

ICT-2010-270253

INTEGRATE

**Driving excellence in Integrative Cancer Research
 through Innovative Biomedical Infrastructures**

STREP
 Contract Nr: 270253

**Deliverable: 3.7 Final version prototype of the semantic
 interoperability layer**

Due date of deliverable: (31-07-2014)
 Actual submission date: (MM-DD-YYYY)

Start date of Project: 01 February 2011

Duration: 45 months

Responsible WP: UPM

Revision: outline

Project co-funded by the European Commission within the Seventh Framework Programme (2007-2013)		
Dissemination level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Service	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (excluding the Commission Services)	

0 DOCUMENT INFO

0.1 Author

Author	Company	E-mail
Santiago Aso Lete	UPM	saso@infomed.dia.fi.upm.es
David Pérez del Rey	UPM	dperez@infomed.dia.fi.upm.es
Kristof De Schepper	Custodix	kristof.deschepper@custodix.com
Jasper Van Leeuwen	Philips	jasper.van.leeuwen@philips.com

0.2 Documents history

Document version #	Date	Change
V0.1	12-06-2014	Starting version, template
V0.2	12-06-2014	Definition of ToC
V0.3	11-09-2014	First complete draft
V0.4	11-09-2014	Integrated version (send to WP members)
V0.5	14-09-2014	Updated version (send PCP)
V0.6	16-09-2014	Updated version (send to project internal reviewers)
Sign off	22-09-2014	Signed off version (for approval to PMT members)
V1.0	22-09-2014	Approved Version to be submitted to EU

0.3 Document data

Keywords	
Editor Address data	Name: Santiago Aso Lete Partner: UPM Address: Facultad de Informática Universidad Politécnica de Madrid Campus de Montegancedo, s/n 28660 Boadilla del Monte, Madrid, Spain Phone: +34 91 336 74 67 Fax: E-mail: saso@infomed.dia.fi.upm.es
Delivery date	

0.4 Distribution list

Date	Issue	E-mailer

TABLE OF CONTENTS

0	DOCUMENT INFO	2
0.1	Author	2
0.2	Documents history	2
0.3	Document data	2
0.4	Distribution list	2
1	INTRODUCTION	5
2	COMMON INFORMATION MODEL UPDATE	6
2.1	Core Dataset	6
2.2	Common Data Model	7
2.3	Query Execution	8
2.4	Query Builder	9
2.4.1	CORE DATASET AUTOCOMPLETE SERVICE	10
3	SEMANTIC NORMALIZATION PIPELINE	13
4	SECURITY INTEGRATION	15
5	SNAQL QUERY ENGINE	17
6	CLINICAL TRIAL METADATA REPOSITORY UPDATE	19
7	CONCLUSIONS	20
8	REFERENCES	21
A.	APPENDIX	22
a.	Mapping ttl (relational to non-relational)	22
b.	Query template library	48
I.	OBSERVATION TEMPLATE	48
II.	PROCEDURE TEMPLATE	50
III.	SUBSTANCE ADMINISTRATION TEMPLATE	52
IV.	DIAGNOSIS TEMPLATE	53
V.	PERSON TEMPLATE	56
VI.	ENTITY TEMPLATE	57

TABLE OF FIGURES

Figure 1: Semantic Interoperability Layer components	6
Figure 2: Common Data Model MySQL Scheme 2.6.2	7
Figure 3: Query Execution	8
Figure 4: Semantic Interoperability Layer brief query process	9
Figure 5: Query Builder example	10
Figure 6: Overview of the AutoComplete within the Semantic Layer	11
Figure 7: Methods provided by the Autocomplete Service	11
Figure 8: Example using Autocomplete Service	12
Figure 9: Normalization Pipeline	13
Figure 10: Query execution security integration	15
Figure 11: SNAQL - Semantic Interoperability Overview	18

1 INTRODUCTION

The purpose of the present deliverable is to define the final state of the semantic interoperability layer within the INTEGRATE project. This deliverable is an update of the deliverable 3.5 [1]. The purpose of its components is to provide a homogeneous access to diverse clinical trial data, being such data coded using terms from different medical vocabularies.

As it is currently described in the deliverable 2.6 [2] section 5.3 "Semantic Layer", the components of the semantic interoperability layer are: query oriented services such as the "Query Builder Service"; access oriented services such as the "CIM Access Service" now named "Query Execution"; binding and semantic reasoning services such as the "CoreDataset Service"; and the relational model component to store data, the "Common Data Model".

There are two additions to the semantic layer, the Autocomplete Service and the Normalization Pipeline. These services enhance Graphical User Interfaces (GUIs) by providing suggestions for concepts to retrieve from the CDM (Autocomplete Service), and automate the normalization process of the databases in order to provide a homogeneous representation of the data (Normalization Pipeline). Along the following sections, all the additions and updates since deliverable 3.5 [1] will be described in detail.

2 COMMON INFORMATION MODEL UPDATE

One of the main components of the Semantic Interoperability Layer is the CIM. This model represents, manages and retrieves clinical information stored in the platform. The Common Information Model is composed by the Core Dataset and the Common Data Model.

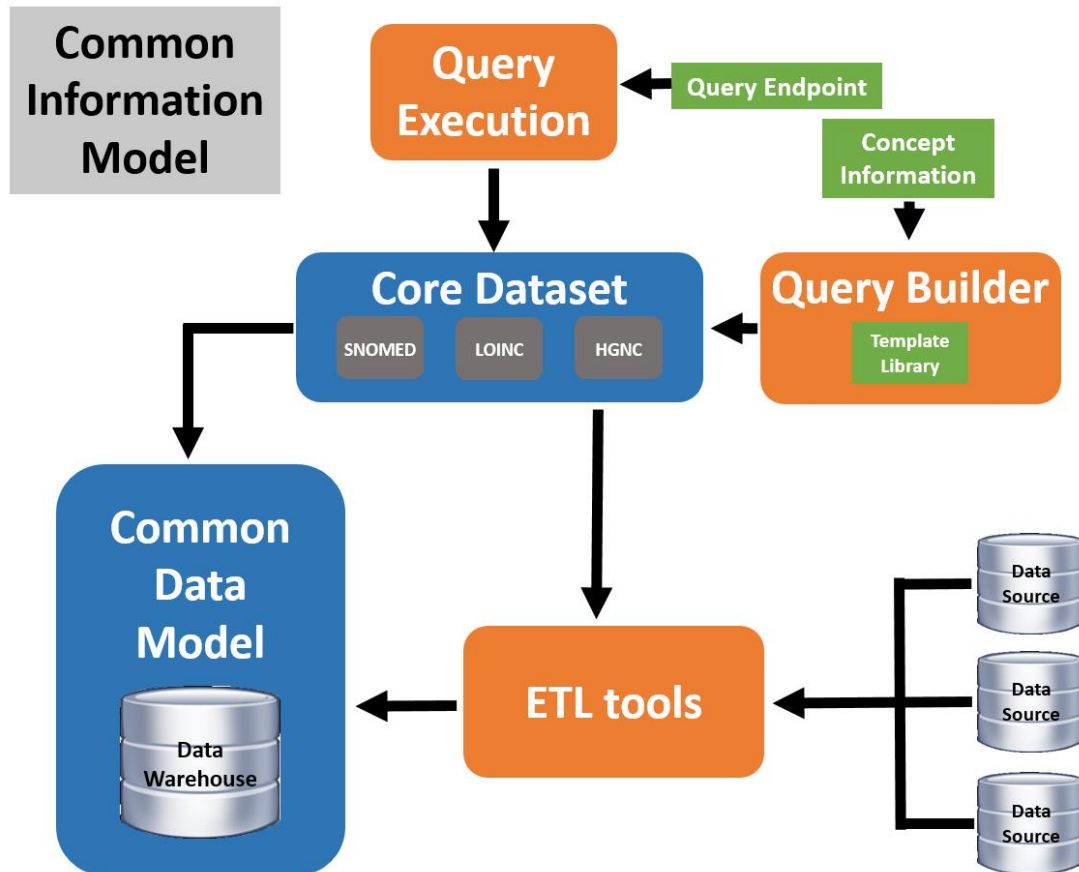


Figure 1: Semantic Interoperability Layer components

The Common Information Model is used by different components, these main components are the Query Execution and the Query Builder. Both components make use of the knowledge of different vocabularies disposed by the Core Dataset, in order to interact with the information stored in the Common Data Model.

2.1 Core Dataset

The Core Dataset establishes a set of standards, vocabularies and rules for information exchange among data models. As it was described in the deliverable 3.5 [1], the vocabularies chosen for the INTEGRATE project were SNOMED CT, LOINC, HGNC and MedDRA. Finally, MedDRA was left out of the project, so that at the final state of the project we are including SNOMED CT, LOINC and HGNC.

With the selected terminologies we are covering general medical terms with SNOMED CT, medical laboratory observations with LOINC and unique and meaningful names for every known human gene with HGNC.

2.2 Common Data Model

The Common Data Model is the scheme of the fundamental base of the semantic interoperability layer, which is the Common Information Model. It is based on the HL7 RIM, as it was decided in Deliverable 3.1 [3], and was described in section 2.2.1 "Relational Model Based on HL7 RIM" from the deliverable 3.5 [1].

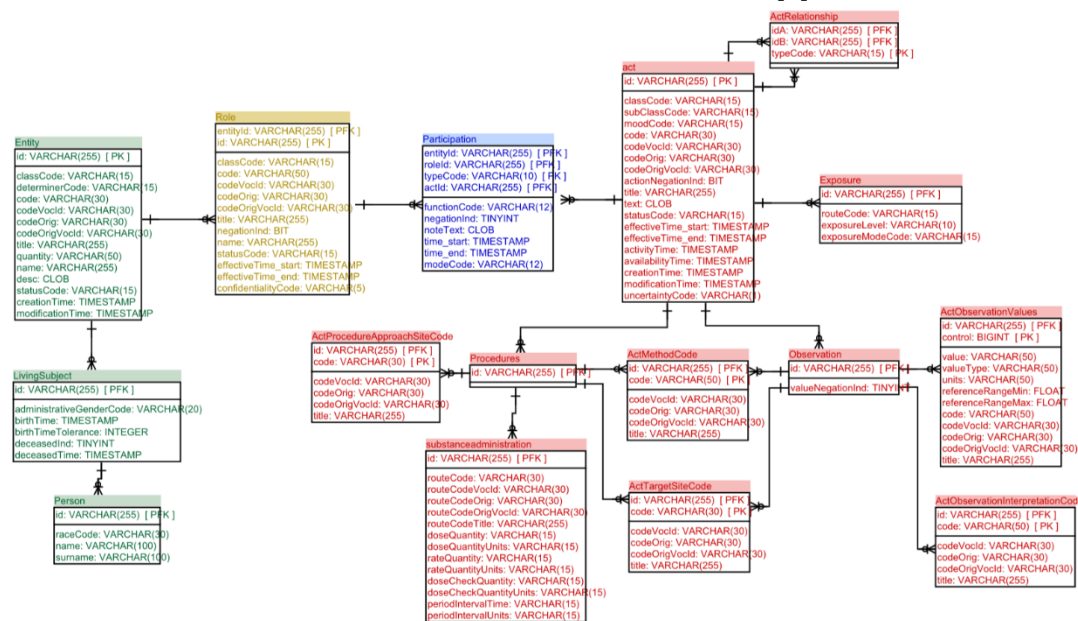


Figure 2: Common Data Model MySQL Scheme 2.6.2

Since the model was deeply described in the previously mentioned section of the deliverable 3.5 [1], just a few changes that took place during the development of the data model are going to be described.

- **Observation interpretation.** This object was completed with the use of attributes that are already present in other objects of the model, as codeVocId, codeOrig, codeOrigVocId and title. This addition focused on providing a better completeness and traceability of the concepts used to express the interpretation of other concepts.
- **Observation value.** `valuePQ` was removed and `value` and `valueType` was added instead,. `code`, `codeVocId`, `codeOrig`, `codeOrigVocId` and `title` were also added. The purpose of this change is to cover cases when the value of an observation is another concept of a terminology, e.g. Observation "247030006|Eye color", and observation value "371246006|Green".
- **Substance Administration.** This object has been updated with several changes, in order to be able to keep all the information needed for substance administrations. The standard addition to provide completeness and traceability with the addition of attributes as `routeCodeVocId`, `routeCodeOrig`, `routeCodeOrigVocId` and `routeCodeTitle`. Attributes `periodIntervalTime` and `periodIntervalUnits` were also added in order to store the periods of the interval of the administration.

- **Act effective time start and end.** The original attribute `effectiveTime` was splitted into `effectiveTime_Start` and `effectiveTime_End`, in order to provide some acts a range of dates where the act took place. (E.g. A substance administration started in some date and ended in another given date).

2.3 Query Execution

As it is described at previous sections, the CDM contains all the data in a relational data model and a SPARQL wrapper to query information with this language. The Query Execution service allows the retrieval of information from the CDM using the SPARQL wrapper by using CIM based queries and CDM based queries. These queries can be expanded as it was explained in the section "Query Expansion" of the deliverable 3.5 [1]. This expansion allows to add to the query, the semantics from the Core Dataset knowledge. Once the query is executed, the Query Execution Service returns the results to a SPARQL resultset format.

In order to transform SPARQL into SQL, an engine that enables this process is required. The engine used for this purpose is the *morph-RDB* [4], an open source project following W3C standards and outperforming similar state-of-the-art tools such as D2R.

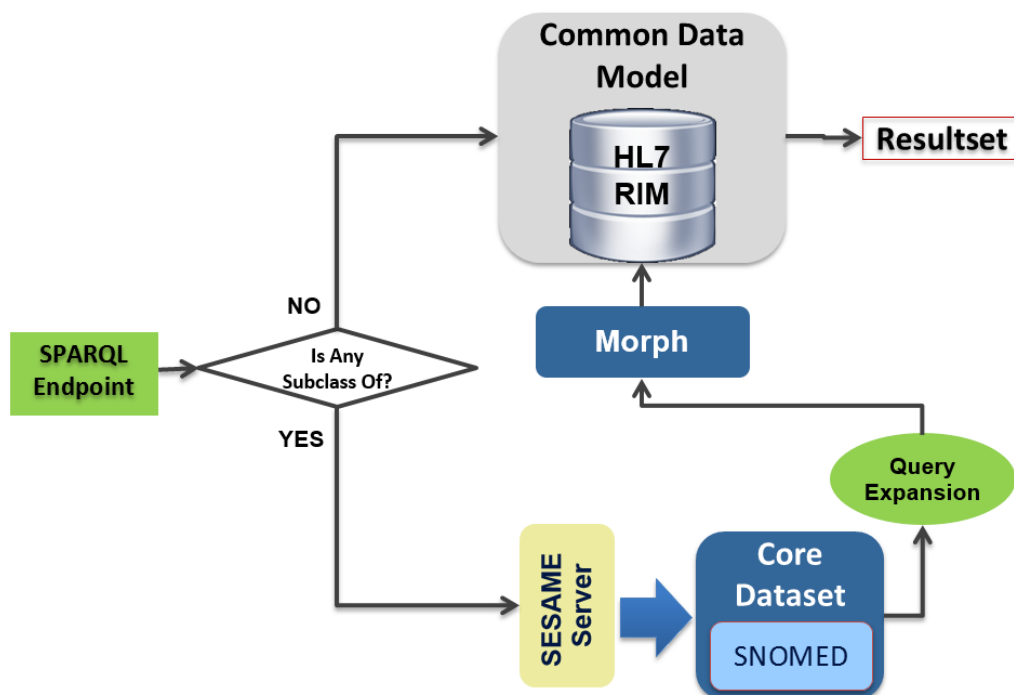


Figure 3: Query Execution

The *morph-RDB* engine interacts with the Query Execution Service whenever this service receives a SPARQL query. The Query Execution expands and applies other semantics extracted from the Core Dataset, and then sends the query to the *morph-RDB* to build the query for the MySQL Common Data Model.

The *morph* mapping can be found in Annex. This mapping enables the translations of a SPARQL query, to the MySQL query of the CDM that retrieves the results from the information stored. Finally a resultset is returned in MySQL language that is processed and returned as a SPARQL resultset.

2.4 Query Builder

In order to maintain coherence between how the data is normalized and stored into specific sections of the CDM, and how this data can be queried, a service that provides this coherence during data retrieval is required.

The Query Builder includes a Query Template Library, defined by a set of SPARQL query templates based on the CDM domain that is used to cover every possible expression stored in the CDM. There are five types of possible queries identified: Observation, Procedure, Substance Administration, Entities (devices, products, genes) and Demographic information. The Query Builder receives a concept from a terminology, this concept is normalized and is classified, and then, according to this information a SPARQL template is returned. The template returned contains a set of optional attributes to extract information regarding the patient from the attributes of the CDM (E.g. date when the act takes place, interpretations, etc.)

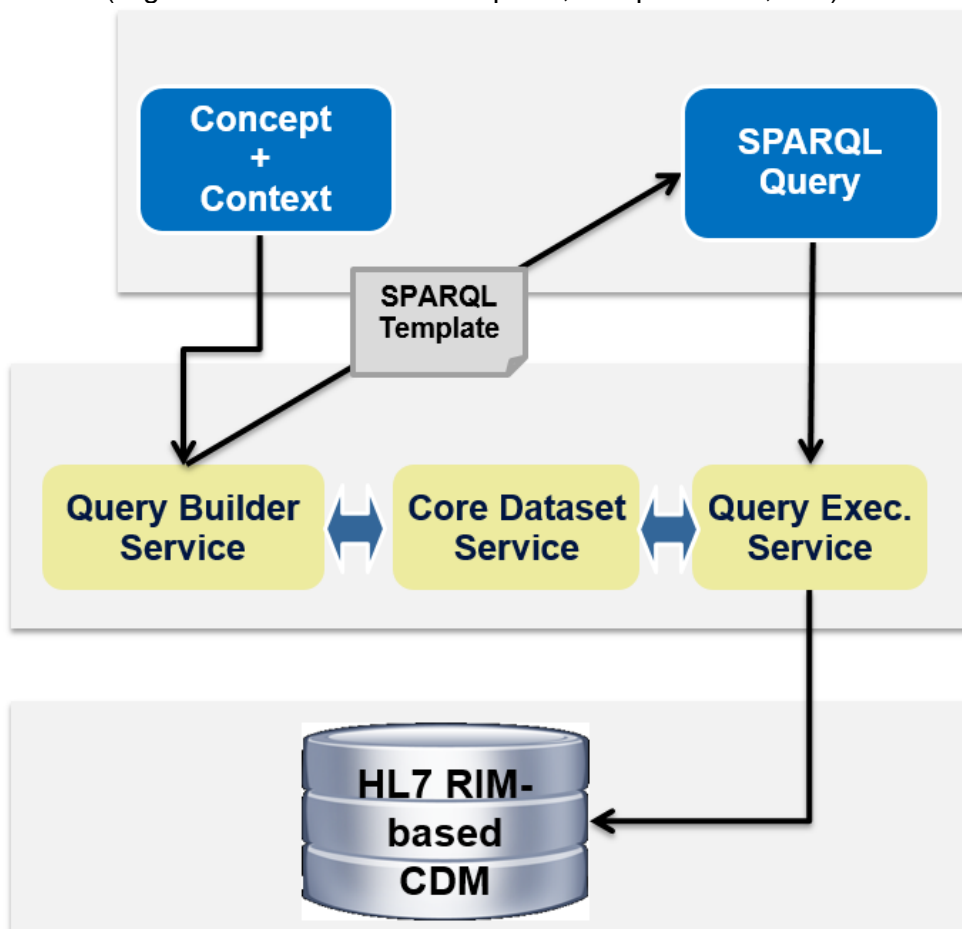


Figure 4: Semantic Interoperability Layer brief query process

As it has been mentioned at the beginning of this section, and as it will be described later in this deliverable (section 3 "Semantic Normalization Pipeline"), the approach followed by this process is the same as in the rest of the components of the semantic layer, in order to keep coherence and integrity. First, the SNOMED CT normal form is extracted from the concept, then, the elements extracted are mapped to objects from the CDM by the Terminology Binding, and finally, once all the information is homogenized, templates can be set to retrieve this information.

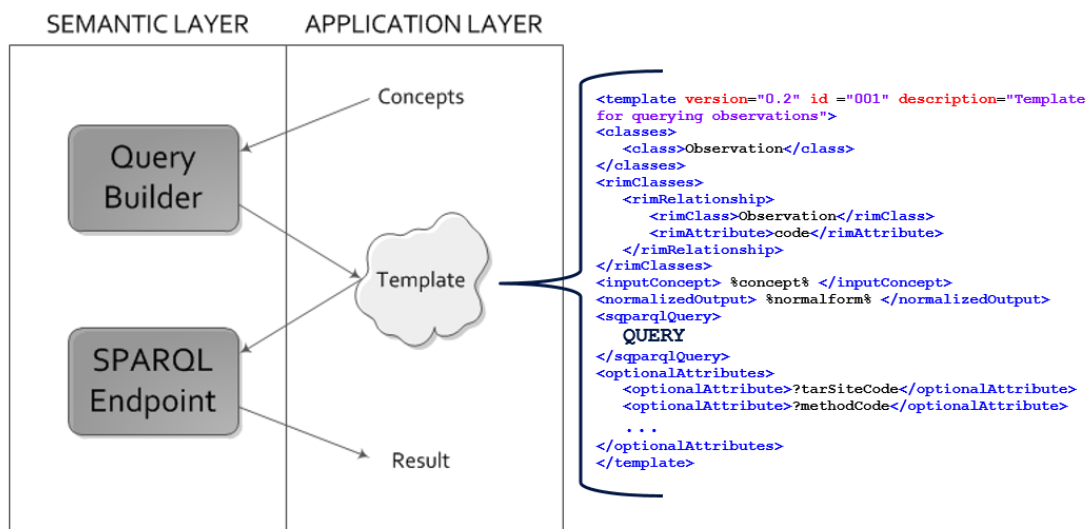


Figure 5: Query Builder example

As shown in the above figure, the Query Builder receives a concept for which information wants to be retrieved, and the service returns a template (taking into account the type of concept that it is and where it can be found at the CDM) adequate to query for all the information stored regarding the given concept. This template, once fulfilled, can be used directly at the SPARQL endpoint or through the Query Execution to finally retrieve the information that the user may be looking for.

2.4.1 Core Dataset Autocomplete Service

The main purpose of the Autocomplete Service is to provide matches of concepts from medical ontologies (SNOMED CT, HGNC and LOINC), from a set of strings typed by the user.

In the Figure 6 we can appreciate where the AutoComplete Service is placed within the Semantic Interoperability layer. As it is shown, this service makes use mainly of the Terminology Binding from the Core Dataset service, and as it will be described later on this section, also makes use of an internal database that collects information from the information stores at the CDM.

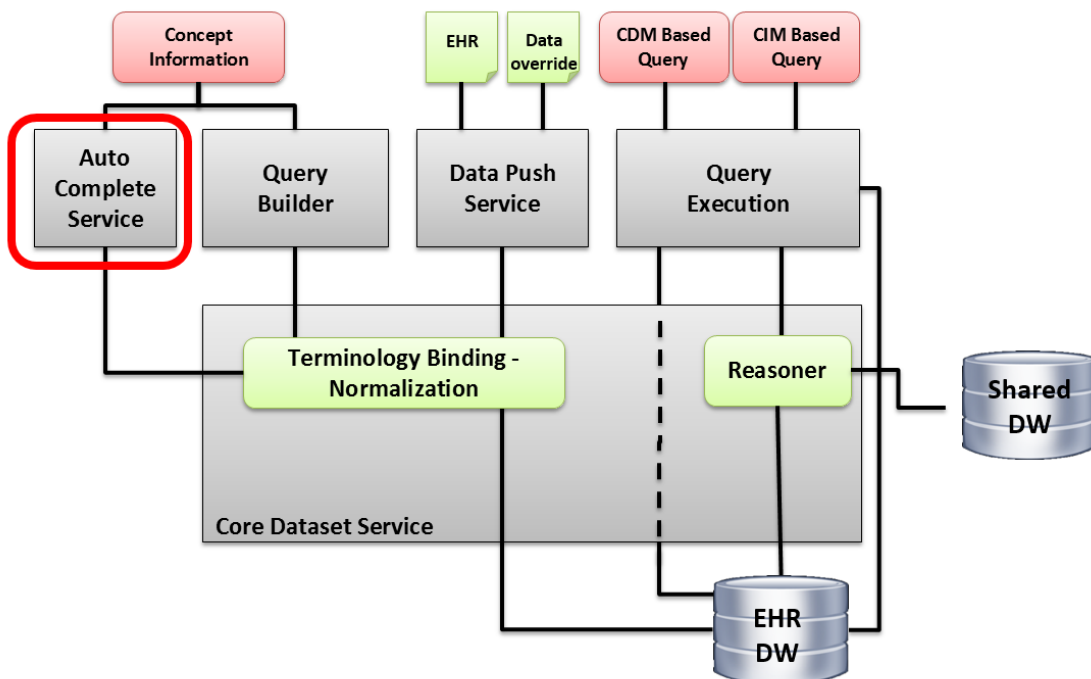


Figure 6: Overview of the AutoComplete within the Semantic Layer

Filtering capabilities are also provided in order to search for medical terms in a more accurate and efficient way. The filters are structured following a two-layer structure. Primary filters, at the first layer, follow the semantic structure of the RIM based CDM (described at the section 2.1.1 of the present deliverable). The secondary filters, at second layer, are the root concepts of the hierarchy of the SNOMED CT vocabulary, matching these secondary filters with the classification of these roots concepts to the different RIM classes of the CDM (Observation, Procedure, Substance Administration, Entity, etc).

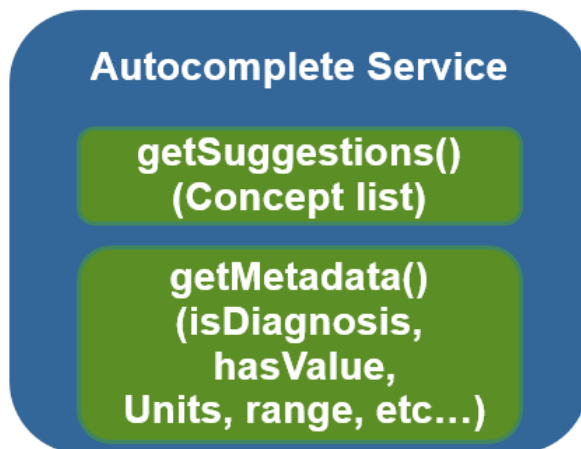


Figure 7: Methods provided by the Autocomplete Service

The Autocomplete Service provides information regarding the context of Core Dataset concepts, information that cannot be obtained from terminologies. The kind of information that is provided varies from concepts associated with a numerical value (e.g.

results of concentration of sodium in blood), their units and ranges, concepts that could be considered diagnosis, and finally, concepts with a different classification for a specific system.

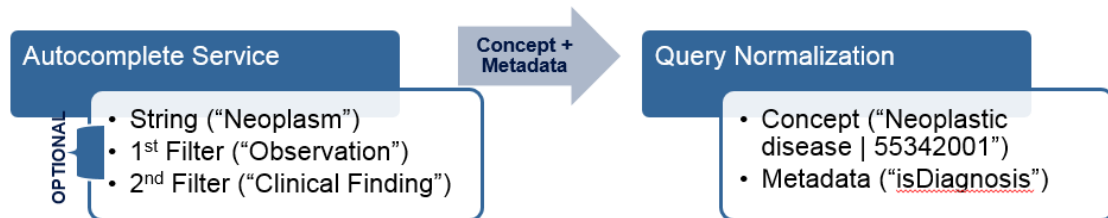


Figure 8: Example using Autocomplete Service

This component is essential in order to be able to obtain all the information required for the Query Normalization component. The inclusion of the Autocomplete Service and the Query Normalization to the current version of the semantic interoperability layer, aims to facilitate data retrieval from the CDM providing just a string, or a set of strings with the information in which the user is interested. Components required to abstract the user from the SPARQL syntax and the CDM structure.

3 SEMANTIC NORMALIZATION PIPELINE

Even following standards such as HL7 and SNOMED, in a complex area such as clinical research, it is common to find different representations of the same information within heterogeneous data. This ambiguity can be produced by the use of a different vocabulary, or by using different concepts of the same vocabulary that represent the same information [7]. For this reason, a process is required to transform data into a common standard in order to avoid different representations of the same information.

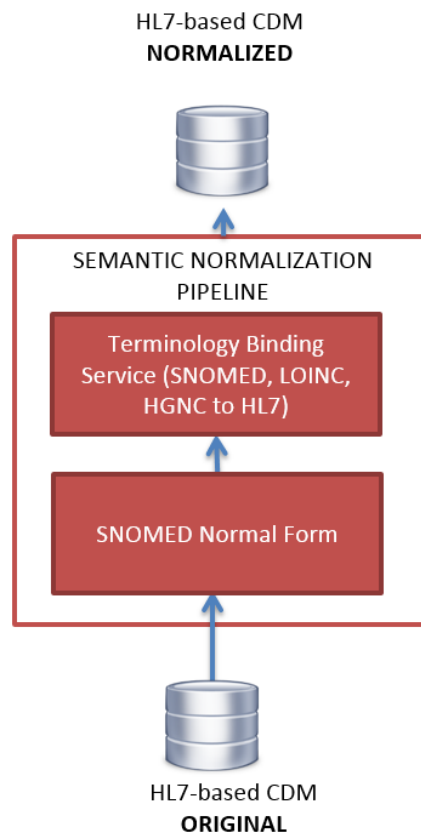


Figure 9: Normalization Pipeline

Once every concept is encoded using a concept from the Core Dataset, the normalization takes place and the concept is transformed into its normal form. This normal form is a standardized way of keeping all the information implicit in a concept, with the minimal amount of information loss. For example, the concept "CT of abdomen | 169070004", is a "CT of chest and abdomen" and a "Diagnostic radiography of abdomen", and is normalized into "Computed tomography imaging | 312251004" that takes place at the "Abdominal structure | 113345001". The previous normalized representation of the concept provides all the implicit information of the concept, so that this concept can be retrieved by querying "Computed tomography imaging" sited at the "Abdominal structure", and all the semantics of these concepts, instead of being limited by just the hierarchy of the original concept "CT of abdomen".

Finally, when the concept is normalized, the concept and all the concepts extracted from its normalization must be linked to objects from the CDM in order to be

stored. This final step is performed by the Terminology Binding, and it is supported by HL7 guidelines from the HL7 Terminology Project [5] and Principles Of Health Interoperability HL7 and SNOMED [5], where concepts are linked to fields of the HL7 RIM.

The different steps of the Semantic Normalization Pipeline are executed for every piece of information that is loaded through the Data Push Service, keeping the original information stored at a different database. While the normalized version is the default repository that final applications should query.

4 SECURITY INTEGRATION

The semantic interoperability layer processes/stores/transforms patient specific data in his components and is, in this way, subjected to the rules of the INTEGRATE legal framework. Special security processes and tools were developed and integrated as part of the INTEGRATE security framework to comply with the legal requirements.

Most of the security integration work was done for protecting the access to the data in the query execution service. Figure 10 gives an overview of the technical solution worked out for this component, which contains 6 security measures.

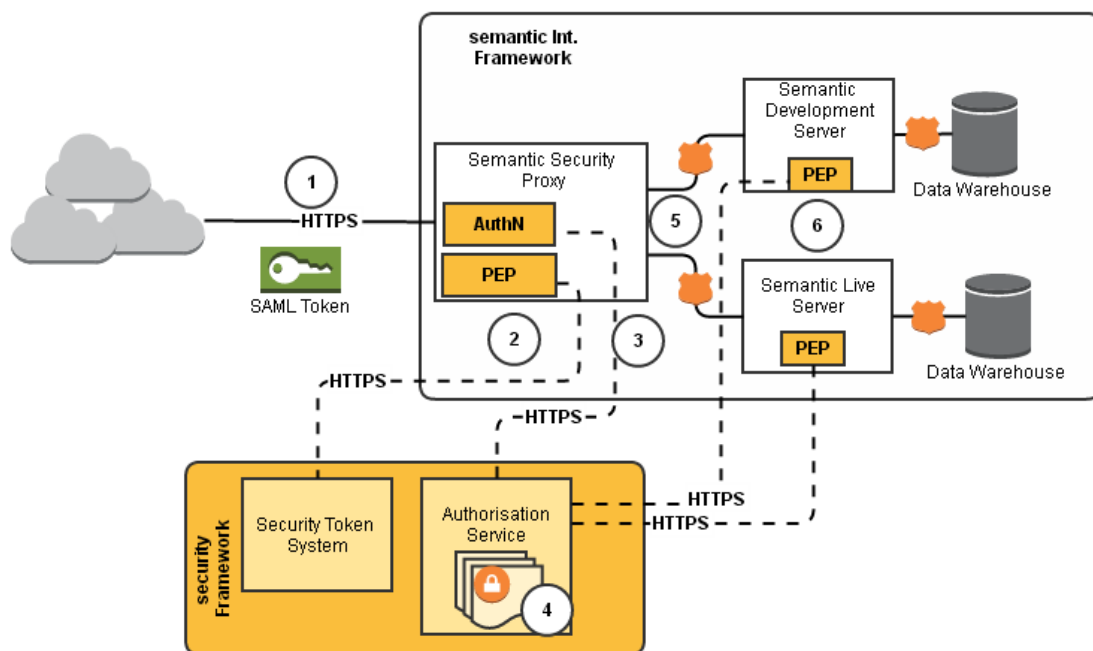


Figure 10: Query execution security integration

- 1) Each incoming data query request (sent over HTTPS) should contain a signed SAML token issued by an INTEGRATE identity provider. It contains attribute information (name, organisation, id) about the requester.
- 2) In the semantic security proxy, the SAML token will be validated (signature and timestamp checking). If the SAML token is found invalid, access will be rejected to the request.
- 3) Next, the policy enforcement point of the semantic security proxy will generate an access request including the attribute information about the requester. This request is sent to the central INTEGRATE authorisation service (over HTTPS).
- 4) The authorisation service will retrieve the request coming from the semantic security proxy and will generate an access decision, based on defined policies. These policies contain the access restrictions defined by the data owner. The authorisation service will return the access decision back to the semantic security proxy.

-
- 5) If access was granted, the query request will be forwarded to the internal semantic services (that are only accessible by the semantic security proxy). If access was denied, access will be rejected to the request.
 - 6) The semantic services contain an extra Policy Enforcement Point which will handle fine grained access control (access control on the internals of the datasets). The flow is similar as in step 3 and 4.

The other services in the semantic interoperability layer contain the same security processes and tools described in the steps above with exception of step 6.

5 SNAQL QUERY ENGINE

The SNAggletooth Query Language (SNAQL) is a Domain-Specific Language (DSL) developed in INTEGRATE to easily query patient data coming from the semantic interoperability layer. The objective was to allow the definition of queries in a language close to the natural language, while keeping the implementation effort to the minimum. The DSL leverages on the DSL-facilitating features of the Groovy¹ scripting language for offering advanced functionality (e.g. temporal query constructs) at minimal effort. These include operator overloading, closures, metaprogramming (Meta Object Protocol (MOP)) support and named parameter support through argument maps. An example of a SNAQL script can be found in Table 1.

Patients having received mastectomy and post-surgical irradiation for the primary diagnosis, who developed a distant relapse during the 5 first years after this surgery.

```
def MASTECTOMY='172043006'  
def MALIGNEOPLAS='128462008'  
def POSTOPRADIO='168525009'  
def condition = diagnosis of:MALIGNEOPLAS occurred  
                    after:{lastOccurrence of:{procedure  
                        of:MASTECTOMY}}, within: 5.years  
result = var{condition} and {procedure of:POSTOPRADIO}
```

Table 1: SNAQE Script Example

The SNAggletooth Query Engine (SNAQE) is responsible for executing SNAQL scripts. The engine is not an interpreter, but rather the definition of a context in which the SNAQL code makes sense. The context defines the classes and methods that represent the “medical concepts” and “functional” keywords of the DSL. Within this context, the SNAQL script is plain groovy code which instantiates objects (patient lists) and performs methods on them (the functions).

The SNAQE relies on two services of the semantic interoperability layer (see Figure 11) for mapping a SNAQL script to the underlying semantic query language (SPARQL).

¹ Groovy - <http://groovy.codehaus.org/>

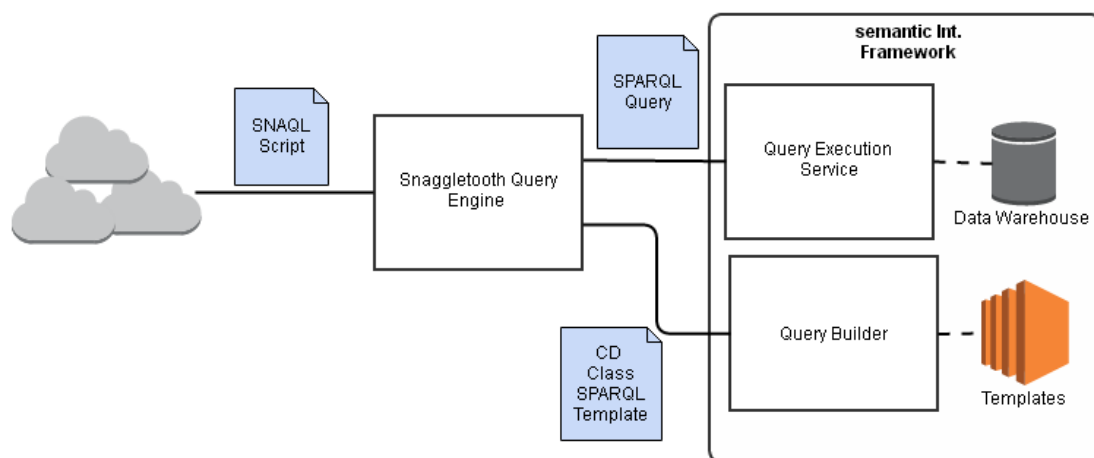


Figure 11: SNAQL - Semantic Interoperability Overview

- Medical concepts in a SNAQL script will be translated to the core dataset binding classes in the SNAQE internally. Each of these core dataset binding classes found will be sent to the **query builder service**, to receive the corresponding SPARQL template. This template will be filled in using the given parameters in SNAQL.
- When the SPARQL template is filled in by the SNAQE, it will be sent to the **query execution service**. The response, which is a cohort of patients, will be converted to a SNAgletooth Query Response (SNAQR).

6 CLINICAL TRIAL METADATA REPOSITORY UPDATE

A considerable amount of the clinical data that will be aggregated in the INTEGRATE clinical data warehouse will be collected in the context of a clinical trial (for instance to determine a patient's eligibility for a trial or to perform post-trial analysis). In the current practice, the trial context of the collected clinical data is often not explicitly defined. Clinical data is typically collected in separate datasets on a per-trial basis.

In INTEGRATE, clinical data is aggregated in the clinical data warehouse and exposed in a uniform manner via the semantic interoperability layer. As the clinical data is aggregated and not collected into separate sets, additional information needs to be introduced to maintain the relation between the clinical data and the (trial) context in which the clinical data was collected. The INTEGRATE platform provisions a clinical trial metadata repository for this purpose.

INTEGRATE's *D3.5 - Initial prototype of the semantic interoperability layer* describes how clinical data is related to the trial context maintained in the clinical trial metadata repository. Functionality wise, the approach hasn't changed since D3.5. Implementation wise, the clinical trial metadata repository is updated to conform to the next iteration of the INTEGRATE security fabric, supporting authorization besides authentication.

7 CONCLUSIONS

Along this deliverable, the final version of the prototype of the semantic interoperability layer has been described, as an update of the previous deliverable 3.5 [1]. There have been some changes and additions since the initial version of the semantic interoperability layer:

- Modifications in the Common Data Model to store specific information that was not expected at the initial state of the model
- Use of Morph instead of D2R for SPARQL to MySQL translation at the Query Execution due to performance reasons
- The Query Builder has been updated to be coherent with the normalization process of the data
- The Autocomplete Service facilitates the querying process by requiring just a word or phrase from which the user requires to extract related information
- The addition of the Normalization Pipeline to set the process that standardizes the information stored
- In order to comply the legal requirements from the INTEGRATE framework, security mechanisms have been integrated in the Query Execution Service.
- For a better understanding of developers during the query construction process, the SNAQL Query engine has been defined to keep these queries close to natural language

With these additions and modifications, issues commented at the conclusions of the deliverable 3.5 [1] has been addressed:

- Improve performance of D2R query system. Solved by the use of Morph.
- Introduce automatic data normalization within the ETL process. Solved by the inclusion of the Normalization Pipeline.
- Store two versions of the data source, original and the normalized. This is achieved by the Normalization Pipeline, which keeps the original data, and then normalizes and stores the data in a different database.

These improvements support a core component such as the Semantic Interoperability Layer, enhancing performance and semantic capabilities for the INTEGRATE project platform.

8 REFERENCES

- [1] Deliverable 3.5, "Initial prototype of the semantic interoperability layer".
- [2] Deliverable 2.6, "System architecture refinement, security framework and implementation status".
- [3] Deliverable 3.1, "Canonical models of CTMS and EHR systems".
- [4] Freddy Priyatna, Oscar Corcho, Juan Sequeda. Formalisation and Experiences of R2RML-based SPARQL to SQL query translation using Morph. World Wide Web Conference (WWW 2014).
- [5] HL7 TermInfo Project http://wiki.hl7.org/index.php?title=TermInfo_Project
- [6] Tim Benson. Principles of Health Interoperability HL7 and SNOMED. London: Springer-Verlag; 2010.
- [7] Andrews, James E., Rachel L. Richesson, and Jeffrey Krischer. "Variation of SNOMED CT coding of clinical research concepts among coding experts." *Journal of the American Medical Informatics Association* 14.4 (2007): 497-506.

A. APPENDIX

a. Mapping ttl (relational to non-relational)

```
@prefix rr: <http://www.w3.org/ns/r2rml#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix rev: <http://purl.org/stuff/rev#> .
@prefix hl7rim: <http://hl7rim.GIB-UPM.org/common-data-model#> .
@base <http://mappingpedia.org/rdb2rdf/r2rml/tc/> .

<TriplesMapAct>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "act" ];

rr:subjectMap [ rr:termType rr:IRI;
rr:template "http://localhost:2020/resource/actno/{id}";
rr:class hl7rim:act;
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:act_participation ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapParticipation>;
rr:joinCondition [ rr:child "id" ; rr:parent "actId" ; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:act_actObservationValues ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapActObservationValues>;
rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:act_actObservationInterpretationCode ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapActObservationInterpretationCode>;
rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:act_actRelationship ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapActRelationship>;
rr:joinCondition [ rr:child "id" ; rr:parent "idA" ; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:act_observation ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapObservation>;
rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:act_observationAct ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapObservationAct>;
rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:act_procedureAct ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapProcedureAct>;
rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:act_substanceAdministrationT ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapSubstanceAdministrationT>;
rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
];
];
```

```
};

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_actTargetSiteCode ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapActTargetSiteCode>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_actMethodCode ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapActMethodCode>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_actProcedureApproachSiteCode ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapActProcedureApproachSiteCode>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_procedure ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapProcedure>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_exposure ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapExposure>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_substanceAdministration ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapSubstanceAdministration>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_id ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "id" ; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_classCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "classCode" ; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_subClassCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "subClassCode" ; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_moodCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "moodCode" ; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_code ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "code" ; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_codeVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeVocId" ; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_codeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrig" ; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_codeOrigVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrigVocId" ; ];
];
```

```

];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_actionNegationInd ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "actionNegationInd"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_title ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "title"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_text ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "text"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_statusCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "statusCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_effectiveTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "effectiveTime_start"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_effectiveTime_start ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "effectiveTime_start"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_effectiveTime_end ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "effectiveTime_end"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_activityTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "activityTime"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_availabilityTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "availabilityTime"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_creationTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "creationTime"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_modificationTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "modificationTime"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:act_uncertaintyCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "uncertaintyCode"; ];
];

.

<TriplesMapActMethodCode>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "actmethodcode" ];

rr:subjectMap [ rr:termType rr:IRI;
  rr:template "http://localhost:2020/resource/actmethodcodeno/{id}";
  rr:class hl7rim:actMethodCode;
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actMethodCode_id ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "id"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actMethodCode_code ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "code"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actMethodCode_codeVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeVocId"; ];
];

```



```

];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actMethodCode_codeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrig"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actMethodCode_codeOrigVocld ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrigVocld"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actMethodCode_title ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "title"; ];
];

.

<TriplesMapProcedureAct>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "procedures" ];

rr:subjectMap [ rr:termType rr:IRI;
  rr:template "http://localhost:2020/resource/procactno/{id}";
  rr:class hl7rim:procedureAct;
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedureAct_act ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapAct>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedureAct_substanceAdministration ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapSubstanceAdministration>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedureAct_substanceAdministrationT ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapSubstanceAdministrationT>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedureAct_actProcedureApproachSiteCode ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapActProcedureApproachSiteCode>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedureAct_actMethodCode ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapActMethodCode>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedureAct_actTargetSiteCode ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapActTargetSiteCode>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

.

<TriplesMapProcedure>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "v_procedure" ];

rr:subjectMap [ rr:termType rr:IRI;
  rr:template "http://localhost:2020/resource/procedureno/{id}";
  rr:class hl7rim:procedure;
];

```

```
rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_act ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapAct>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_substanceAdministration ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapSubstanceAdministration>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_id ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "id" ; ]
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_clinicalTrialId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "clinicalTrialId" ; ]
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_classCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "classCode" ; ]
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_subClassCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "subClassCode" ; ]
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_moodCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "moodCode" ; ]
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_code ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "code" ; ]
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_codeVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeVocId" ; ]
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_codeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrig" ; ]
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_codeOrigVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrigVocId" ; ]
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_actionNegationInd ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "actionNegationInd" ; ]
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_title ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "title" ; ]
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_text ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "text" ; ]
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_statusCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "statusCode" ; ]
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_effectiveTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "effectiveTime_start" ; ]
];

rr:predicateObjectMap [
```

```
rr:predicateMap [ rr:constant hl7rim:procedure_effectiveTime_start ;
rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "effectiveTime_start"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_effectiveTime_end;
rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "effectiveTime_end"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_activityTime;
rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "activityTime"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_availabilityTime;
rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "availabilityTime"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_uncertaintyCode;
rr:objectMap [ rr:termType rr:Literal; rr:column "uncertaintyCode"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_approachSiteCode;
rr:objectMap [ rr:termType rr:Literal; rr:column "approachSiteCode"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_approachSiteCodeTitle;
rr:objectMap [ rr:termType rr:Literal; rr:column "approachSiteCodeTitle"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_approachSiteCodeVocId;
rr:objectMap [ rr:termType rr:Literal; rr:column "approachSiteCodeVocId"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_approachSiteCodeOrig;
rr:objectMap [ rr:termType rr:Literal; rr:column "approachSiteCodeOrig"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_approachSiteCodeOrigVocId;
rr:objectMap [ rr:termType rr:Literal; rr:column "approachSiteCodeOrigVocId"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_methodCode;
rr:objectMap [ rr:termType rr:Literal; rr:column "methodCode"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_methodCodeTitle;
rr:objectMap [ rr:termType rr:Literal; rr:column "methodCodeTitle"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_methodCodeVocId;
rr:objectMap [ rr:termType rr:Literal; rr:column "methodCodeVocId"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_methodCodeOrig;
rr:objectMap [ rr:termType rr:Literal; rr:column "methodCodeOrig"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_methodCodeOrigVocId;
rr:objectMap [ rr:termType rr:Literal; rr:column "methodCodeOrigVocId"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_targetSiteCode;
rr:objectMap [ rr:termType rr:Literal; rr:column "targetSiteCode"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_targetSiteCodeTitle;
rr:objectMap [ rr:termType rr:Literal; rr:column "targetSiteCodeTitle"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:procedure_targetSiteCodeVocId;
rr:objectMap [ rr:termType rr:Literal; rr:column "targetSiteCodeVocId"; ];
];
```

```

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_targetSiteCodeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "targetSiteCodeOrig"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:procedure_targetSiteCodeOrigVocld ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "targetSiteCodeOrigVocld"; ];
];

.

<TriplesMapSubstanceAdministrationT>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "substanceadministration" ];

rr:subjectMap [ rr:termType rr:IRI;
  rr:template "http://localhost:2020/resource/substanceadministrationno/{id}";
  rr:class hl7rim:substanceAdministrationT;
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_actMethodCode ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapActMethodCode>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_actTargetSiteCode ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapActTargetSiteCode>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_actProcedureApproachSiteCode ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapActProcedureApproachSiteCode>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

  rr:predicateObjectMap [
    rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_id ];
    rr:objectMap [ rr:termType rr:Literal; rr:column "id" ; ];
  ];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_routeCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "routeCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_routeCodeVocld ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "routeCodeVocld"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_routeCodeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "routeCodeOrig"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_routeCodeOrigVocld ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "routeCodeOrigVocld"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_routeCodeTitle ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "routeCodeTitle"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_doseQuantity ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "doseQuantity"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_doseQuantityUnits ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "doseQuantityUnits"; ];
];

```

```

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_rateQuantity ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "rateQuantity"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_rateQuantityUnits ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "rateQuantityUnits"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_doseCheckQuantity ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "doseCheckQuantity"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_doseCheckQuantityUnits ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "doseCheckQuantityUnits"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_periodIntervalTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "periodIntervalTime"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministrationT_periodIntervalUnits ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "periodIntervalUnits"; ];
];

.

<TriplesMapSubstanceAdministration>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "v_substanceadministration" ];

rr:subjectMap [ rr:termType rr:IRI;
  rr:template "http://localhost:2020/resource/substanceadministrationno/{id}";
  rr:class hl7rim:substanceAdministration;
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_procedure ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapProcedure>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_participation ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapParticipation>;
    rr:joinCondition [ rr:child "id" ; rr:parent "actId" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_id ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "id"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_routeCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "routeCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_routeCodeVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "routeCodeVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_routeCodeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "routeCodeOrig"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_routeCodeOrigVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "routeCodeOrigVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_routeCodeTitle ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "routeCodeTitle"; ];
];

```

```
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_doseQuantity ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "doseQuantity"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_doseQuantityUnits ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "doseQuantityUnits"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_rateQuantity ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "rateQuantity"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_rateQuantityUnits ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "rateQuantityUnits"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_doseCheckQuantity ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "doseCheckQuantity"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_doseCheckQuantityUnits ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "doseCheckQuantityUnits"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_clinicalTrialId ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "clinicalTrialId"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_classCode ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "classCode"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_subClassCode ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "subClassCode"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_moodCode ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "moodCode"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_code ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "code"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_codeVocId ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "codeVocId"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_codeOrig ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrig"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_codeOrigVocId ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrigVocId"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_actionNegationInd ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "actionNegationInd"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_title ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "title"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_text ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "text"; ];  
  ];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_statusCode ;  
    rr:objectMap [ rr:termType rr:Literal; rr:column "statusCode"; ];  
  ];
```

```
};

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_effectiveTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "effectiveTime_start"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_effectiveTime_start ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "effectiveTime_start"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_effectiveTime_end ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "effectiveTime_end"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_activityTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "activityTime"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_availabilityTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "availabilityTime"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_uncertaintyCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "uncertaintyCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_approachSiteCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "approachSiteCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_approachSiteCodeTitle ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "approachSiteCodeTitle"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_approachSiteCodeVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "approachSiteCodeVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_approachSiteCodeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "approachSiteCodeOrig"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_approachSiteCodeOrigVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "approachSiteCodeOrigVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_methodCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "methodCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_methodCodeTitle ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "methodCodeTitle"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_methodCodeVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "methodCodeVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_methodCodeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "methodCodeOrig"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_methodCodeOrigVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "methodCodeOrigVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:substanceAdministration_targetSiteCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "targetSiteCode"; ];
];

rr:predicateObjectMap [
```

```

rr:predicateMap [ rr:constant hl7rim:substanceAdministration_targetSiteCodeTitle ];
rr:objectMap [ rr:termType rr:Literal; rr:column "targetSiteCodeTitle"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:substanceAdministration_targetSiteCodeVocId ];
rr:objectMap [ rr:termType rr:Literal; rr:column "targetSiteCodeVocId"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:substanceAdministration_targetSiteCodeOrig ];
rr:objectMap [ rr:termType rr:Literal; rr:column "targetSiteCodeOrig"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:substanceAdministration_targetSiteCodeOrigVocId ];
rr:objectMap [ rr:termType rr:Literal; rr:column "targetSiteCodeOrigVocId"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:substanceAdministration_periodIntervalTime ];
rr:objectMap [ rr:termType rr:Literal; rr:column "periodIntervalTime"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:substanceAdministration_periodIntervalUnits ];
rr:objectMap [ rr:termType rr:Literal; rr:column "periodIntervalUnits"; ];
];
.
<TriplesMapObservationAct>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "observation" ];

rr:subjectMap [ rr:termType rr:IRI;
rr:template "http://localhost:2020/resource/obsactno/{id}";
rr:class hl7rim:observationAct;
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:observationAct_act ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapAct>;
rr:joinCondition [ rr:child "id"; rr:parent "id"; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:observationAct_actObservationInterpretationCode ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapActObservationInterpretationCode>;
rr:joinCondition [ rr:child "id"; rr:parent "id"; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:observationAct_participation ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapParticipation>;
rr:joinCondition [ rr:child "id"; rr:parent "actId"; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:observationAct_actObservationValues ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapActObservationValues>;
rr:joinCondition [ rr:child "id"; rr:parent "id"; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:observationAct_actMethodCode ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapActMethodCode>;
rr:joinCondition [ rr:child "id"; rr:parent "id"; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:observationAct_actTargetSiteCode ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapActTargetSiteCode>;
rr:joinCondition [ rr:child "id"; rr:parent "id"; ]
];
];

```



```

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observationAct_id ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "id"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observationAct_valueNegationInd ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "valueNegationInd"; ];
];

.

<TriplesMapObservation>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "v_observation" ];

rr:subjectMap [ rr:termType rr:IRI;
  rr:template "http://localhost:2020/resource/obsno/{id}";
  rr:class hl7rim:observation;
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_act ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapAct>;
    rr:joinCondition [ rr:child "id"; rr:parent "id"; ];
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_participation ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapParticipation>;
    rr:joinCondition [ rr:child "id"; rr:parent "actId"; ];
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_actObservationValues ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapActObservationValues>;
    rr:joinCondition [ rr:child "id"; rr:parent "id"; ];
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_id ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "id"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_classCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "classCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_subClassCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "subClassCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_moodCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "moodCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_code ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "code"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_codeVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_codeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrig"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_codeOrigVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrigVocId"; ];
];

```

```
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_actionNegationInd ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "actionNegationInd"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_title ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "title"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_text ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "text"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_statusCode ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "statusCode"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_effectiveTime ];  
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "effectiveTime_start"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_effectiveTime_start ];  
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "effectiveTime_start"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_effectiveTime_end ];  
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "effectiveTime_end"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_activityTime ];  
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "activityTime"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_availabilityTime ];  
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "availabilityTime"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_creationTime ];  
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "creationTime"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_modificationTime ];  
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "modificationTime"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_valueNegationInd ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "valueNegationInd"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_uncertaintyCode ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "uncertaintyCode"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_clinicalTrialId ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "clinicalTrialId"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_value ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "value"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_valueType ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "valueType"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_units ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "units"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:observation_valueCode ];
```

```
rr:objectMap [ rr:termType rr:Literal; rr:column "valueCode"; ];
};

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_valueCodeVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "valueCodeVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_valueCodeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "valueCodeOrig"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_valueCodeOrigVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "valueCodeOrigVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_valueTitle ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "valueTitle"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_refRangeMin ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "referenceRangeMin"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_refRangeMax ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "referenceRangeMax"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_refRange ];
  rr:objectMap [ rr:termType rr:Literal; rr:template "[{referenceRangeMin} - {referenceRangeMax}"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_fullValue ];
  rr:objectMap [ rr:termType rr:Literal; rr:template "{value} - {units} - [{referenceRangeMin} - {referenceRangeMax}]"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_interpretationCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "interpretationCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_interpretationCodeTitle ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "interpretationCodeTitle"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_interpretationCodeVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "interpretationCodeVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_interpretationCodeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "interpretationCodeOrig"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_interpretationCodeOrigVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "interpretationCodeOrigVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_methodCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "methodCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_methodCodeTitle ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "methodCodeTitle"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_methodCodeVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "methodCodeVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_methodCodeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "methodCodeOrig"; ];
];
```

```

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_methodCodeOrigVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "methodCodeOrigVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_targetSiteCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "targetSiteCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_targetSiteCodeTitle ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "targetSiteCodeTitle"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_targetSiteCodeVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "targetSiteCodeVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_targetSiteCodeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "targetSiteCodeOrig"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:observation_targetSiteCodeOrigVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "targetSiteCodeOrigVocId"; ];
];

.

<TriplesMapActObservationValues>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "actobservationvalues" ];

rr:subjectMap [ rr:termType rr:IRI;
  rr:template "http://localhost:2020/resource/actobservationvaluesno/{id}";
  rr:class hl7rim:actObservationValues;
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationValues_observation ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapObservation>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationValues_actObservationInterpretationCode ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapActObservationInterpretationCode>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationValues_id ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "id"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationValues_control ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "control"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationValues_value ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "value"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationValues_valueType ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "valueType"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationValues_units ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "units"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationValues_refRangeMin ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "referenceRangeMin"; ];
];

```

```

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationValues_refRangeMax ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "referenceRangeMax"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationValues_code ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "code"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationValues_codeVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationValues_codeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrig"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationValues_codeOrigVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrigVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationValues_title ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "title"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationValues_refRange ];
  rr:objectMap [ rr:termType rr:Literal; rr:template "[{referenceRangeMin} - {referenceRangeMax}"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationValues_fullValue ];
  rr:objectMap [ rr:termType rr:Literal; rr:template "{value} - {units} - {referenceRangeMin} - {referenceRangeMax}"; ];
];

<TriplesMapActProcedureApproachSiteCode>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "actprocedureapproachsitecode" ];

rr:subjectMap [ rr:termType rr:IR;
  rr:template "http://localhost:2020/resource/actprocedureapproachsitecodeno/{id}";
  rr:class hl7rim:actProcedureApproachSiteCode;
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actProcedureApproachSiteCode_id ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "id"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actProcedureApproachSiteCode_code ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "code"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actProcedureApproachSiteCode_codeVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actProcedureApproachSiteCode_codeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrig"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actProcedureApproachSiteCode_codeOrigVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrigVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actProcedureApproachSiteCode_title ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "title"; ];
];

<TriplesMapActObservationInterpretationCode>
a rr:TriplesMap;

```

```

rr:logicalTable [ rr:tableName "actobservationinterpretationcode" ];

rr:subjectMap [ rr:termType rr:IRI;
  rr:template "http://localhost:2020/resource/actobservationinterpretationcodeno/{id}";
  rr:class hl7rim:actObservationInterpretationCode;
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationInterpretationCode_id ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "id"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationInterpretationCode_code ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "code"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationInterpretationCode_codeVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationInterpretationCode_codeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrig"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationInterpretationCode_codeOrigVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrigVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actObservationInterpretationCode_title ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "title"; ];
];
.

<TriplesMapActRelationship>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "actrelationship" ];

rr:subjectMap [ rr:termType rr:IRI;
  rr:template "http://localhost:2020/resource/actrelationshipno/{idA}/{idB}/{typeCode}";
  rr:class hl7rim:actRelationship;
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actRelationship_actA ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapAct>;
    rr:joinCondition [ rr:child "idA"; rr:parent "id"; ];
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actRelationship_actB ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapAct>;
    rr:joinCondition [ rr:child "idB"; rr:parent "id"; ];
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actRelationship_idA ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "idA"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actRelationship_idB ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "idB"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:actRelationship_typeCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "typeCode"; ];
];
.

<TriplesMapActTargetSiteCode>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "acttargetsitecode" ];

```

```

rr:subjectMap [ rr:termType rr:IR;
rr:template "http://localhost:2020/resource/acttargetsitecodeno/{id}";
rr:class hl7rim:actTargetSiteCode;
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:actTargetSiteCode_id ];
rr:objectMap [ rr:termType rr:Literal; rr:column "id"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:actTargetSiteCode_code ];
rr:objectMap [ rr:termType rr:Literal; rr:column "code"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:actTargetSiteCode_codeVocId ];
rr:objectMap [ rr:termType rr:Literal; rr:column "codeVocId"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:actTargetSiteCode_codeOrig ];
rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrig"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:actTargetSiteCode_codeOrigVocId ];
rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrigVocId"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:actTargetSiteCode_title ];
rr:objectMap [ rr:termType rr:Literal; rr:column "title"; ];
];

.

<TriplesMapParticipation>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "participation" ];

rr:subjectMap [ rr:termType rr:IR;
rr:template "http://localhost:2020/resource/participationno/{actId}/{entityId}/{roleId}";
rr:class hl7rim:participation;
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:participation_act ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapAct>;
rr:joinCondition [ rr:child "actId"; rr:parent "id"; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:participation_role ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapRole>;
rr:joinCondition [ rr:child "roleId"; rr:parent "id"; ];
rr:joinCondition [ rr:child "entityId"; rr:parent "entityId"; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:participation_observation ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapObservation>;
rr:joinCondition [ rr:child "actId"; rr:parent "id"; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:participation_entity ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapEntity>;
rr:joinCondition [ rr:child "entityId"; rr:parent "id"; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:participation_entityId ];
rr:objectMap [ rr:termType rr:Literal; rr:column "entityId"; ];
];

```

```

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:participation_roleId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "roleId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:participation_actId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "actId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:participation_typeCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "typeCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:participation_functionCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "functionCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:participation_negationInd ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "negationInd"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:participation_noteText ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "noteText"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:participation_time_start ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "time_start"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:participation_time_end ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "time_end"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:participation_modeCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "modeCode"; ];
];

.

<TriplesMapPerson>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "v_person" ];

rr:subjectMap [ rr:termType rr:IRI;
  rr:template "http://localhost:2020/resource/personno/{id}";
  rr:class hl7rim:person;
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_livingSubject ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapLivingSubject>;
    rr:joinCondition [ rr:child "id" ; rr:parent "id" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_participation ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapParticipation>;
    rr:joinCondition [ rr:child "id" ; rr:parent "entityId" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_role ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapRole>;
    rr:joinCondition [ rr:child "id" ; rr:parent "entityId" ; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_id ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "id"; ];
];

rr:predicateObjectMap [

```



```
rr:predicateMap [ rr:constant hl7rim:person_raceCode ];
rr:objectMap [ rr:termType rr:Literal; rr:column "raceCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_classCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "classCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_determinerCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "determinerCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_code ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "code"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_codeVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_codeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrig"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_codeOrigVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrigVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_title ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "title"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_quantity ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "quantity"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_name ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "name"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_desc ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "desc"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_statusCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "statusCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_administrativeGenderCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "administrativeGenderCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_creationTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "creationTime"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_modificationTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "modificationTime"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_birthTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "birthTime"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_birthTimeTolerance ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "birthTimeTolerance"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_deceasedInd ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "deceasedInd"; ];
];
```

```

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_deceasedTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "deceasedTime"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_personName ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "personName"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:person_personSurname ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "personSurname"; ];
];

<TriplesMapRole>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "role" ];

rr:subjectMap [ rr:termType rr:IR;
  rr:template "http://localhost:2020/resource/roleno/{entityId}/id";
  rr:class hl7rim:role;
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_entity ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapEntity>;
    rr:joinCondition [ rr:child "entityId"; rr:parent "id"; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_participation ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapParticipation>;
    rr:joinCondition [ rr:child "id"; rr:parent "roleId"; ];
    rr:joinCondition [ rr:child "entityId"; rr:parent "entityId"; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_person ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapPerson>;
    rr:joinCondition [ rr:child "entityId"; rr:parent "id"; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_entityId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "entityId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_id ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "id"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_classCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "classCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_code ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "code"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_codeVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeVocId"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_codeOrig ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrig"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_codeOrigVocId ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrigVocId"; ];
];

```

```

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_title ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "title"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_negationInd ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "negationInd"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_name ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "name"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_statusCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "statusCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_effectiveTime_start ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "effectiveTime_start"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_effectiveTime_end ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "effectiveTime_end"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_effectiveTimeIvl ];
  rr:objectMap [ rr:termType rr:Literal; rr:template "{effectiveTime_start} - {effectiveTime_end}"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:role_confidentialityCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "confidentialityCode"; ];
];

<TriplesMapEntity>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "entity" ];

rr:subjectMap [ rr:termType rr:IRI;
  rr:template "http://localhost:2020/resource/entityno/{id}";
  rr:class hl7rim:entity;
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:entity_livingSubject ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapLivingSubject>;
    rr:joinCondition [ rr:child "id"; rr:parent "id"; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:entity_role ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapRole>;
    rr:joinCondition [ rr:child "id"; rr:parent "entityId"; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:entity_participation ];
  rr:objectMap [
    rr:parentTriplesMap <TriplesMapParticipation>;
    rr:joinCondition [ rr:child "id"; rr:parent "entityId"; ]
  ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:entity_id ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "id"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:entity_classCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "classCode"; ];
];

rr:predicateObjectMap [

```

```

rr:predicateMap [ rr:constant hl7rim:entity_determinerCode ];
rr:objectMap [ rr:termType rr:Literal; rr:column "determinerCode"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:entity_code ];
rr:objectMap [ rr:termType rr:Literal; rr:column "code"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:entity_codeVocId ];
rr:objectMap [ rr:termType rr:Literal; rr:column "codeVocId"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:entity_codeOrig ];
rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrig"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:entity_codeOrigVocId ];
rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrigVocId"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:entity_title ];
rr:objectMap [ rr:termType rr:Literal; rr:column "title"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:entity_quantity ];
rr:objectMap [ rr:termType rr:Literal; rr:column "quantity"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:entity_name ];
rr:objectMap [ rr:termType rr:Literal; rr:column "name"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:entity_desc ];
rr:objectMap [ rr:termType rr:Literal; rr:column "desc"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:entity_statusCode ];
rr:objectMap [ rr:termType rr:Literal; rr:column "statusCode"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:entity_creationTime ];
rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "creationTime"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:entity_modificationTime ];
rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "modificationTime"; ];
];

<TriplesMapExposure>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "v_exposure" ];

rr:subjectMap [ rr:termType rr:IRI;
rr:template "http://localhost:2020/resource/exposureno/{id}";
rr:class hl7rim:exposure;
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:exposure_act ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapAct>;
rr:joinCondition [ rr:child "id"; rr:parent "id"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:exposure_id ];
rr:objectMap [ rr:termType rr:Literal; rr:column "id"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:exposure_routeCode ];
rr:objectMap [ rr:termType rr:Literal; rr:column "routeCode"; ];

```

```
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_exposureLevel ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "exposureLevel"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_exposureModeCode ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "exposureModeCode"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_clinicalTrialId ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "clinicalTrialId"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_classCode ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "classCode"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_subClassCode ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "subClassCode"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_moodCode ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "moodCode"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_code ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "code"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_codeVocId ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeVocId"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_codeOrig ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrig"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_codeOrigVocId ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrigVocId"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_actionNegationInd ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "actionNegationInd"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_title ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "title"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_text ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "text"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_statusCode ];  
  rr:objectMap [ rr:termType rr:Literal; rr:column "statusCode"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_effectiveTime_start ];  
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "effectiveTime_start"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_effectiveTime_end ];  
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "effectiveTime_end"; ];  
];  
  
rr:predicateObjectMap [  
  rr:predicateMap [ rr:constant hl7rim:exposure_activityTime ];  
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "activityTime"; ];  
];  
  
rr:predicateObjectMap [  

```

```

rr:predicateMap [ rr:constant hl7rim:exposure_availabilityTime ];
rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "availabilityTime"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:exposure_uncertaintyCode ];
rr:objectMap [ rr:termType rr:Literal; rr:column "uncertaintyCode"; ];
];

.

<TriplesMapLivingSubject>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "v_livingsubject" ];

rr:subjectMap [ rr:termType rr:IRI;
rr:template "http://localhost:2020/resource/livingsubjectno/{id}";
rr:class hl7rim:livingSubject;
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:livingSubject_entity ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapEntity>;
rr:joinCondition [ rr:child "id"; rr:parent "id"; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:livingSubject_person ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapPerson>;
rr:joinCondition [ rr:child "id"; rr:parent "id"; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:livingSubject_role ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapRole>;
rr:joinCondition [ rr:child "id"; rr:parent "entityId"; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:livingSubject_participation ];
rr:objectMap [
rr:parentTriplesMap <TriplesMapParticipation>;
rr:joinCondition [ rr:child "id"; rr:parent "entityId"; ]
];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:livingSubject_id ];
rr:objectMap [ rr:termType rr:Literal; rr:column "id"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:livingSubject_classCode ];
rr:objectMap [ rr:termType rr:Literal; rr:column "classCode"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:livingSubject_determinerCode ];
rr:objectMap [ rr:termType rr:Literal; rr:column "determinerCode"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:livingSubject_code ];
rr:objectMap [ rr:termType rr:Literal; rr:column "code"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:livingSubject_codeVocId ];
rr:objectMap [ rr:termType rr:Literal; rr:column "codeVocId"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:livingSubject_codeOrig ];
rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrig"; ];
];

rr:predicateObjectMap [
rr:predicateMap [ rr:constant hl7rim:livingSubject_codeOrigVocId ];
rr:objectMap [ rr:termType rr:Literal; rr:column "codeOrigVocId"; ];
];

```

```

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:livingSubject_quantity ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "quantity"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:livingSubject_name ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "name"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:livingSubject_desc ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "desc"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:livingSubject_statusCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "statusCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:livingSubject_administrativeGenderCode ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "administrativeGenderCode"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:livingSubject_creationTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "creationTime"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:livingSubject_modificationTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "modificationTime"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:livingSubject_birthTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "birthTime"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:livingSubject_birthTimeTolerance ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "birthTimeTolerance"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:livingSubject_deceasedInd ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "deceasedInd"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:livingSubject_deceasedTime ];
  rr:objectMap [ rr:termType rr:Literal; rr:datatype xsd:dateTime; rr:column "deceasedTime"; ];
];

.

<TriplesMapSecurity>
a rr:TriplesMap;

rr:logicalTable [ rr:tableName "security" ];

rr:subjectMap [ rr:termType rr:IR;
  rr:template "http://localhost:2020/resource/securityno/{securityVar}/{dataset}";
  rr:class hl7rim:security;
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:security_securityVar ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "securityVar"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:security_dataset ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "dataset"; ];
];

rr:predicateObjectMap [
  rr:predicateMap [ rr:constant hl7rim:security_decision ];
  rr:objectMap [ rr:termType rr:Literal; rr:column "decision"; ];
];

.

```

b. Query template library

i. Observation template

Grid view of the XML

The screenshot shows an XML editor interface. At the top, there is a table with the following content:

@version	1.00
@id	002
@description	Template for querying observations

Below the table, the XML structure is shown in a tree view:

- <>template
 - <>templateClass Observation
 - <>rimClasses
 - <>rimRelationship
 - <>rimClass Observation
 - <>rimAttribute code
 - <>inputConcept %concept%
 - <>normalizedOutput %normalform%
 - <>sparqlQuery


```
#cdata-section
SELECT DISTINCT ?id ?code ?patientId ?birthTime ?effectiveTime $$optionalAttributes$$
WHERE {
  ?instPerson      hl7rim:person_id ?patientId;
                  hl7rim:person_code '337915000';
                  hl7rim:person_birthTime ?birthTime;
                  hl7rim:person_participation ?instPart2.
  ?instPart2      hl7rim:participation_act ?instAct.
  ?instAct        hl7rim:act_code ?code;
                  hl7rim:act_id ?id.
  OPTIONAL{?instAct  hl7rim:act_effectiveTime ?effectiveTime}

  FILTER (?code IN (isAnySubclassOf(%Observation_code%)))
  $$OptionalStructures$$
}
```

Grid view of the Optionals

<> optionals		
OptionalStructure (26)		
@ id	<> OptionalHeaderAttributes	<> OptionalFilter
<> classCode	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> subclassCode	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> moodCode	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> codeVocId	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> codeOrig	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> codeOrigVocId	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> actionNegationInd	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> title	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> text	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> statusCode	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> activityTime	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> availabilityTime	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> uncertaintyCode	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> valueFilter	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> valueFilterOptional	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> interpretationCode	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> interpretationCodeOptional	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> methodCode	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> methodCodeOptional	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> targetSiteCode	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> targetSiteCodeOptional	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> entity	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> entityOptional	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> relationshipFilter	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> patientFilter	▶ OptionalHeaderAttributes	▶ OptionalFilter
<> void	▶ OptionalHeaderAttributes	▶ OptionalFilter

Optional SPARQL block

entity	OptionalHeaderAttributes	<div style="border: 1px solid black; padding: 5px;"> <div style="border-bottom: 1px solid black; margin-bottom: 5px;"> OptionalFilter <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 10%;">@ id</td><td>entity</td></tr> <tr><td>@ tag</td><td>\$\$OptionalStructures\$\$</td></tr> </table> </div> <div style="border-bottom: 1px solid black; margin-bottom: 5px;"> queryFilter <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td colspan="2" style="text-align: right;">filterAttributes</td></tr> <tr><td colspan="2" style="text-align: right;">attribute</td></tr> <tr><td style="width: 10%;">@ id</td><td>value</td></tr> <tr><td>@ tag</td><td>%Entity_code%</td></tr> </table> </div> <div style="border-bottom: 1px solid black; margin-bottom: 5px;"> queryRetrieval <div style="border: 1px solid black; padding: 5px; margin-top: 5px;"> <pre>#cdata-section ?instAct h7rim:act_participation ?instPart. ?instPart h7rim:participation_entity ?instEntity. ?instEntity h7rim:entity_id ?entityId; h7rim:entity_classCode ?entityClassCode; h7rim:entity_determinerCode ?entityDeterminerCode; h7rim:entity_code ?entityCode; h7rim:entity_title ?entityTitle. FILTER (?entityCode != '337915000') \$\$OptionalStructures\$\$</pre> </div> </div> </div>	@ id	entity	@ tag	\$\$OptionalStructures\$\$	filterAttributes		attribute		@ id	value	@ tag	%Entity_code%
@ id	entity													
@ tag	\$\$OptionalStructures\$\$													
filterAttributes														
attribute														
@ id	value													
@ tag	%Entity_code%													

ii. Procedure template

Grid view of the XML

The screenshot shows a grid view of an XML template. The grid is organized into several sections:

- <> template**: A table with the following rows:

@version	1.00
@id	004
@description	Template for querying procedures
- <> templateClass**: Procedure
- <> rimClasses**: A nested grid containing:
 - <> rimRelationship**: A nested grid containing:
 - <> rimClass**: Procedure
 - <> rimAttribute**: code
- <> inputConcept**: %concept%
- <> normalizedOutput**: %normalform%
- <> sparqlQuery**: A text area containing a SPARQL query:


```
#cdata-section
SELECT DISTINCT ?id ?code ?patientId ?birthTime ?effectiveTime $$optionalAttributes$$
WHERE {
  ?instPerson      hl7rim:person_id ?patientId;
                  hl7rim:person_code '337915000';
                  hl7rim:person_birthTime ?birthTime;
                  hl7rim:person_participation ?instPart2.
  ?instPart2      hl7rim:participation_act ?instAct.
  ?instAct        hl7rim:act_code ?code;
                  hl7rim:act_id ?id.
  OPTIONAL{?instAct  hl7rim:act_effectiveTime ?effectiveTime}

  FILTER (?code IN (isAnySubclassOf(%Procedure_code%)))
  $$OptionalStructures$$
}
```
- <> optionals**: A collapsed section at the bottom.

Grid view of the optionals

<> optionals		
OptionalStructure (23)		
@ id	<> OptionalHeaderAttributes	<> OptionalFilter
<> classCode	OptionalHeaderAttributes	OptionalFilter
<> subclassCode	OptionalHeaderAttributes	OptionalFilter
<> moodCode	OptionalHeaderAttributes	OptionalFilter
<> codeVocId	OptionalHeaderAttributes	OptionalFilter
<> codeOrig	OptionalHeaderAttributes	OptionalFilter
<> codeOrigVocId	OptionalHeaderAttributes	OptionalFilter
<> actionNegationInd	OptionalHeaderAttributes	OptionalFilter
<> title	OptionalHeaderAttributes	OptionalFilter
<> text	OptionalHeaderAttributes	OptionalFilter
<> statusCode	OptionalHeaderAttributes	OptionalFilter
<> activityTime	OptionalHeaderAttributes	OptionalFilter
<> availabilityTime	OptionalHeaderAttributes	OptionalFilter
<> uncertaintyCode	OptionalHeaderAttributes	OptionalFilter
<> approachSiteCode	OptionalHeaderAttributes	OptionalFilter
<> methodCode	OptionalHeaderAttributes	OptionalFilter
<> methodCodeOptional	OptionalHeaderAttributes	OptionalFilter
<> targetSiteCode	OptionalHeaderAttributes	OptionalFilter
<> targetSiteCodeOptional	OptionalHeaderAttributes	OptionalFilter
<> entity	OptionalHeaderAttributes	OptionalFilter
<> entityOptional	OptionalHeaderAttributes	OptionalFilter
<> relationshipFilter	OptionalHeaderAttributes	OptionalFilter
<> patientFilter	OptionalHeaderAttributes	OptionalFilter
<> void	OptionalHeaderAttributes	OptionalFilter

Grid view of an optional

OptionalStructure (23)		
@ id	<> OptionalHeaderAttributes	<> OptionalFilter
classCode	OptionalHeaderAttributes	OptionalFilter @ id classCode @ tag \$\$OptionalStructures\$\$ <> queryFilter \$\$OptionalStructures\$\$ <> filterAttributes <> queryRetrieval #cdata-section OPTIONAL{?instAct hl7rim:act_classCode ?classCode} \$\$OptionalStructures\$\$

iii. Substance Administration template

Grid view of the XML

The screenshot displays a software interface for editing an XML template. The main content is organized into several sections:

- Metadata Table:** A table with three rows:

@version	1.00
@id	005
@description	Template for querying administrations of substance
- XML Structure Tree:** A tree view showing the hierarchy of the XML template:
 - <> template
 - <> templateClass SubstanceAdministration
 - <> rimClasses
 - <> rimRelationship
 - <> rimClass SubstanceAdministration
 - <> rimAttribute code

- Input Concept:** A text field containing the value `%concept%`.
- Normalized Output:** A text field containing the value `%normalform%`.
- SPARQL Query:** A text area containing the following query:


```
#cdata-section
SELECT DISTINCT ?id ?code ?patientId ?birthTime ?effectiveTime_start ?effectiveTime_end
$$optionalAttributes$$
WHERE {
  ?instPerson      hl7rim:person_id ?patientId;
                   hl7rim:person_code '337915000';
                   hl7rim:person_birthTime ?birthTime;
                   hl7rim:person_participation ?instPart2.
  ?instPart2      hl7rim:participation_act ?instAct.
  ?instAct        hl7rim:act_code ?code;
                   hl7rim:act_id ?id.
  OPTIONAL{?instAct  hl7rim:act_effectiveTime_start ?effectiveTime_start}
  OPTIONAL{?instAct  hl7rim:act_effectiveTime_end ?effectiveTime_end}

  FILTER (?code IN (isAnySubclassOf(%SubstanceAdministration_code%)))
  $$OptionalStructures$$
}
```
- Options:** A section at the bottom with a right-pointing arrow and the label `<> options`.

Grid view of the options

<> optionals			
OptionalStructure (24)			
@ id	<> OptionalHeaderAttributes	<> OptionalFilter	
<> classCode	OptionalHeaderAttributes	OptionalFilter	
<> subclassCode	OptionalHeaderAttributes	OptionalFilter	
<> moodCode	OptionalHeaderAttributes	OptionalFilter	
<> codeVocId	OptionalHeaderAttributes	OptionalFilter	
<> codeOrig	OptionalHeaderAttributes	OptionalFilter	
<> codeOrigVocId	OptionalHeaderAttributes	OptionalFilter	
<> actionNegationInd	OptionalHeaderAttributes	OptionalFilter	
<> title	OptionalHeaderAttributes	OptionalFilter	
<> text	OptionalHeaderAttributes	OptionalFilter	
<> statusCode	OptionalHeaderAttributes	OptionalFilter	
<> activityTime	OptionalHeaderAttributes	OptionalFilter	
<> availabilityTime	OptionalHeaderAttributes	OptionalFilter	
<> uncertaintyCode	OptionalHeaderAttributes	OptionalFilter	
<> substanceAdministration	OptionalHeaderAttributes	OptionalFilter	
<> targetSiteCode	OptionalHeaderAttributes	OptionalFilter	
<> targetSiteCodeOptional	OptionalHeaderAttributes	OptionalFilter	
<> methodCode	OptionalHeaderAttributes	OptionalFilter	
<> methodCodeOptional	OptionalHeaderAttributes	OptionalFilter	
<> approachSiteCode	OptionalHeaderAttributes	OptionalFilter	
<> entity	OptionalHeaderAttributes	OptionalFilter	
<> entityOptional	OptionalHeaderAttributes	OptionalFilter	
<> relationshipFilter	OptionalHeaderAttributes	OptionalFilter	
<> patientFilter	OptionalHeaderAttributes	OptionalFilter	
<> void	OptionalHeaderAttributes	OptionalFilter	

Grid view of an optional

statusCode	▶ OptionalHeaderAttributes	◀ OptionalFilter										
		<table border="1"> <tr> <td>@ id</td> <td>statusCode</td> </tr> <tr> <td>@ tag</td> <td>\$\$OptionalStructures\$\$</td> </tr> <tr> <td><> queryFilter</td> <td>\$\$OptionalStructures\$\$</td> </tr> <tr> <td><> filterAttributes</td> <td></td> </tr> <tr> <td><> queryRetrieval</td> <td> <pre> #CDATA-section OPTIONAL {?instAct hl7rim:act_statusCode ?statusCode} \$\$OptionalStructures\$\$ </pre> </td> </tr> </table>	@ id	statusCode	@ tag	\$\$OptionalStructures\$\$	<> queryFilter	\$\$OptionalStructures\$\$	<> filterAttributes		<> queryRetrieval	<pre> #CDATA-section OPTIONAL {?instAct hl7rim:act_statusCode ?statusCode} \$\$OptionalStructures\$\$ </pre>
@ id	statusCode											
@ tag	\$\$OptionalStructures\$\$											
<> queryFilter	\$\$OptionalStructures\$\$											
<> filterAttributes												
<> queryRetrieval	<pre> #CDATA-section OPTIONAL {?instAct hl7rim:act_statusCode ?statusCode} \$\$OptionalStructures\$\$ </pre>											

iv. Diagnosis template

Grid view of the XML

The screenshot displays a software interface for defining a template. It is organized into several sections:

- template**: A table with the following data:

@version	1.00
@id	002
@description	Template for querying diagnosis
- templateClass**: Diagnosis
- rimClasses**: A nested structure containing:
 - rimRelationship**: Contains a **rimClass** (Diagnosis) and a **rimAttribute** (code).
- inputConcept**: %concept%
- normalizedOutput**: %normalform%
- sparqlQuery**: A SPARQL query within a `#cdata-section` container:


```
SELECT DISTINCT ?id ?code ?patientId ?birthTime ?effectiveTime $$optionalAttributes$$
WHERE {
  ?instPerson      hl7rim:person_id ?patientId;
                  hl7rim:person_code '337915000';
                  hl7rim:person_birthTime ?birthTime;
                  hl7rim:person_participation ?instPart2.
  ?instPart2      hl7rim:participation_act ?instAct.
  ?instAct        hl7rim:act_code ?code;
                  hl7rim:act_subClassCode 'DIAG';
                  hl7rim:act_id ?id.
  OPTIONAL{?instAct  hl7rim:act_effectiveTime ?effectiveTime}

  FILTER (?code IN (isAnySubclassOf(%Diagnosis_code%)))
  $$OptionalStructures$$
}
```
- optionals**: A section at the bottom for defining optional structures.

Grid view of the options

<> optionals			
OptionalStructure (26)			
@ id	<> OptionalHeaderAttributes	<> OptionalFilter	
<> classCode	OptionalHeaderAttributes	OptionalFilter	
<> subclassCode	OptionalHeaderAttributes	OptionalFilter	
<> moodCode	OptionalHeaderAttributes	OptionalFilter	
<> codeVocId	OptionalHeaderAttributes	OptionalFilter	
<> codeOrig	OptionalHeaderAttributes	OptionalFilter	
<> codeOrigVocId	OptionalHeaderAttributes	OptionalFilter	
<> actionNegationInd	OptionalHeaderAttributes	OptionalFilter	
<> title	OptionalHeaderAttributes	OptionalFilter	
<> text	OptionalHeaderAttributes	OptionalFilter	
<> statusCode	OptionalHeaderAttributes	OptionalFilter	
<> activityTime	OptionalHeaderAttributes	OptionalFilter	
<> availabilityTime	OptionalHeaderAttributes	OptionalFilter	
<> uncertaintyCode	OptionalHeaderAttributes	OptionalFilter	
<> valueFilter	OptionalHeaderAttributes	OptionalFilter	
<> valueFilterOptional	OptionalHeaderAttributes	OptionalFilter	
<> interpretationCode	OptionalHeaderAttributes	OptionalFilter	
<> interpretationCodeOptional	OptionalHeaderAttributes	OptionalFilter	
<> methodCode	OptionalHeaderAttributes	OptionalFilter	
<> methodCodeOptional	OptionalHeaderAttributes	OptionalFilter	
<> targetSiteCode	OptionalHeaderAttributes	OptionalFilter	
<> targetSiteCodeOptional	OptionalHeaderAttributes	OptionalFilter	
<> entity	OptionalHeaderAttributes	OptionalFilter	
<> entityOptional	OptionalHeaderAttributes	OptionalFilter	
<> relationshipFilter	OptionalHeaderAttributes	OptionalFilter	
<> patientFilter	OptionalHeaderAttributes	OptionalFilter	
<> void	OptionalHeaderAttributes	OptionalFilter	

Grid view of an optional

codeOrig	OptionalHeaderAttributes	OptionalFilter
----------	--------------------------	----------------

```

@ id codeOrig
@ tag $$OptionalStructures$$

<> queryFilter $$OptionalStructures$$
<> filterAttributes
<> queryRetrieval
  #cdata-section
  OPTIONAL {?instAct
  hl7rim:act_codeOrig ?codeOrig}
  $$OptionalStructures$$
  
```

v. *Person template*

Grid view of the XML

@version	1.00
@id	001
@description	Template for querying patients

```

<templateClass> Person
<rimClasses>
  <rimRelationship>
    <rimClass> Person
    <rimAttribute> code
</rimClasses>
<inputConcept> %concept%
<normalizedOutput> %normalform%
<sparqlQuery>
  #cdata-section
  SELECT DISTINCT ?patientId ?birthTime $$optionalAttributes$$
  WHERE
  {
    ?instLiSu      hl7rim:livingSubject_id ?patientId;
                  hl7rim:livingSubject_birthTime ?birthTime.
  }
  $$OptionalStructures$$
</sparqlQuery>
<options>
  
```

Grid view of the options

<> optionals

OptionalStructure (9)		
@ id	<> OptionalHeaderAttributes	<> OptionalFilter
<> patientDeterminerCode	OptionalHeaderAttributes	OptionalFilter
<> gender	OptionalHeaderAttributes	OptionalFilter
<> birthTimeTolerance	OptionalHeaderAttributes	OptionalFilter
<> deceasedTime	OptionalHeaderAttributes	OptionalFilter
<> patientFilter	OptionalHeaderAttributes	OptionalFilter
<> birthTimeFilter	OptionalHeaderAttributes	OptionalFilter
<> enrolledInTrial	OptionalHeaderAttributes	OptionalFilter
<> enrolledInTrialArm	OptionalHeaderAttributes	OptionalFilter
<> void	OptionalHeaderAttributes	OptionalFilter

Grid view of an optional

<> optionals

OptionalStructure (9)		
@ id	<> OptionalHeaderAttributes	<> OptionalFilter
<> patientDeterminerCode	OptionalHeaderAttributes	OptionalFilter
<> gender	OptionalHeaderAttributes	OptionalFilter
<> birthTimeTolerance	OptionalHeaderAttributes	OptionalFilter
<> deceasedTime	OptionalHeaderAttributes	OptionalFilter
<> patientFilter	OptionalHeaderAttributes	OptionalFilter
<> birthTimeFilter	OptionalHeaderAttributes	OptionalFilter
<> enrolledInTrial	OptionalHeaderAttributes	OptionalFilter
<> enrolledInTrialArm	OptionalHeaderAttributes	OptionalFilter
<> void	OptionalHeaderAttributes	OptionalFilter

<> queryFilter

```

#xdata-section
?instLSu          hl7rim:livingSubject_participation ?instPart.
?instPart        hl7rim:participation_actId ?id.
?actRel          hl7rim:actRelationship_idA ?id;
                 hl7rim:actRelationship_idB ?idB;
                 hl7rim:actRelationship_typeCode 'belongTo'.
?actRel2         hl7rim:actRelationship_idA ?idB;
                 hl7rim:actRelationship_idB ?idC;
                 hl7rim:actRelationship_typeCode 'belongTo'.
FILTER(?idC IN ("$$comparedValue$$"))
$$OptionalStructures$$

```

<> filterAttributes

<> queryRetrieval \$\$OptionalStructures\$\$

vi. Entity template

Grid view of the XML

<> template

@ version	1.00
@ id	003
@ description	Template for querying entities

<> templateClass Entity

<> rimClasses

<> rimRelationship

<> rimClass Entity

<> rimAttribute code

<> inputConcept %concept%

<> normalizedOutput %normalform%

<> sparqlQuery

```

#cddata-section
SELECT DISTINCT ?id ?code ?patientId ?birthTime ?effectiveTime $$optionalAttributes$$
WHERE {
  ?instPerson      hl7rim:person_id ?patientId;
                  hl7rim:person_code '337915000';
                  hl7rim:person_birthTime ?birthTime;
                  hl7rim:person_participation ?instPart2.
  ?instPart2      hl7rim:participation_act ?instAct.
  ?instAct        hl7rim:act_code ?code;
                  hl7rim:act_id ?id.
  OPTIONAL{?instAct  hl7rim:act_effectiveTime ?effectiveTime}
  ?instAct        hl7rim:act_participation ?instPart.
  ?instPart       hl7rim:participation_entity ?instEntity.
  ?instEntity     hl7rim:entity_code ?entityCode.

  FILTER (?entityCode IN (isAnySubclassOf(%Entity_code%)))
  $$OptionalStructures$$
}
          
```

<> options

Grid view of the options

<> options

OptionalStructure (21)

@ id	<> OptionalHeaderAttributes	<> OptionalFilter
<> partType	OptionalHeaderAttributes	OptionalFilter
<> entityId	OptionalHeaderAttributes	OptionalFilter
<> entityType	OptionalHeaderAttributes	OptionalFilter
<> classCode	OptionalHeaderAttributes	OptionalFilter
<> subclassCode	OptionalHeaderAttributes	OptionalFilter
<> moodCode	OptionalHeaderAttributes	OptionalFilter
<> codeVocId	OptionalHeaderAttributes	OptionalFilter
<> codeOrig	OptionalHeaderAttributes	OptionalFilter
<> codeOrigVocId	OptionalHeaderAttributes	OptionalFilter
<> actionNegationInd	OptionalHeaderAttributes	OptionalFilter
<> title	OptionalHeaderAttributes	OptionalFilter
<> text	OptionalHeaderAttributes	OptionalFilter
<> statusCode	OptionalHeaderAttributes	OptionalFilter
<> activityTime	OptionalHeaderAttributes	OptionalFilter
<> availabilityTime	OptionalHeaderAttributes	OptionalFilter
<> uncertaintyCode	OptionalHeaderAttributes	OptionalFilter
<> role	OptionalHeaderAttributes	OptionalFilter
<> entity	OptionalHeaderAttributes	OptionalFilter
<> entityOptionals	OptionalHeaderAttributes	OptionalFilter
<> patientFilter	OptionalHeaderAttributes	OptionalFilter
<> void	OptionalHeaderAttributes	OptionalFilter

Grid view of an optional

entityOptionals

OptionalHeaderAttributes

OptionalFilter

@ id entityOptionals

@ tag \$\$OptionalStructures\$\$

<> queryFilter \$\$OptionalStructures\$\$

<> filterAttributes

<> queryRetrieval

```

#CDATA-section
OPTIONAL {?instEntity hl7rim:entity_classCode ?entityClassCode.}
OPTIONAL {?instEntity hl7rim:entity_determinerCode ?entityDeterminerCode.}
OPTIONAL {?instEntity hl7rim:entity_codeVocId ?entityCodeVocId.}
OPTIONAL {?instEntity hl7rim:entity_codeOrig ?entityCodeOrig.}
OPTIONAL {?instEntity hl7rim:entity_codeOrigVocId ?entityCodeOrigVocId.}
OPTIONAL {?instEntity hl7rim:entity_title ?entityTitle.}
OPTIONAL {?instEntity hl7rim:entity_name ?entityName. }.
$$OptionalStructures$$
  
```