbol' (CDIM_000003) yields: `NPCD.Client.ID_Client.`

Once all the fragments have been identified a single relation must be obtained that covers all the fragments. For example the two fragments above refer to the tables *Results* and *Client*. The source model is therefore parsed looking for a relationship between these two tables yielding the relation:

```
Results left join Client on Results.ID_CLIENT=Client.ID_CLIENT
```

This continues until all the fragments have been incorporated. For this example the final SQL is (in Dutch):

```
Select UITSLAGEN.NHGNUMMER as OGMS_0000056,
UITSLAGEN.WAARDE as CDIM_000032,
HULP_UITSLAGHIS.eenheid as IAO_0000003,
UITSLAGEN.REGISTRATIEDATUM as CDIM_000029,
CLIENT.ID_CLIENT as CDIM_000003,
PRAKTIJK.ID_PRAKTIJK as OMRSE_00000017
 From
(((UITSLAGEN) left join CLIENT on UITSLAGEN.ID_CLIENT=CLIENT.ID_CLIENT)
left join PRAKTIJK on CLIENT.ID_PRAKTIJK=PRAKTIJK.ID_PRAKTIJK) left join
HULP_UITSLAGHIS on UITSLAGEN.NHGNUMMER=HULP_UITSLAGHIS.nhgnummer
 Where UITSLAGEN.TYPEUITSLAG=1
```

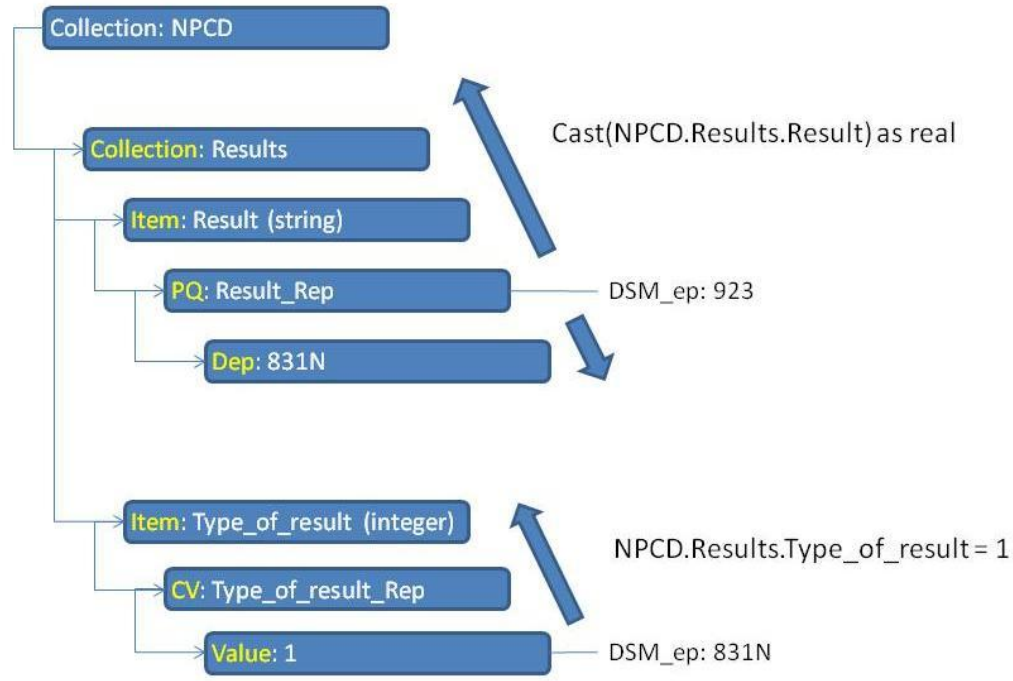


Figure 8.2

In general, multiple entry points to the data source model can be processed into fragments and combined in a single relation. For BMI a particular mapping model for a source might specify the entry points for height and weight and compute the BMI as the mapping model provides for arithmetic operations.

Further example artefacts are provided for comparison with the source model (NPCD DSM v1.0.xml) and mapping mode (CDIM-NPCD v1.0.xml).

See https://transform.kcl.ac.uk/groups/models/wiki/c1d01/Models_for_D64.html

(Contact info@transformproject.eu to access these files.)

Artefact Height.xml
Artefact Gender.xml
Artefact Diagnosis type-2 diabetes.xml
Artefact Medication Zantac.xml

9. Conclusion

Three distinct models have been created and successfully instantiated for two clinical repositories. The Clinical Data Integration Model (CDIM, WT6.5) in combination with the eligibility representation of the Clinical Research information model (CRIM, WT 6.4) and the vocabulary service (WT 7.2) have been used to encode the criteria for the diabetes use-case (WT 1.1, D1.1). The data source models and mapping models have been instantiated for NPCD and CPRD, again covering the diabetes use case. Software developers in WP7 have, using the LexEVS API and the guidance in this document, been able to create the semantic mediator for the TRANSFoRm platform and successfully translate workbench formulated queries to local SQL queries against both NPCD and CPRD. We await genetic sources of data to complete that part of the work in WT 6.6.

10. Abbreviations

BFO	Basic Formal Ontology
Biomina	Biomedical informatics research center Antwerp
BRIDG	Biomedical Research Integrated Domain Group
CCR	Continuity of Care Record
CDASH	Clinical Data Acquisition Standards Harmonisation
CDIM	Clinical Data Integration Model
CIMI	Clinical Information Modelling Initiative
CPRD	Clinical Practice Research Datalink (MHRA), formerly GPRD
CRIM	Clinical Research Information Model
CTS2	Clinical Terminology Services v2
DSM	Data Source Model
EHR4CR	EHR for Clinical Research
ePCRN	Electronic Primary Care Research Network
ER	Entity-Relation
ETL	Extract, Transform and Load
GIM	General information Model
GPRD	General Practice Research Database, now CPRD
HL7	Health Level 7
I2B2	Informatics for Integrating Biology and Bedside
IAO	Information Artefact Ontology
ISO	International Standards Organisation
LexEVS	Lexical Enterprise Vocabulary Service
MHRA	Medicines and Healthcare products Regulatory Agency
NIVEL	Netherlands Institute of Health Services Research
NPCD	National Primary Care Database (NIVEL)
OGMS	Ontology of General Medical Science
OMG	Object Management Group
RDBMS	Relational Database Management System
RIM	Reference Information Model
SHRINE	Shared Health Research Information Network
SQL	Structured Query Language
STRIDE	Stanford Translational Research Integrated Database Environment
TranSMART	Knowledge management platform
UML	Unified Modelling Language
VSO	Vital Signs Ontology
WP	Work package
WT	Work Task
XML	Extensible Markup Language

11. References

[1] LexEVS 6, https://wiki.nci.nih.gov/display/LexEVS/LexEVS

[2] ACGT Project, http://www.ncbi.nlm.nih.gov/pubmed/18694140

[3] NIVEL, Netherlands Institute for Health Services Research, http://www.nivel.nl/en

[4] NPCD is the successor to the LINH database (http://www.nivel.nl/en/netherlands-information-network-general-practice-linh)

[5] LOINC, http://loinc.org/

[6] SNOMED-CT, http://www.ihtsdo.org/snomed-ct/

[7] UCUM, http://unitsofmeasure.org/

[8] UO, http://code.google.com/p/unit-ontology/

12. Appendix A

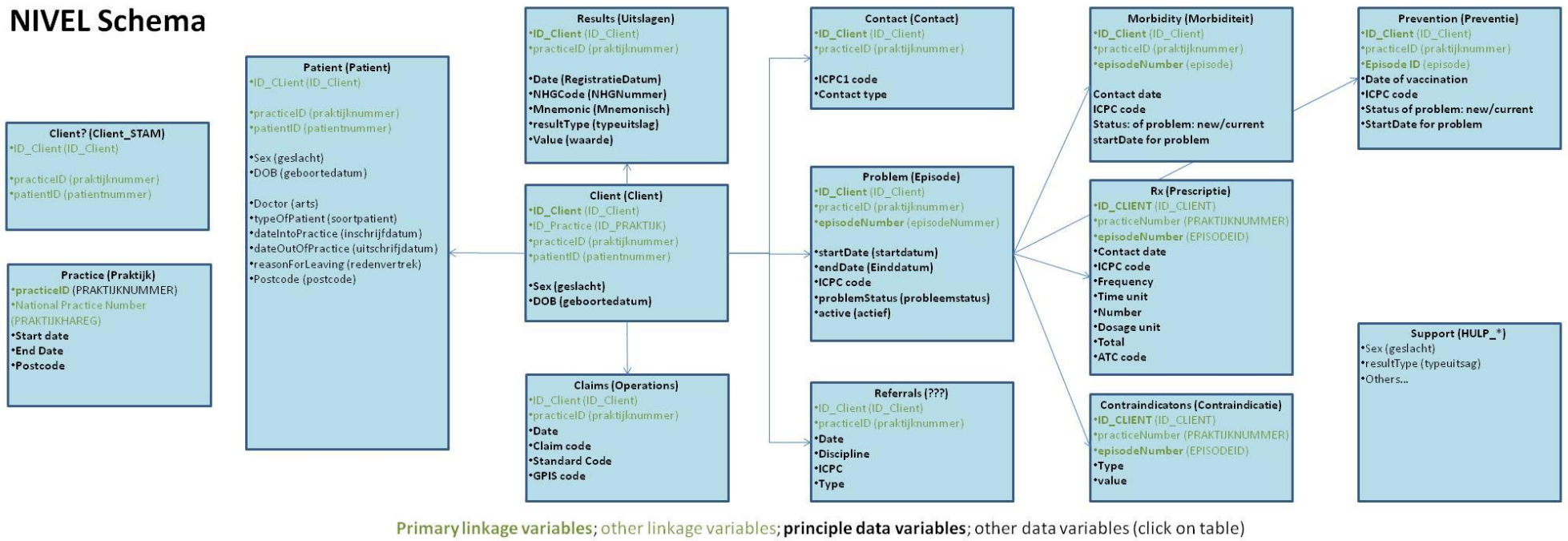


Figure 12.1: National Primary Care Database (NPCD) of the Netherlands Institute of Health Services Research (NIVEL).

This data model is problem-oriented, and almost all other tables holding clinical data are related to the Problem table, these being Morbidity, Rx (prescribing), Prevention, Referrals and Contact. The main exception is the Results table which is held independently of any problems, and at the patient level. The remaining tables are for administrative purposes, i.e. Client, Patient, Claims (for financial claims) and Practice. The HULP_* tables are for code definitions and lookups and define the local semantic content of the database.

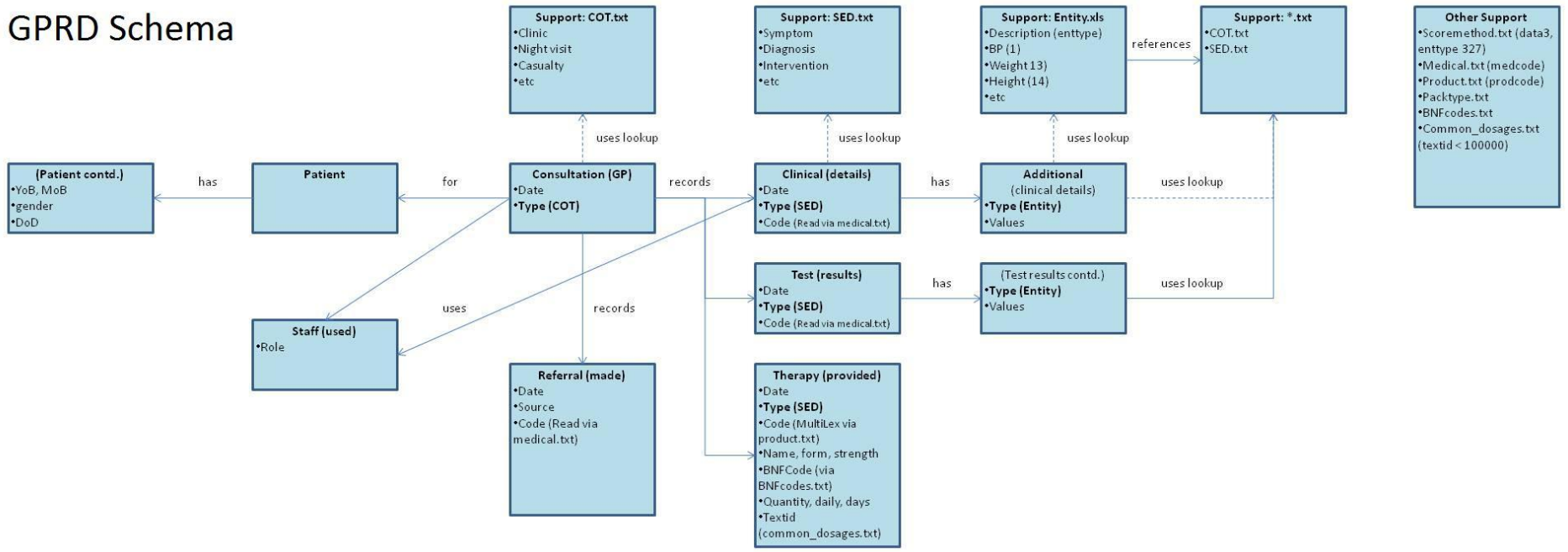


Figure 12.2: General Practice Research Database (CPRD) of Medicines and Healthcare products Regulatory Agency (MHRA).

This data model is consultation-oriented with clinical details, test results, therapies (Rx) and referrals being linked to the consultation. The other main tables are for administrative purposes, these being Patient and Staff tables. All remaining tables are for code definitions and other lookups and define the local semantic content of the database.