PAPER I

Marco-Ruiz L, Moner D, Maldonado JA, Kolstrup N, Bellika JG. Archetype-based data warehouse environment to enable the reuse of electronic health record data. International Journal of Medical Informatics 2015;84:702–14. doi:10.1016/j.ijmedinf.2015.05.016.

International Journal of Medical Informatics 84 (2015) 702–714

Contents lists available at ScienceDirect

International Journal of Medical Informatics

journal homepage: www.ijmijournal.com

Archetype-based data warehouse environment to enable the reuse of electronic health record data

Luis Marco-Ruiz[a,b,*], David Moner[c], José A. Maldonado[c,d], Nils Kolstrup[a,e], Johan G. Bellika[a,b]

[a] Norwegian Centre for Integrated Care and Telemedicine, University Hospital of North Norway, Norway
[b] Department of Clinical Medicine, Faculty of Health Sciences, UIT The Arctic University of Norway, Norway
[c] Instituto de Aplicaciones de las Tecnologías de la Información y de las Comunicaciones Avanzadas, Universitat Politècnica de València, Valencia, Spain
[d] VeraTech for Health SL, Valencia, Spain
[e] General Practice Research Unit, UIT The Arctic University of Norway, Norway

ARTICLE INFO

Article history:
Received 24 November 2014
Received in revised form 26 May 2015
Accepted 28 May 2015

Keywords:
Data reuse
Semantic interoperability
Electronic health record
openEHR
Data warehouse

ABSTRACT

Background: The reuse of data captured during health care delivery is essential to satisfy the demands of clinical research and clinical decision support systems. A main barrier for the reuse is the existence of legacy formats of data and the high granularity of it when stored in an electronic health record (EHR) system. Thus, we need mechanisms to standardize, aggregate, and query data concealed in the EHRs, to allow their reuse whenever they are needed.
Objective: To create a data warehouse infrastructure using archetype-based technologies, standards and query languages to enable the interoperability needed for data reuse.
Materials and methods: The work presented makes use of best of breed archetype-based data transformation and storage technologies to create a workflow for the modeling, extraction, transformation and load of EHR proprietary data into standardized data repositories. We converted legacy data and performed patient-centered aggregations via archetype-based transformations. Later, specific purpose aggregations were performed at a query level for particular use cases.
Results: Laboratory test results of a population of 230,000 patients belonging to Troms and Finnmark counties in Norway requested between January 2013 and November 2014 have been standardized. Test records normalization has been performed by defining transformation and aggregation functions between the laboratory records and an archetype. These mappings were used to automatically generate open EHR compliant data. These data were loaded into an archetype-based data warehouse. Once loaded, we defined indicators linked to the data in the warehouse to monitor test activity of Salmonella and Pertussis using the archetype query language.
Discussion: Archetype-based standards and technologies can be used to create a data warehouse environment that enables data from EHR systems to be reused in clinical research and decision support systems. With this approach, existing EHR data becomes available in a standardized and interoperable format, thus opening a world of possibilities toward semantic or concept-based reuse, query and communication of clinical data.

1. Background

The reuse of data inside the electronic health record (EHR) is considered of paramount importance to support clinical research [1,2] and clinical decision support (CDS) [2,3] systems. This reuse needs clinical data to flow from the systems where it was captured to a specialized system that processes it with different objectives, other than clinical care. Agile aggregation during this flow of information can enable the creation of large repositories containing samples to perform inferences over populations [4]. Many different actors like physicians, hospitals, pharmaceutical and biotech companies can benefit from this data flow to deliver their services more effectively [5].

* Corresponding author at: Norwegian Centre for Integrated Care and Telemedicine, University hospital of North Norway, P.O. Box 35, N-9038 Tromsø, Norway. Tel.: +47 41441113.
E-mail address: Luis.Marco.Ruiz@telemed.no (L. Marco-Ruiz).

http://dx.doi.org/10.1016/j.ijmedinf.2015.05.016
1386-5056/

Mechanisms to improve clinical accessibility, better integration of care across settings and advances in information exchange are needed to reuse data effectively [6]. Several initiatives have contributed with advances in those directions [7–9]. However, challenges in semantic interoperability and technical infrastructure are still present [10].

1.1. Semantic interoperability

On the semantic interoperability side the challenges are: (a) the integration and harmonization of the formats among different sources [2]; and (b) the definition of shared clinical information models, and their terminological binding. To harmonize and integrate EHR data, the adoption of EHR standards such as HL7CDA [11], openEHR [12] or EN ISO 13606 [13] is needed to exchange health information extracts. The use of such standards provides a common syntax and representation of clinical data. In addition, the definition of clinical information models that represent domain-oriented data structures allows the implementation of solutions for specific clinical use-cases. Finally, the terminology binding of clinical data and clinical information models provides semantics to the standardized information. The use of terminologies such as SNOMED-CT, LOINC or ICD delivers a common vocabulary used to specify the exact meaning of clinical data despite cultural or language differences. These actions are recognized as essential steps towards the semantic interoperability of health information [14]. In this sense, the memorandum of understanding signed between the Department of Health and Human Services and the European Commission stated the immediate importance in adopting international standards and interoperability specifications for EHRs. Also, the definition of harmonized clinical information models is of importance to achieve semantic interoperability across health information systems [15]. Internationally, the clinical information modelling initiative (CIMI) [16] is a not-for-profit community of users and stakeholders working in the definition of shared clinical information models. At the European level, the epSOS project [17] has defined the minimal infrastructure and clinical information models to communicate the patient summary, prescription and dispensation documents across Europe using the HL7 CDA standard. National initiatives like the NHS interoperability toolkit [18] or the meaningful use regulations [19,20] are powering the adoption of standards providing interoperability specifications and establishing payment for health professionals who adopt certified EHRs [4].

1.2. Linkage of the EHR with inference models

On the infrastructure side, the "impedance mismatch" [21] present between the information model, (where EHRs lay) and the inference model (where CDS systems and data mining are located) [22] can jeopardize clinical data reuse. The reason is that clinical information services are not structured to support ad hoc queries to allow reuse; therefore the use of data warehouses (DW), registers and repositories is needed [23]. To overcome the mismatch between the EHR and inference models several approaches have been proposed with application both in CDS and clinical research. The Arden syntax [24] was the first standard to create independent medical logic modules (MLM). Its syntax separated knowledge expressions from data access to EHR. However, it encapsulated data access between curly braces inside the MLM leading to the "curly braces problem". The former can be explained as the necessity to adapt the data access sections when moving the system among different production environments. GELLO [25] advanced in the direction of decoupling the EHR data access from the CDS system allowing to work with object models. Other computer interpretable guidelines formalisms like SAGE or GLIF used a summary view of the EHR, a.k.a. virtual medical record (VMR) using HL7 RIM [26]. Peleg et al. [27], following the former approach, defined a RIM VMR but added a mapping ontology to transform EHR granular data from the VMR into highly aggregated concepts for CDS [27]. Kawamoto et al. contributed by identifying the features needed to create a CDS VMR specific standard [28] to solve some limitations of the HL7 RIM based views. With regard to archetypes, Marcos et al. [29] and Fernandez-Breis et al. [30] relied on archetypes to generate an EHR view. Over that view they used additional layers of archetypes to increase the level of abstraction to generate the concepts needed by a patient cohort identification system for clinical research. Hybrid approaches to take advantage of the best feature of each standard have also been proposed. For example, in MobiGuide [31], Gonzalez-Ferrer et al. [32] propose to use HL7 RIM for backend systems and define a layer of archetypes compliant with the HL7 VMR for integration with the CDS system [32]. More oriented to the data reuse in research the SHARPn consortium has defined a complete pipeline to extract data from the EHR, normalize it into clinical element models and allow its reuse through health quality measures format (HQMF) [33].

1.3. Data reuse for CDS in North Norway primary care

Several studies have documented the benefits of CDS systems in primary care [3,34]. When applied to laboratory interventions, CDS systems have been documented to reduce costs avoiding redundant tests [35]. However, population information for general practitioners (GPs) is usually limited by the patients they are assigned and their personal communications with colleagues. They seldom have access to real time population test results or colleagues requests. Access to anonymized and aggregated population data about laboratory interventions of other colleagues and laboratory personnel can empower their environmental awareness of communicable infectious diseases and help them to determine which set of tests should be ordered when suspecting certain conditions.

In the North Norway health region tests are ordered by a GP who usually sends a request with a sample to perform several tests to confirm or discard a set of infectious agents to the microbiology service. At the microbiology service, the staff can add new tests for other infectious agents to the ordered request based on their knowledge; e.g., if the beginning of an epidemic is suspected, an outbreak of a communicable disease etc.

Conceptually, a request is a composition of some demographical data related to the patient and requester (GP/microbiology service) which contains a battery of individual tests, each aimed to detect an infectious agent. Test results are stored in the laboratory information system (LIS) in a proprietary format. Currently, reports of the tests performed in a geographical area in a period of time are generated using the SNOW system [36,37]. The system uses an agent-based architecture to extract the data related to tests and aggregate it per municipality and date to monitor the number of confirmed cases of several diseases. It applies transformation and aggregation rules to generate derived data from the individual results. For example, the infectious agent and its category (common cold viruses, gastrointestinal bacteria etc.) are derived from the type of test (DNA, culture, enzyme immunoassay etc). The aggregation of tests by request is done by defining grouping rules using the metadata of each result related to fields as patient id, request data, requester identifier etc.

In the case of transformation rules, the SNOW system implements a total of 25 rules to derive the values of some fields from other fields. These rules are implemented using the Drools business rule management system [38] as result of collaboration with the laboratory personnel. As no standard terminologies such as LOINC are implemented, such rules are based on the internal codes and names. As an example, the rule displayed in Fig. 1 is used to infer the infectious agent, the family of symptoms and the subcategory

GASTROINTESTINAL VIRUSES PSEUDOCODE

```
IF analysisType IS "FEC-ROTA" OR "FEC-ADNO" OR "FEC-NORO" OR "OPP-NORO" OR "FEC-SAPO" OR "OPP-SAPO"
THEN
   SET symptomGroup AS "Gastrointestinalt";
   SET subcategory AS "Virus";
   SET infectiousAgent AS analysisType.subStringAfter("-") + "virus";
ENDIF
```

Fig. 1. Rule to infer symptom group, virus subcategory and infectious agent from gastrointestinal viruses' analysis type.

GASTROINTESTINAL BACTERIA PSEUDOCODE

```
SET bacteriaSet AS {"SALMONELLA", "SHIGELLA", "YERSINIA", "CAMPYLOBACTER", "VIBRIO", "PLESIOMONAS", "AEROMONAS"}

IF analysisType IS "FEC-TPUS" OR "FEC-FAER"
THEN
   SET symptomGroup AS "Gastrointestinalt";
   SET subcategory AS "Bakterie";

   FOR EACH value IN bacteriaSet
      IF originalResult CONTAINS value
      THEN
         SET infectiousAgent AS value;
      ENDIF
   ENDFOR

   IF NOT originalResult CONTAINS "NEGATIVE"
   THEN
      SET testResult AS "POSITIVE";
   ENDIF
ENDIF
```

Fig. 2. Rule to infer symptom group, agent subcategory and infectious agent from bacteria analysis tests and the text in the original result.

of the agent from the analysis type. It also checks that test result among the valid result set. The infectious agent is set by taking the second part of the analysis type and adding the word "virus".

Fig. 2 shows the logic to infer the symptom group, the subcategory, and the infectious agent and result from the name of the test performed and the text fulfilled by the laboratory staff in the original result.

Although the SNOW system has been used successfully, the current strategic program of the Norwegian health authorities [39] for health IT has raised some architectural and interoperability challenges. Specifically, the requirements of interoperability among different systems of the Norwegian health network and the needs of data reuse for different purposes (CDS, clinical research etc.) has motivated work in the direction of standard-based solutions. As a result, we have implemented a new DW architecture based on openEHR archetypes and their query language, the archetype query language (AQL), to perform rapid and flexible aggregations of data for each of the interoperability and reuse scenarios.

1.4. Objective

This work aims to define the data warehouse environment to resolve interoperability and infrastructure challenges in clinical data reuse leveraging archetype-based EHR standards and technologies. A workflow is defined and implemented to facilitate the reuse of EHR data. The modeling, extraction, transformation, load and exploitation from existing legacy data in the EHR to its final use to implement indicators is covered. This paper describes how transformation and aggregation functions can be used to build openEHR compliant instances stored in an openEHR-based repository and how this repository can be queried at an archetype level to define indicators of microbiology test results. Policy and security challenges will remain out the scope of this work.

2. Materials and methods

2.1. The openEHR dual model architecture

The dual model methodology [12] proposes a separate definition of the information level, that represents the generic clinical data structures, and the conceptual level, that defines specific representations and meanings of those data structures. The information level or reference model (RM) defines the entities and properties that are not likely to change over time. This model must be generic enough to avoid modifications for supporting new characteristics or requirements from a clinical domain perspective. The RM entities are the basic building blocks for the conceptual or archetype level. The archetype model (AM) allows defining clinical information models by constraining specific data structures of the RM, to support specific clinical use cases. Such definitions are called archetypes. Archetypes define the maximum data schema of a clinical concept. To attach a formal specification of the meaning of archetypes, they can be linked to clinical terminologies. This makes archetypes a powerful mechanism to define information structures with attached meaning that support semantic interoperability among systems.

To achieve the objective of this work we will use openEHR archetypes as a mechanism for the modeling of the clinical data structures needed in our use case.

2.2. The archetype query language

The archetype query language (AQL) [40] is a declarative language to query clinical data which schema has been structured as archetypes. Queries are specified at an archetype level referencing the EHR information entities rather than the proprietary IT infrastructure. Therefore, queries are independent of the persistence schema or the particular technologies used in the implementation of the EHR where it is executed. This allows sharing queries among systems which have defined the clinical information models as archetypes. Table 1 shows the structure of an AQL query with its five components. The SELECT component specifies the path to the fields to be returned. The FROM section specifies the EHR or id of the patient medical records and the archetypes used by the query. The WHERE section specifies restrictions over the selection. ORDER BY allows ordering the result of the query. Finally, the TIME WINDOW section indicates the valid time period of the instances recorded. Similarly to SQL, AQL does support aggregation functions such as count, max, top and arithmetic and logical operators.

Fig. 3 represents an AQL query over the archetype openEHR-EHR-OBSERVATION lab_test_full_blood_count.v1 taken from the International Clinical Knowledge Manager [41] to retrieve all the tests of full blood count indicating a moderate leukocytosis. Following the openEHR RM class hierarchy, the query selects for the EHR identified as 1ADC27 any encounter composition that contains a full blood count test observation. The condition of the where clause constraints to values in white cell count between 11×10^9 and 17×10^9. The TIME WINDOW section is indicating that the fetched values should be restricted to the period of 1 year (P1Y) before 2014-02-12 (ISO-8601).

It must be noted that, in order to execute an AQL query, an archetype based database (DB) capable to process it is needed.

2.3. Normalization platform

It is a common situation that existing clinical data in EHR systems do not follow any standard, being stored in proprietary formats instead. To effectively reuse that information, it is necessary to transform it into a standard format. In our case, in openEHR format following the constraints defined in an archetype. To perform this transformation we make use of LinkEHR studio [42,43]. This tool is designed to facilitate the transformation of legacy data into archetype-compliant data. LinkEHR studio allows the definition of mappings between a legacy data schema (the schema of the original data) and an archetype of a specific standard (for example, an openEHR archetype). A mapping is defined by a set of pairs, each consisting of a function that specifies how to calculate the value of the target atomic attribute and a condition that must be satisfied to apply the function. Once these mappings have been defined by the user, the tool automatically generates a transformation script in XQuery format. The generated transformation script ensures that only data that satisfies the conditions defined in the mappings and the archetype constraints are finally transformed. Moreover, it applies a grouping logic by default based on the partition normal form for nested relations [44] that tries to minimize redundant information in the output result. When this program is executed over the existing data it generates an XML document that follows the standard information model and the constraints defined in the archetype.

2.4. OpenEHR persistence platform

As persistence and query mechanism an openEHR-based database (Think!EHR) provided by an industrial partner [45] was used. Such platform allows storing clinical information using as data schema the openEHR template object model. This means that data instances are stored as archetypes instances in a template schema rather than tables defined in a DB schema. The platform allows querying data via AQL at a clinical level; i.e., specifying restrictions over clinical concepts instead of the proprietary DB schema.

2.5. Modeling, extraction, transformation and load

The pipeline for processing data can be divided in the four main stages depicted in Fig. 4. Thick arrows represent the information flow across the stages of the pipeline, thin continuous lines represent interaction among components (definitions, generation of scripts or parameters that feeds some subsystem), dotted lines represent references to data schemas (XML schema or archetypes). The information modeling has been detached from the transformation stage and treated separately due to its complexity in comparison with traditional DW environments. Below follows the description of each stage.

(1) Modeling: The first step needed is to model the information requirements in the form of archetypes. These archetypes will become the schema of the data to be managed by the systems. In the modeling stage, existing archetypes can be reused or new ones can be created to fit the information requirements. Moreover, archetypes can be annotated with SNOMED-CT [34] or other terminologies to facilitate semantic interoperability and concept-based queries.
(2) Extraction: Existing results of laboratory tests are extracted from the LIS, cached and serialized in XML format (marshaling), containing plain representations of all the laboratory tests extracted without any nesting or association among them. The XML Schema of the marshalled cache is used in the transformation stage as the source schema for the definition of transformation and aggregation rules.
(3) Transformation: The modeled archetypes are mapped to the XML schema of the marshalled laboratory test results. The mapping specifies how to transform or group the original information fields that are available into a structure compliant with the constraints defined in the archetype. Once these mappings are defined using LinkEHR studio, the tool is able to automatically generate a transformation XQuery script. This XQuery script runs in a RESTful extract server which executes it on demand, creating an openEHR extract for a provided patient id. The extract aggregates the laboratory tests per request as specified in the mapping rules and delivers it through the service.
(4) Load: The extract server is called sequentially for each patient id to perform a load of the EHR into the openEHR-compliant database. When sources compliant with openEHR are available, the load stage can be used to integrate information from different health information systems (HIS).

2.6. Evaluation

Data from January 2013 to November 2014 of the Troms and Finnmark counties laboratory with an assigned population of 230,000 individuals (circa 270,000 tests) was standardized and transformed through the described pipeline and finally loaded into the data warehouse.

For the extraction stage the time to perform a full load was measured over 5 repetitions. For the transformation and load stages a

Table 1
AQL query structure.

Section		Data to be specified in the section
SELECT		Data elements to be returned and aggregation functions to use over it.
FROM	EHR	Id of the EHR to be queried.
	Containment criteria	Archetype sections that need to be contained in the specified EHR.
WHERE		Criteria that needs to be applied to the result values in order to be returned.
ORDER BY		Order criteria to apply to the result set.
TIME WINDOW		Date from which the specified data will be queried ignoring those older.

WHITE BLOOD CELLS COUNT AQL

```
SELECT o/data/events/data/items[at0078.13]/value AS WhiteCellCount

FROM EHR[ehr_id=1ADC27]
   CONTAINS COMPOSITION c [openEHR-EHR-COMPOSITION.encounter.v1]
   CONTAINS OBSERVATION o [openEHR-EHR-OBSERVATION.lab_test_full_blood_count.v1]

WHERE o/data/events/data/items[at0078.13]/value > 11000000000
        AND o/data/events/data/items[at0078.13]/value < 17000000000

TIME WINDOW P1Y/2014-02-12
```

Fig. 3. Example of white blood cells count AQL query.

random sample of 200 patient ids were used to measure the average time required to transform and load an extract containing the tests of each patient. To test the query response of the warehouse 30 repetitions of each query were performed.

To define specific query examples to be tested within the developed system, meetings where conducted with a medical advisor to define indicators of interest over the population that had been tested. Three types of indicators where defined. A first type was defined to monitor the consumption of tests; a second type to control the number of positive tested patients for a given municipality; and a third one to monitor the ratio between the positive tests for an infectious agent and the total ordered tests for that agent.

3. Results

3.1. System description

The architecture of the system to build an archetype based data warehouse environment has been presented. The workflow to conduct raw data to openEHR standardized instances is characterized by a segmentation into the standard ETL [46] stages, preceded by a modeling of clinical data stage. To increase modularity and scalability of the system it has been implemented as cooperative RESTful web services. The extraction service, placed on the original SNOW platform, runs a scheduled task, a.k.a. cron job, every night to extract the test records available. The records are then maintained as a cache and serialized on demand into a single XML file that is consumed by the transformation service. The transformation service is a standalone web service that can be queried given a patient id and returns the openEHR extract that is used to populate the data warehouse. The load of the warehouse is carried out by another scheduled task that calls the transformation service sequentially until all patient extracts have been loaded. The extraction and transformation services were implemented with the JavaEE 7 distribution, whereas the load service was implemented with the Spring Framework.

3.2. Information modeling

First, archetype reuse was attempted by reviewing the openEHR clinical knowledge manager [41]. Although, the openEHR-EHR-OBSERVATION.lab_test.v1 and its specialization openEHR-EHR-OBSERVATION.lab_test-microbiology.v1 had many of the necessary fields, the need to add some demographical information and fields useful for CDS, like infectious agent, the subcategory of the infectious agent (parasite, respiratory virus, gastrointestinal virus etc.), symptom group (respiratory, gastrointestinal), finally required the definition of new archetypes. The openEHR-EHR-OBSERVATION.lab_test-microbiology.v1 was found to be too specific for our purpose, as it describes detailed information of the analysis as macro and microscopic findings, culture findings etc. As a consequence, two archetypes had to be defined to represent the laboratory test results, namely openEHR-EHR-COMPOSITION.micro_lab_patient_request_composition.v1 and openEHR-EHR-OBSERVATION.micro_lab_test_request.v1. The first one contains the second one through a slot definition (a reference that can be defined inside an archetype to point to another external archetype). These archetypes, with the slot resolved at the level of the OBSERVATION node, are shown in Fig. 5.

The COMPOSITION archetype allows grouping all the results related to a single patient. The OBSERVATION archetype represents one request of laboratory tests. It is designed to group inside a CLUSTER all the individual or simple tests performed for each infectious agent. Each simple test contains the information related to an infectious agent analysis (test result, infectious agent, analysis type etc). Fields common to every simple test (material, patient id, patient gender etc.) are taken outside the CLUSTER as they are common for the request.

3.3. Data extraction

Original laboratory test results are extracted from the LIS databases, using a proprietary extraction library, and marshaled into an XML representation. As shown in Fig. 6, the XML representation contains a collection of plain elements without any aggregation per requester, date or patient id (which is previously anonymized).

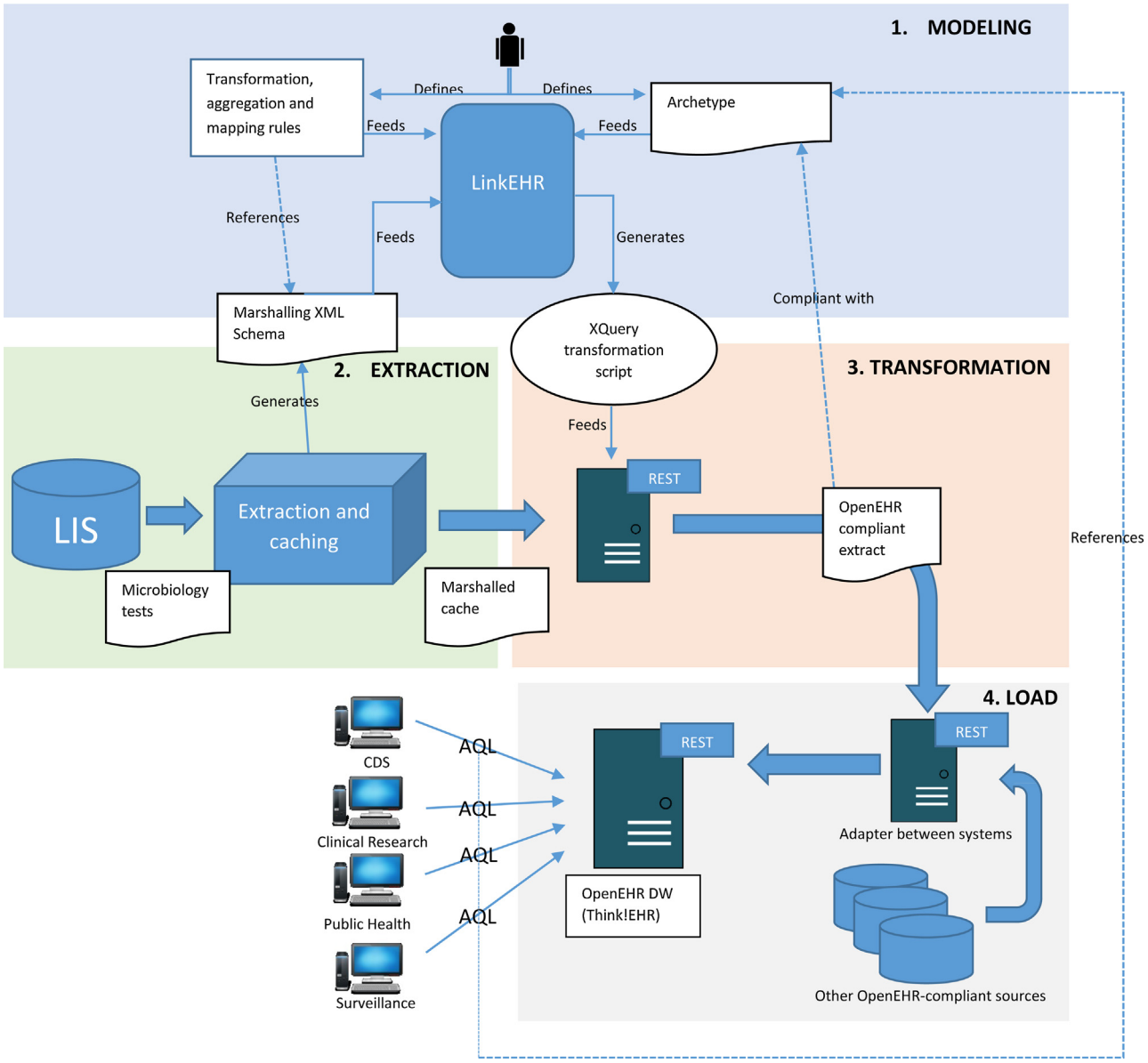

Fig. 4. Stages for the modeling, extraction, transformation and load of tests into openEHR compliant instances.

The description of the fields of each element in the XML representation is shown in Table 2. An XML schema for these XML files is also defined, to be used in the transformation process described in the next section.

3.4. Data transformation

A set of mappings between the XML schema of the marshaled LIS data and the selected archetypes were defined using LinkEHR. Many of the mappings were direct ones, binding a source field from the legacy schema to an archetype attribute without needing any additional transformation. This is the case of the patient identifier, analysis date, result date or gender of the patient.

In other cases, it was needed to define complex transformation rules. There is a mismatch between the existing data, shown in Table 2, and the archetype data structure shown in Fig. 5. The archetype defines a more extensive data set and thus, a direct mapping between the legacy data and the archetype is not enough to complete all possible data elements of the archetype. As explained before, the original SNOW system included a set of Drools rules and functions to derive new information based on the existing one. These rules were also implemented in the form of mappings to the archetype. Fig. 7 shows an example of this kind of complex mapping implementing in Drools and LinkEHR the pseudocode of Fig. 1. In this case, one single rule exists to derive the symptom group, the subcategory and the infectious agent from the analysis type identification. For example, the rule shown in Fig. 7 uses the "FECROTA" analysis type to generate the "Gastrointestinalt" symptom group (gastrointestinal in English), the "virus" subcategory and the "rotavirus" infectious agent. Note that, in the original system, this last value is calculated by using an external java implementation. This rule can be represented as a set of mappings, as shown in the same figure. In this case, the mappings are assigned to three different attributes of the archetype (symptom group, subcategory and infectious agent). The condition to be evaluated in the three cases is the same (to test if the analysis type is one of those of the predefined list) and the mapping function defines the values to be taken by the archetype attribute. In the case of the infectious agent, the data transformation can be defined in the mapping itself, without needing any external implementation. A similar work has to be

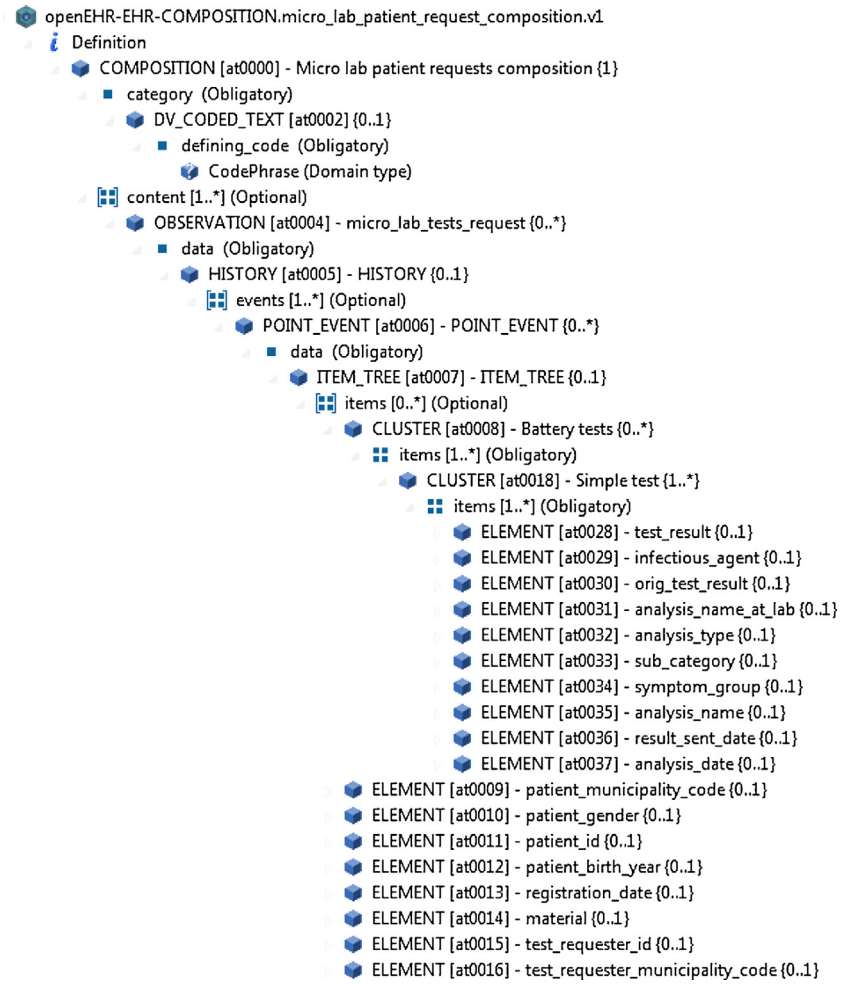

Fig. 5. Archetypes defined to structure the tests request.

```
<microlabresult>
  <id>2350459475284566896</id>
  <registrationDate>2013-02-24T12:56:00+01:00</registrationDate>
  <analysisDate>2013-02-25T15:35:20+01:00</analysisDate>
  <resultSentDate>2013-02-25T15:39:30+01:00</resultSentDate>
  <testRequesterId>68C1C359608B3C5HF3355544DBD9357A9D927EC6</testRequesterId>
  <analysisName>Nasopharynx-Chlamydophila pneumoniae DNA</analysisName>
  <analysisType>VNX-CPP</analysisType>
  <originalTestResult>NEGATIV</originalTestResult>
  <material>Nasopharynx</material>
  <requesterMunicipalityCode>1905</requesterMunicipalityCode>
  <gender>K</gender>
  <patientMunicipalityCode>1902</patientMunicipalityCode>
  <patientId>18E87T6711779EFE0X2V4T717D335A3A0F5422AD</patientId>
  <patientBornYear>1972</patientBornYear>
</microlabresult>
```

Fig. 6. Marshaled laboratory result.

done for every attribute of the archetypes which may have a data value extracted from the original laboratory data.

The transformation of data also includes an aggregation process. The original data from the LIS is a collection of results for simple tests, i.e., there exist a different data instance for each test performed to a single material, pertaining to the same laboratory request for the same patient.

Since the archetypes define a nested structure focused on a single patient, existing data has to be aggregated observing some rules. A different COMPOSITION instance was generated for each patient and laboratory request. Contained in it, an OBSERVATION instance is created aggregating all simple tests done to the same material into a CLUSTER called test battery. The test battery is thus composed by the complete set of simple sets that are part of the same laboratory request.

To generate the battery of simple tests inside the observation a grouping is performed by patient municipality, gender, birth year, registration date, material, requester id and requester. To perform such operation the default grouping semantics of LinkEHR matched our needs. This allows having all tests performed for one request

Drools rule

```
#
# Virus:
# Sets the infectious agent based on the letters after the '-' in analysisType. This is implemented in a Java class.
#
rule "Gastrointestinal symptoms related to virus"
        no-loop true salience 1
        when
                dataModel : SampleData (analysisType == "FEC-ROTA" || analysisType == "FEC-ADNO" || analysisType == "FEC-NORO"
                || analysisType == "OPP-NORO" || analysisType == "FEC-SAPO" || analysisType == "OPP-SAPO")
        then
                dataModel.setSymptomGroup(SymptomGroup.GASTROINTESTINAL.getName());
                dataModel.setSubCategory("Virus");
                dataModel.setInfectiousAgentFromAnalysisTypeAddVirus();
                update(dataModel);
end
```

LinkEHR mapping

Target archetype attribute	Condition	Mapping function
Simple test → Symptom group → Value	(/microlabresultscollection/microlabresult/analysisType = "FEC-ROTA") OR (/microlabresultscollection/microlabresult/analysisType = "FEC-ADNO") OR (/microlabresultscollection/microlabresult/analysisType = "FEC-NORO") OR (/microlabresultscollection/microlabresult/analysisType = "OPP-NORO") OR (/microlabresultscollection/microlabresult/analysisType = "FEC-SAPO") OR (/microlabresultscollection/microlabresult/analysisType = "OPP-SAPO")	"Gastrointestinalt"
Simple test → Subcategory → Value	(/microlabresultscollection/microlabresult/analysisType = "FEC-ROTA") OR (/microlabresultscollection/microlabresult/analysisType = "FEC-ADNO") OR (/microlabresultscollection/microlabresult/analysisType = "FEC-NORO") OR (/microlabresultscollection/microlabresult/analysisType = "OPP-NORO") OR (/microlabresultscollection/microlabresult/analysisType = "FEC-SAPO") OR (/microlabresultscollection/microlabresult/analysisType = "OPP-SAPO")	"Virus"
Simple test → Infectious agent → Value	(/microlabresultscollection/microlabresult/analysisType = "FEC-ROTA") OR (/microlabresultscollection/microlabresult/analysisType = "FEC-ADNO") OR (/microlabresultscollection/microlabresult/analysisType = "FEC-NORO") OR (/microlabresultscollection/microlabresult/analysisType = "OPP-NORO") OR (/microlabresultscollection/microlabresult/analysisType = "FEC-SAPO") OR (/microlabresultscollection/microlabresult/analysisType = "OPP-SAPO")	@concat(@substring-after(/ microlabresultscollection/ microlabresult/ analysisType,"-"),"virus")

Fig. 7. Mapping rules in Drools and LinkEHR formats.

Table 2
Description of laboratory test results fields.

Field	Description
Id	Internal unique identifier of the simple test.
Registration date	The date the test order was issued by the GP.
Analysis date	Data when the analysis was performed.
Result sent date	Date when the result was issued.
Test requester Id	Identifier of the entity requesting the test. GP identifier or microbiology laboratory identifier.
Analysis name	Complete name of the analysis. E.g., Nasopharynx–Rhinovirus RNA.
Analysis type	Code identifying the type of analysis. E.g., VNX–RHP.
Original test result	Test result issued by the laboratory without any transformation.
Material	The material of the sample to perform the test.
Requester municipality code	Identifier of the municipality where the test requester is settled.
Gender	Gender of the patient.
Patient municipality code	Identifier of the municipality where the patient dwells.
Patient Id	Identifier of the patient generated with a hashing function.
Patient born year	Year when the patient was born.

in one battery regardless of the entity ordering of those (GP or laboratory staff). As a result, an openEHR instance for each patient is created. It documents each of the requests performed, each containing a battery of all the plain tests carried out related to it.

3.5. Data load

The data load is performed sequentially for each patient. Given a patient id, his or her information is transformed and aggregated by executing the transformation defined in the previous step, obtaining an openEHR COMPOSITION data instance as a result. According to the openEHR specifications, this instance should be embedded into a proper openEHR EHR Extract message [47]. However, current implementations of the openEHR architecture have preferred the usage of simple serializations of the COMPOSITION instances instead of using the extract model. It was found that the generated COMPOSITION data instances slightly differed from the serialization accepted by the openEHR data warehouse in namespaces and message wrapping, complicating the seamless interoperability between systems. Therefore, preprocessing and reconciliation had to be performed before submitting each instance to the data warehouse.

As a result of all the previous processes, data from LIS was extracted, transformed and stored in openEHR format in an openEHR-compliant database, ready to be queried using AQL. Such queries can be defined dynamically performing rapid aggregations creating data instances at different levels of abstraction according with the needs of each use case. For example, the answer to a request of a patient extract from an external EHR system would only require querying the COMPOSITION whereas a query for health quality measures would require defining a higher level of abstraction requiring counting and selection over a particular time frame. As a proof of concept and evaluation of the developed platform, we developed several use-case scenarios, which are described in the next section.

4. Evaluation: *Pertussis* and *Salmonella* infection tests monitoring

The evaluation of the developed platform was made through the implementation of a case study focused in primary care testing interventions accessibility for *Pertussis* and *Salmonella*.

Pertussis is an infection caused by the gram-negative bacterium *Bordetella pertussis*. Norway, as other European countries, has suffered a significant increase in the incidence of the disease since 1997 [48]. Testing for pertussis is recommended for cases with an epidemiological link to a confirmed case, or in outbreak situations [49]. To alter the clinical course of the disease the antimicrobial treatment must be administered during the catharal stage (first 2 weeks).

Salmonella outbreak occurred in Norway mainly as a consequence of infections when living abroad [50]. It usually appears in located focus of infection which are required to be controlled.

We defined a set of indicators to allow any GP at any moment to access the tendencies in *Pertussis* and *Salmonella* tests and cultures. Thus, GPs can evaluate the eligibility of the test based in the tests results and behavior of other practitioners.

4.1. Indicators

For *Pertussis*, three indicators have been implemented. One to count the number of positive tests per day, a second one to monitor the number of negative tests per day, and a third one to count the total of tests performed for *Pertussis*. The implementation in AQL of the indicators to plot the values for January the 4th 2013 are presented in Table 3. The queries must be executed sequentially for each day that needs to be monitored to generate the values that form the evolution lines of positives, negatives and total number of tests.

For *Salmonella* two indicators have been implemented. The first one is an indicator to plot the number of positive cases for *Salmonella* in a selected municipality in a given window of time. For example, first row in Table 4 selects the number of positive tested patients for the municipality coded with 1917 in the period from January 1st to January 15th 2013. This indicator allows the GP to evaluate the eligibility of a test for a patient depending on the confirmed cases in his municipality. A second indicator for Salmonella was implemented to monitor the evolution of positive vs. total tests in the population. Second and third row of Table 4 depict the queries to implement the former indicator for August the 20th 2013. This second query is executed for each day to plot the evolution of positives vs. tested.

4.2. Performance

The performance was measured over an Intel Xenon 2.9 GHz with 12 GB RAM memory. The extraction stage had to be evaluated as an atomic operation for the full load of records from January 2013 to November 2014. Including caching and marshaling of all tests records, the operation took an average of 36.544 s (95% confidence interval [CI], 35.830–37.758).

The evaluation for the transformation and load stages was carried out by randomly selecting a sample of 200 records and performing the transformation and load. The average transformation time to create an openEHR instances for one patient id was 13.483 s (95% CI, 12.981–13.985). The average load time for one openEHR instance into the archetype-based repository, including format reconciliation, was 1.567 s (95% CI, 0.87–2.264).

The average response time for monitoring *Pertussis* for the count queries defined in the previous section was 1.287 s (95% CI, 1.227–1.347). The average time to execute the query to monitor *Salmonella* per municipality took 2.419 s (95% CI, 2.326–2.512). The count query to monitor of the ratio positives vs. tested for *Salmonella* took 0.656 s (95% CI, 0.609–0.703).

5. Discussion

The components and tasks needed to build a DW environment have been presented and deployed for the primary care case study described. The proposed environment takes advantage of the available archetype-based technologies to perform each of the necessary operations; i.e., modeling, extraction, transformation, load and query.

5.1. Archetype-based data warehouse systems

Several studies have approached the reuse of clinical information, some from the standardization and aggregation perspective while others from the DW perspective. The pipeline presented tries to take the advantages of both to allow generating standard aggregated data sets for different purposes; e.g., indicators, clinical research, quality measures, etc. These data can be queried using an archetype query language such as AQL, be semantically interoperable with other openEHR-based systems or be integrated with analytical or rule based systems that make use of archetypes, such as the guideline definition language (GDL) proposed by Chen [51]. An advantage of the standard-based approach with respect to traditional DW is the modeling capabilities provided by EHR standards. For example, what is modeled as a tree in an archetype would need to be modelled as a complex snowflaked schema or OLAP cube in a traditional DW. Such schema would need to re-model some RM classes and relationships, disallowing interoperability and creating a different information model from the one already used and validated by domain experts as archetypes. We also find that queries in non EHR standard-based warehouses are performed at a database schema level rather than at a clinical domain level. This compels the user defining the query to have a detailed knowledge of the underlying database. This limitation can be softened by using an archetype-based data repository, since the same archetypes that the clinicians model to represent the EHR information are used as the interface to define queries. Moreover, since these queries are based on an archetype definition and not on a particular database implementation, they can be shared and executed in other systems seamlessly.

5.2. Extraction, transformation and load process

A limitation related to the proposed infrastructure is the transactional control at the ETL stages when comparing it to commercial DW. A problematic load not rolled back would lead to inconsistent inferences. The operations performed are long lasting executions among stateless systems and therefore production systems need to deal with global transactions. Whereas the presented proposal

Table 3
Queries executed to monitor Pertussis to plot the values corresponding to the 4th of January 2013.

Pertussis monitoring	
Count positive tests of Pertussis for the day specified in the parameter (e.g., 2013-01-04)	SELECT count (o1/data[at0001]/events[at0002]/data[at0003]/items[at0022]) – – count (patient Id) FROM EHR e CONTAINS COMPOSITION c CONTAINS (OBSERVATION o1[openEHR-EHR-OBSERVATION.micro_lab_test.v1] and OBSERVATION o2[openEHR-EHR-OBSERVATION.micro_lab_test.v1]) WHERE (o1/data[at0001]/events[at0002]/data[at0003]/items[at0010]/items[at0043]/items[at0036]/value='Kikhoste'and o1/data[at0001]/events[at0002]/data[at0003]/items[at0010]/items[at0043]/items[at0037]/value='Positiv') and o1/data[at0001]/events[at0002/data[at0003]/items[at0024]/value>='2013-01-04' and o1/data[at0001]/events[at0002/data[at0003]/items[at0024]/value<'2013-01-05'
Count negative tests of *Pertussis* for the day specified in the parameter (e.g., 2013-01-04)	SELECT count (o1/data[at0001]/events[at0002]/data[at0003]/items[at0022]) FROM EHR e CONTAINS COMPOSITION c CONTAINS (OBSERVATION o1[openEHR-EHR-OBSERVATION.micro_lab_test.v1] and OBSERVATION o2[openEHR-EHR-OBSERVATION.micro_lab_test.v1]) WHERE (o1/data[at0001]/events[at0002]/data[at0003]/items[at0010]/items[at0043]/items[at0036]/value='Kikhoste'and o1/data[at0001]/events[at0002]/data[at0003]/items[at0010]/items[at0043]/items[at0037]/value='Negativ') and o1/data[at0001]/events[at0002]/data[at0003]/items[at0024]/value>='2013-01-04' and o1/data[at0001]/events[at0002]/data[at0003]/items[at0024]/value<'2013-01-05'
Total tests of *Pertussis* (in Norwegian 'Kikhoste') performed for the day specified in the parameter (e.g., 2013-01-04)	SELECT count (o1/data[at0001]/events[at0002]/data[at0003]/items[at0022]) FROM EHR e CONTAINS COMPOSITION c CONTAINS (OBSERVATION o1[openEHR-EHR-OBSERVATION.micro_lab_test.v1] and OBSERVATION o2[openEHR-EHR-OBSERVATION.micro_lab_test.v1]) WHERE (o1/data[at0001]/events[at0002]/data[at0003]/items[at0010]/items[at0043]/items[at0036]/value='Kikhoste') and o1/data[at0001]/events[at0002]/data[at0003]/items[at0024]/value>='2013-01-04' and o1/data[at0001]/events[at0002]/data[at0003]/items[at0024]/value<'2013-01-05'

is more powerful in interoperability and integration, the use of state of the art technologies makes it less robust in aspects as transactional control of the ETL stages with respect to classic DW environments.

In the transformation step, it has been shown how existing rules were implemented as mappings defining the transformation of existing data into openEHR-compliant data. Only one rule could not be implemented. It defined that if it already exists a result for the same test and the same patient in the last 90 days, the new result is not processed. Since the transformation process is made at an individual instance level each time, it is not possible to check the historic information to apply this rule.

In comparison to approaches like [27,30], where fixed mappings are used to raise the level of abstraction, our approach allows to perform specific purpose aggregations and views over the openEHR platform at run-time with the query language. This way it is intended to maximize flexibility and minimize maintenance while, at the same time, promote control over the state of the data set at any time to avoid problems of real time solutions; e.g., experiments with non-reproducible results [23].

Table 4
Queries to implement the indicators for *Salmonella*.

Salmonella monitoring	
Salmonella cases in the specified municipality (same as patient just confirmed) in the first 2 weeks of January	SELECT count (o1/data[at0001]/events[at0002]/data[at0003]/items[at0022]/value) – – count (patient Id) FROM EHR e CONTAINS COMPOSITION c CONTAINS (OBSERVATION o1[openEHR-EHR-OBSERVATION.micro_lab_test.v1] and OBSERVATION o2[openEHR-EHR-OBSERVATION.micro_lab_test.v1]) WHERE (o1/data[at0001]/events[at0002]/data[at0003]/items[at0010]/items[at0043]/items[at0036]/value='*Salmonella*' and o1/data[at0001]/events[at0002]/data[at0003]/items[at0010]/items[at0043]/items[at0037]/value='Positiv') and o1/data[at0001]/events[at0002]/data[at0003]/items[at0020]/value='1917' and o1/data[at0001]/events[at0002]/data[at0003]/items[at0024]/value>='2013-01-01' and o1/data[at0001]/events[at0002]/data[at0003]/items[at0024]/value<'2013-01-15'
Positives in the whole region to plot evolution per day (# abbreviates the path to the CLUSTER)	SELECT count (o1/data[at0001]/events[at0002]/data[at0003]/items[at0022]) FROM EHR e CONTAINS COMPOSITION c CONTAINS (OBSERVATION o1#micro_lab_test and OBSERVATION o2#micro_lab_test) WHERE o1#battery/Simple_test/infectious_agent='*Salmonella*' and o1#battery/Simple_test/test_result='Positiv' and o1#registration_date>='2013-01-01' and o1#registration_date<'2013-01-15'
Negatives in the whole region to plot evolution per day (# abbreviates the path to the CLUSTER)	SELECT count (o1/data[at0001]/events[at0002]/data[at0003]/items[at0022]) FROM EHR e CONTAINS COMPOSITION c CONTAINS (OBSERVATION o1#micro_lab_test and OBSERVATION o2#micro_lab_test) WHERE o1#battery/Simple_test/infectious_agent='*Salmonella*' and o1#battery/Simple_test/test_result='Negativ' and o1#registration_date>='2013-01-01' and o1#registration_date<'2013-01-15'

With regards to the lack of implementation of the openEHR, EHR extract potential issues in synchronization and version control can arise when integrating several sources and deciding which entities need to be updated. The adoption of such information model could alleviate these situations by keeping track of the modifications of entities wrapping them in "versioned objects". The mismatch in the XML serialization of COMPOSITIONs is another issue as can cause an overload in the load stage, since format reconciliation can require a big effort [52] and nullify the main advantages of implementing EHR standards; i.e., interoperability.

5.3. Query process

In our approach, once data is transformed and stored in openEHR format, AQL is used to aggregate data for each scenario. A first limitation comes from the nature of AQL, as it is intended to query the EHR rather than generate aggregated views for inference models. When a higher level of abstraction is required; for example, to feed CDS or indicators that need to be based on queries, we found that some needed features are not currently supported by the AQL specification; e.g., subqueries, group by etc. A possible solution is to move the queries to an ontological level and triplets-based query languages like SPARQL [53]. While an ontological approach could jeopardize structural interoperability among systems, some studies [54] have proposed techniques to convert archetypes into OWL ontologies enabling SPARQL queries execution over their ontological representation. However these transformations rely on complex archetype to ontology mappings to preserve structural, entities relationships and cardinality consistency. Research in the direction of semi-automatic inference of knowledge models from archetypes annotations and structure is needed if ontological reasoning over archetypes instances is desired. A second limitation linked to AQL is that the current specification is intended for openEHR environments hampering its use in other archetype-based standards such as EN ISO 13,606. For example, the reference to the EHR entity in the FROM clause cannot be implemented in EN ISO 13,606 environments as such entity is only part of the openEHR EHR information model. Thirdly, since AQL is a query language designed to query the EHR, it lacks data manipulation functionalities (insert, update, delete etc.) as EHR data should not be manipulated without a strict versioning control. For data reuse in research or CDS, manipulation functions are needed to implement certain transformations that cannot be performed at a transformation stage. An example is the rule presented in the previous section where consideration of historical data was needed.

We are aware that the annotation of the standard extracts with clinical terminologies can allow to overcome some of the limitations discussed. For example, the use of the terminology semantics may enhance our query capabilities and enable inferences over annotated concepts. However, this are challenges to explore as future work.

5.4. Performance

The performance times show acceptable times for the caching and marshalling stage with a mean of 36.544 s per month, specially taking into account the amount of records loaded. The transformation times were also acceptable with 13.483 s on average to produce an instance. This operation is a batch process and can be performed without disrupting the production environment and therefore not hampering the practitioners' acceptance. Similarly the average load time for each register (1.567 s) is acceptable to perform as a scheduled batch process. In regards to the query times for the use case indicators we find times between 0.656 and 2.419 s. While queries like those related to the monitoring of *Pertussis* have acceptable execution times, considering that speed is one of the crucial factors for the success of CDS [55], we believe that long lasting queries like the monitoring of *Salmonella* need improvement. A feasible cause and limitation of our system is that we were operating the openEHR platform under the minimum specifications of memory recommended in production environments. Improvement of performance times remains as future work.

5.5. Comparison with other data reuse environments

Recently, DW infrastructures for reuse of information have been proposed [56–58]. However, to the best of the authors' knowledge, these systems were to be used inside one organization and not based in clinical information architecture standards. For example, Hu et al. [56] relied on ontologies and clinical terminologies as SNOMED-CT to model clinical concepts. This is useful when the structure of the concept is known inside one organization but might be inaccurate for data reuse across several organizations or levels of the health care system, since the structure of information is unknown. The SHARPn consortium [33] approaches that challenge defining a powerful data reuse pipeline which normalizes to clinical element models. This solution counts on natural language processing tools and data transfer verification mechanisms that our proposal lacks. However, a drawback of such platform with respect to ours is that, once data is normalized, it is stored in a relational or documental DB. This forces queries to be performed over nonstandard persistence platforms. Besides, when SHARPn queries are specified in a standard for quality measures (HQMF) an ad-hoc transformation needs to be done to the particular query format of the repository that stores the extracts. This can lead to the misinterpretation of the underlying semantics when aggregations are performed, since the persistence schema differs from the information standard schema.

Another highly related work is the informatics for integrating biology and the bedside system (i2b2) [59], a clinical research analytics platform gaining momentum in the US. Its architecture provides a set of components that cover from project management and natural language processing to ontology management and correlation analysis. However, it relies in a relational data model not directly compliant with EHR interoperability standards, with the limitations we have discussed in previous paragraphs. We envision a combination of i2b2 infrastructure with ours, where i2b2 is used as intra-organization solution taking advantage of its components for de-identification, data cleaning, NPL etc and its data model is mapped to our infrastructure, thus allowing the combination with other archetype-based data sources and where AQL can be used to perform queries.

6. Conclusion

Achieving an efficient reuse of clinical data is a must to guarantee the future of clinical research and CDS. This work contributes by proposing a DW environment based on EHR standards to allow the interoperability, agile aggregation of data sets, and reuse of data in different scenarios; e.g., clinical research, CDS, surveillance etc. This paper has described the technologies and steps necessary to allow the modeling, transformation, integration, standardization and aggregation of the data flowing from the EHR to reuse it. The integration and standardization are carried out in the ETL stages with mapping over archetypes. The aggregation for particular reuse scenarios is performed at a warehouse level to allow a maximum adaptability to the different reuse scenarios without need to disrupt the ETL infrastructure.

Archetype based technologies and standards are mature enough to be combined into a pipeline that allows applying transformation and aggregation functions to proprietary data to standardize it and,

later, query it at an EHR level regardless the underlying technologies. Inference models in clinical research and CDS can benefit from this by defining queries to fetch the data sets needed. Moreover, the standard nature of the information model allows an easy integration with new systems to allow the DW dataset grow in an ordered manner.

When working with health information we find that the modeling usually needs to be performed with domain experts thinking both in the architecture and semantics of information. Therefore, for clinical ETL environments it may be recommendable to treat information modeling as a separate stage, thus redefining the ETL process into a METL (modeling, extraction, transformation and load). That would mean recognizing the importance of clinical information modeling processes, not only for routine health care delivery but also for clinical data reuse.

Summary points

- EHR data reuse is necessary to support clinical research and decision support.
- To allow reuse it is necessary to define new methods to standardize EHR data and allow to query and aggregate it for different reuse scenarios.
- Archetype based methodologies for modelling, transformation and storage of EHR data can be effectively combined to create a data warehouse environment to allow clinical data reuse.
- The use of EHR information standards and query languages enable different stakeholders a seamless reuse of information by accessing it at an EHR level regardless the underlying technical infrastructure.

Conflict of interest

We wish to confirm that there are no known conflicts of interest associated with this publication and there has been no significant financial support for this work that could have influenced its outcome.

We confirm that the manuscript has been read and approved by all named authors and that there are no other persons who satisfied the criteria for authorship but are not listed. We further confirm that the order of authors listed in the manuscript has been approved by all of us.

We confirm that we have given due consideration to the protection of intellectual property associated with this work and that there are no impediments to publication, including the timing of publication, with respect to intellectual property. In so doing we confirm that we have followed the regulations of our institutions concerning intellectual property.

We understand that the Corresponding author is the sole contact for the Editorial process (including Editorial Manager and direct communications with the office). He is responsible for communicating with the other authors about progress, submissions of revisions and final approval of proofs. We confirm that we have provided a current, correct email address which is accessible by the Corresponding author and which has been configured to accept email from (Luis.Marco.Ruiz@telemed.no).

Signed by all authors as follows:
Luis Marco-Ruiz.
David Moner.
José A. Maldonado.
Nils Kolstrup.
Johan G. Bellika.

Author contribution

All authors contributed in the conception of the study. L.M. and D.M. designed, implemented the infrastructure and drafted the manuscript. J.A.M. and J.G.B. have contributed to the technical developments and validation of the work. N.K. contributed advising the work from a clinical perspective and defined the use cases. All authors revised critically and contributed to the manuscript before approving the final version.

Acknowledgements

This work was supported by Helse Nord [grant HST1121-13 and 9057/HST1120-13]; the NILS Science and Sustainability Programme [grant number 005-ABEL-IM-2013] from Iceland, Liechtenstein and Norway through the EEA Financial Mechanism, operated by Universidad Complutense de Madrid; and by the Spanish Ministry of Economy and Competitiveness [grant PTQ-12-05620]. We would like to thank to Marand d.o.o. and Torje S. Henriksen for the products provided and their assistance and support during this work. We would like to acknowledge Gunnar Skov Simonsen and Marit Wiklund at the microbiology laboratory service of the University Hospital of North Norway for their support for this work.

References

[1] J.V. Selby, H.M. Krumholz, R.E. Kuntz, F.S. Collins, Network news powering clinical research, Sci. Transl. Med. 5 (April 24 (182)) (2013) 182fs13.
[2] P.B. Jensen, L.J. Jensen, S. Brunak, Mining electronic health records: towards better research applications and clinical care, Nat. Rev. Genet. 13 (June (6)) (2012) 395–405.
[3] K. Kawamoto, C.A. Houlihan, E.A. Balas, D.F. Lobach, Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success, BMJ 330 (April 2 (7494)) (2005) 765.
[4] C.P. Friedman, A.K. Wong, D. Blumenthal, Achieving a nationwide learning health system, Sci. Transl. Med. 2 (Oct 11 (57)) (2010) 57cm29.
[5] S.F. Terry, P.F. Terry, Power to the people: participant ownership of clinical trial data, Sci. Transl. Med. 3 (September 2 (69)) (2011) 69cm3.
[6] A.H. Krist, J.W. Beasley, J.C. Crosson, D.C. Kibbe, M.S. Klinkman, C.U. Lehmann, et al., Electronic health record functionality needed to better support primary care, J. Am. Med. Inf. Assoc. 21 (January 9 (5)) (2014) 764–771.
[7] EHR4CR. Electronic Health Records for Clinical Research [Internet]. [cited 14.09.14]. Available from: <http://www.ehr4cr.eu/>.
[8] PCORnet [Internet]. PCORnet. [cited 14.09.14]. Available from: <http://www.pcornet.org>.
[9] HCS Research Collaboratory [Internet]. [cited 14.09.14]. Available from: <http://commonfund.nih.gov/hcscollaboratory/index>.
[10] J.J. Nadler, G.J. Downing, Liberating health data for clinical research applications, Sci. Transl. Med. 2 (October 2 (18)) (2010) 18cm6.
[11] R.H. Dolin, L. Alschuler, S. Boyer, C. Beebe, F.M. Behlen, P.V. Biron, et al., HL7Clinical document architecture, release 2, J. Am. Med. Inf. Assoc. 13 (January 1 (1)) (2006) 30–39.
[12] T. Beale, Archetypes Constraint-Bbased Domain Models for Futureproof Information Systems, OOPSLA, 2002, Workshop Behav Semant, 2002.
[13] ISO 13606:2008 – Health informatics – Electronic health record communication, (2008).
[14] V.N. Stroetmann, D. Kalra, P. Lewalle, A. Rector, J.M. Rodrigues, K.A. Stroetmann, et al., Semantic Interoperability for Better Health and Safer Healthcare [Internet], January, European Commission, Directorate-General Information Society and Media, 2009 http://dx.doi.org/10.2759/38514
[15] W. Goossen, A. Goossen-Baremans, M. van der Zel, Detailed clinical models: a review, Health Inf. Res. 16 (4) (2010) 201.
[16] Clinical Information Model Initiative [Internet]. [cited 14.11.14]. Available from: <http://www.opencimi.org/>.
[17] epSOS: Home [Internet]. [cited 13.11.14]. Available from: <http://www.epsos.eu/>.
[18] The Interoperability Toolkit (ITK) – Health and Social Care Information Centre [Internet]. [cited 28.03.14]. Available from: <http://systems.hscic.gov.uk/interop/background/itk>.
[19] Medicare C for Baltimore MS 7500 SB, USA M. Overview [Internet]. 2014 [cited 08.09.14]. Available from: <https://www.cms.gov/Regulations-and-Guidance/Legislation/EHRIncentivePrograms/index.html?redirect=/EHRIncentivePrograms/>.
[20] D. Blumenthal, M. Tavenner, The 'meaningful use' regulation for electronic health records, N. Engl. J. Med. 363 (July 13 (6)) (2010) 501–504.

[21] G. Schadow, D.C. Russler, C.N. Mead, C.J. McDonald, Integrating medical information and knowledge in the HL7 RIM, Proc. AMIA Annu. Symp. AMIA Symp. (2000) 764–768.
[22] A.L. Rector, P.D. Johnson, S. Tu, C. Wroe, J. Rogers, S. Quaglini, et al., Interface of inference models with concept and medical record models, Proc. Artif. Intell. Med. Eur. AIME-2001 1 (January) (2001) 314–323.
[23] C. Safran, Reuse of clinical data, IMIA 9 (1) (2014) 52–54.
[24] Arden Syntax [Internet]. [cited 13.11.14]. Available from: <http://www.hl7org/Special/Committees/arden/index.cfm>.
[25] HL7 Standards Product Brief – GELLO (HL7 Version 3 Standard: Gello: A Common Expression Language, Release 2) [Internet]. [cited 13.11.14]. Available from: <http://www.hl7org/implement/standards/product_brief.cfm?product_id=5>.
[26] R.A. Greenes, Clinical Decision Support: The Road to Broad Adoption, Academic Press, 2014, pp. 929.
[27] M. Peleg, S. Keren, Y. Denekamp, Mapping computerized clinical guidelines to electronic medical records: knowledge-data ontological mapper (KDOM), J. Biomed. Inform. 41 (February (1)) (2008) 180–201.
[28] K. Kawamoto, G. Del Fiol, H.R. Strasberg, N. Hulse, C. Curtis, J.J. Cimino, et al., Multi-national, multi-institutional analysis of clinical decision support data needs to inform development of the HL7 virtual medical record standard, AMIA Annu. Symp. Proc. AMIA Symp. AMIA Symp. 2010 (2010) 377–381.
[29] M. Marcos, J.A. Maldonado, B. Martínez-Salvador, D. Boscá, M. Robles, Interoperability of clinical decision-support systems and electronic health records using archetypes: a case study in clinical trial eligibility, J. Biomed. Inf. 46 (August (4)) (2013) 676–689.
[30] J.T. Fernández-Breis, J.A. Maldonado, M. Marcos, M.D.C. Legaz-García, D. Moner, J. Torres-Sospedra, et al., Leveraging electronic healthcare record standards and semantic web technologies for the identification of patient cohorts, J. Am. Med. Inf. Assoc. JAMA (August 9) (2013).
[31] MobiGuide [Internet]. [cited 14.11.14]. Available from: <http://www.mobiguide-project.eu/>.
[32] A. González-Ferrer, M. Peleg, B. Verhees, J.-M. Verlinden, C. Marcos, Data Integration for Clinical Decision Support Based on OpenEHR Archetypes and HL7 Virtual Medical Record. Process Support and Knowledge Representation in Health Care [Internet], Springer, Berlin Heidelberg, 2013, pp. 71–84 [cited 21.03.14] http://link.springer.com/chapter/10.1007/978-3-642-36438-9_5
[33] J. Pathak, K.R. Bailey, C.E. Beebe, S. Bethard, D.C. Carrell, P.J. Chen, et al., Normalization and standardization of electronic health records for high-throughput phenotyping: the SHARPn consortium, J. Am. Med. Inf. 20 (December (e2)) (2013) e341–e348.
[34] J. Holstiege, T. Mathes, D. Pieper, Effects of computer-aided clinical decision support systems in improving antibiotic prescribing by primary care providers: a systematic review, J. Am. Med. Inf. Assoc. JAMIA (August 14) (2014).
[35] P. Chen, M.J. Tanasijevic, R.A. Schoenenberger, J. Fiskio, G.J. Kuperman, D.W. Bates, A computer-based intervention for improving the appropriateness of antiepileptic drug level monitoring, Am. J. Clin. Pathol. 119 (March (3)) (2003) 432–438.
[36] J.G. Bellika, T. Hasvold, G. Hartvigsen, Propagation of program control: a tool for distributed disease surveillance, Int. J. Med. Inf. 76 (April (4)) (2007) 313–329.
[37] Data Mining in Clinical Medicine [Internet]. [cited 27.10.14]. Available from: <http://www.springer.com/life+sciences/systems+biology+and+bioinformatics/book/978-1-4939-1984-0>.
[38] Drools – Drools – Business Rules Management System (Java™, Open Source) [Internet]. [cited 20.11.14]. Available from: <http://www.drools.org/>.
[39] Tiltak 15.5 Folkeregisteret i helsenettet – Nasjonal IKT [Internet]. [cited 10.11.14]. Available from: <http://nasjonalikt.no/no/satsingsomrader/2_struktur_-_systemarkitektur_informasjonsgrunnlag_og_sikkerhet/tiltak_155_folkeregisteret_i_helsenettet/Tiltak+15.5+Folkeregisteret+i+helsenettet.9UFRjK5S.ips>.
[40] Archetype Query Language [Internet]. Available from: <http://www.openehr.org/wiki/display/spec/Archetype+Query+Language+Description>.
[41] Ocean Informatics. Clinical Knowledge Manager [Internet]. OpenEHR Clinical Knowledge Manager. [cited 14.10.13]. Available from: <http://www.openehr.org/ckm/>.
[42] J.A. Maldonado, D. Moner, D. Boscá, J.T. Fernández-Breis, C. Angulo, M. Robles, LinkEHR-Ed. A multi-reference model archetype editor based on formal semantics, Int. J. Med. Inf. 78 (August (8)) (2009) 559–570.
[43] J.A. Maldonado, C.M. Costa, D. Moner, M. Menárguez-Tortosa, D. Boscá, J.A. Miñarro Giménez, et al., Using the researchEHR platform to facilitate the practical application of the EHR standards, J. Biomed. Inf. 45 (August (4)) (2012) 746–762.
[44] W.Y. Mok, Y.-K. Ng, D.W. Embley, A normal form for precisely characterizing redundancy in nested relations, ACM Trans. Database Syst. 21 (March (1)) (1996) 77–106.
[45] Marand d.o.o. Think!MED Clinical TM.
[46] R. Kimball, M. Ross, The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling, 3rd ed., Wiley, Indianapolis, IN, 2013, pp. 600.
[47] H. Frankel, T. Beale, OpenEHR EHR. Extract Information Model, The OpenEHR Foundation, 2007.
[48] Kikhoste (pertussis) – veileder for helsepersonell – Folkehelseinstituttet [Internet]. [cited 18.09.14]. Available from: <http://www.fhi.no/eway/default.aspx?pid=239&trg=Content_6493&Main_6157=6287:0:25,5499&MainContent_6287=6493:0:25,6833&Content_6493=6441:82766::0:6446:62:::0:0>.
[49] Ministry of Health, New Zeland. Pertussis – Communicable Disease Control Manual.
[50] Salmonellose – veileder for helsepersonell – Folkehelseinstituttet [Internet]. [cited 08.09.14]. Available from: <http://www.fhi.no/eway/default.aspx?pid=239&trg=Content_6493&Main_6157=6287:0:25,5499&MainContent_6287=6493:0:25,6833&Content_6493=6441:82847::0:6446:106:::0:0>.
[51] Rong Chen, Guide Definition Language (GDL) [Internet]. (2013). Available from: <http://www.openehr.org/downloads/ds_and_guidelines>.
[52] J.G. Klann, M.D. Buck, J. Brown, M. Hadley, R. Elmore, G.M. Weber, et al., Query health: standards-based, cross-platform population health surveillance, J. Am. Med. Inf. Assoc. (April 3) (2014), amiajnl –2014–002707.
[53] SPARQL Query Language for RDF [Internet]. [cited 21.09.14]. Available from: <http://www.w3org/TR/rdf-sparql-query/>.
[54] L. Lezcano, M.-A. Sicilia, C. Rodríguez-Solano, Integrating reasoning and clinical archetypes using OWL ontologies and SWRL rules, J. Biomed. Inform. 44 (April (2)) (2011) 343–353.
[55] D.W. Bates, G.J. Kuperman, S. Wang, T. Gandhi, A. Kittler, L. Volk, et al., Ten commandments for effective clinical decision support: making the practice of evidence-based medicine a reality, J. Am. Med. Inf. Assoc. JAMIA 10 (December (6)) (2003) 523–530.
[56] H. Hu, M. Correll, L. Kvecher, M. Osmond, J. Clark, A. Bekhash, et al., DW4TR: a data warehouse for translational research, J. Biomed. Inf. 44 (December (6)) (2011) 1004–1019.
[57] S. Yoo, S. Kim, K.-H. Lee, C.W. Jeong, S.W. Youn, K.U. Park, et al., Electronically implemented clinical indicators based on a data warehouse in a tertiary hospital: its clinical benefit and effectiveness, Int. J. Med. Inf. 83 (July (7)) (2014) 507–516.
[58] I. Danciu, J.D. Cowan, M. Basford, X. Wang, A. Saip, S. Osgood, et al., Secondary use of clinical data: the vanderbilt approach, J. Biomed. Inf. (2014) [cited 21.09.14] http://www.sciencedirect.com/science/article/pii/S1532046414000392
[59] S.N. Murphy, G. Weber, M. Mendis, V. Gainer, H.C. Chueh, S. Churchill, et al., Serving the enterprise and beyond with informatics for integrating biology and the bedside (i2b2), J. Am. Med. Inf. Assoc. JAMIA 17 (April (2)) (2010) 124–130.