Universidade de Aveiro
2015

Departamento de Eletrónica, Telecomunicações e Informática
MAP-I - Doctoral program in Computer Science

Luís António Bastião Silva

Arquiteturas federadas para integração de dados biomédicos

Federated architecture for biomedical data integration

Universidade de Aveiro
2015

Departamento de Eletrónica, Telecomunicações e Informática
MAP-I – Doctoral program in Computer Science

Luís António Bastião Silva

Arquiteturas federadas para integração de dados biomédicos

Federated architecture for biomedical data integration

tese apresentada às Universidades de Aveiro, Minho e Porto para cumprimento dos requisitos necessários à obtenção do grau de Doutor em ciências da computação (MAP-I), realizada sob a orientação científica do Doutor Carlos Manuel de Azevedo Costa, Professor Auxiliar do Departamento de Electrónica e Telecomunicações e Informática da Universidade de Aveiro e do Doutor José Luís Guimarães Oliveira, Professor Associado do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro.

Trabalho financiado sob identificação SFRH / BD / 79389 / 2011 por:

o júri / the jury

presidente / president

Doutor **João de Lemos Pinto**
Professor Catedrático, Universidade de Aveiro

Vogais / examiners committee

Doutor **Luís Manuel Dias Coelho Soares Barbosa**
Professor Associado, Universidade do Minho

Doutor **Álvaro Manuel Reis da Rocha**
Professor Auxiliar com Agregação, Faculdade de Ciências e Tecnologia, Universidade de Coimbra

Doutor **Rui Pedro Charters Lopes Rijo**
Professor Adjunto, Escola Superior de Tecnologia e Gestão de Leiria, Instituto Politécnico de Leiria

Doutor **José Antonio Seoane Fernández**
Pós-doctoral Research Fellow, Curtis Lab. Stanford Cancer Institute, Palo Alto, Estados Unidos da América.

Doutor **Carlos Manuel Azevedo Costa**
Professor Auxiliar, Universidade de Aveiro (orientador)

acknowledgements

I would like to express my sincere appreciation to the bioinformatics group at “Instituto de Engenharia Electrónica e Informática de Aveiro” (IEETA) for the great working environment provided, the productive discussions and overall cooperation. In that regard, a special thanks to prof. Carlos Costa and prof. José Luis Oliveira for giving me the opportunity to carry out my research project, for their guidance and unconditional support. I am also grateful to the members of the research group of Erasmus Medical Center (EMC) in Rotterdam, in which I have been involved during the international internship, for their support and advices, in particular to Peter Rijnbeek and Marius Gheorghe. I also thank Milton Santos for his important comments and availability to discuss ideas.
Last but not least, I want to extend my deepest gratitude to my family, friends and girlfriend Jeniffer for the friendship, support and patience.
Finally, I gratefully acknowledge the “Fundação para a Ciência e Tecnologia” (FCT) for making possible this Ph.D. work, through the grant SFRH/ BD/ 79389/ 2011.

agradecimentos

Gostaria de expressar o meu especial agradecimento ao grupo de Bioinformática do Instituto de Engenharia Electrónica e Informática de Aveiro (IEETA) pelo grande espaço de cooperação e ambiente de trabalho disponibilizado pelos colegas. Em particular, um muito obrigado aos meus orientadores prof. Carlos Costa e prof. José Luis Oliveira pela oportunidade, partilha das suas visões, recomendações e todo o processo de orientação que foram fundamentais para a execução deste trabalho. Gostaria de agradecer também aos membros do grupo de investigação do Erasmus Medical Center (EMC) em Roterdão, em que estive durante um estágio internacional pelo apoio e conselhos, em particular ao Peter Rijnbeek e Marius Gheorghe. Agradeço também a colaboração do Milton Santos pelo apoio prestado e disponibilidade para discussão de ideias.
Por último, mas não menos importante, gostaria de agradecer à minha família, amigos e à minha namorada Jeniffer pela apoio, amizade e paciência.
Finalmente, um grande agredecimento à Fundação para a Ciência e Tecnologia (FCT) por fazer possível este doutoramento, através da bolsa SFRH/ BD/ 79389/ 2011.

resumo

A adoção sucessiva das tecnologias de comunicação e de informação na área da saúde tem permitido um aumento na diversidade e na qualidade dos serviços prestados, mas, ao mesmo tempo, tem gerado uma enorme quantidade de dados, cujo valor científico está ainda por explorar. A partilha e o acesso integrado a esta informação poderá permitir a identificação de novas descobertas que possam conduzir a melhores diagnósticos e a melhores tratamentos clínicos.
Esta tese propõe novos modelos de integração e de exploração de dados com vista à extração de conhecimento biomédico a partir de múltiplas fontes de dados.
A primeira contribuição é uma arquitetura baseada em nuvem para partilha de serviços de imagem médica. Esta solução oferece um mecanismo de registo simplificado para fornecedores e serviços, permitindo o acesso remoto e facilitando a integração de diferentes fontes de dados. A segunda proposta é uma arquitetura baseada em sensores para integração de registos electrónicos de pacientes. Esta estratégia segue um modelo de integração federado e tem como objetivo fornecer uma solução escalável que permita a pesquisa em múltiplos sistemas de informação. Finalmente, o terceiro contributo é um sistema aberto para disponibilizar dados de pacientes num contexto europeu. Todas as soluções foram implementadas e validadas em cenários reais.

palavras-chave

Bases de dados biomédicas, integração de dados, gestão de dados, PACS, imagem médica, DICOM

abstract

The last decades have been characterized by a continuous adoption of IT solutions in the healthcare sector, which resulted in the proliferation of tremendous amounts of data over heterogeneous systems. Distinct data types are currently generated, manipulated, and stored, in the several institutions where patients are treated. The data sharing and an integrated access to this information will allow extracting relevant knowledge that can lead to better diagnostics and treatments.
This thesis proposes new integration models for gathering information and extracting knowledge from multiple and heterogeneous biomedical sources.
The scenario complexity led us to split the integration problem according to the data type and to the usage specificity. The first contribution is a cloud-based architecture for exchanging medical imaging services. It offers a simplified registration mechanism for providers and services, promotes remote data access, and facilitates the integration of distributed data sources. Moreover, it is compliant with international standards, ensuring the platform interoperability with current medical imaging devices. The second proposal is a sensor-based architecture for integration of electronic health records. It follows a federated integration model and aims to provide a scalable solution to search and retrieve data from multiple information systems. The last contribution is an open architecture for gathering patient-level data from disperse and heterogeneous databases. All the proposed solutions were deployed and validated in real world use cases.

keywords

Biomedical databases, data integration, data management, PACS, medical imaging, DICOM

Table of Contents

List of Figures

List of Tables

Acronyms

AETitle	Application Entity Title
API	Application Programming Interface
BBMRI	Biobanking and Biomolecular Resources Research
CDA	Clinical Document Architecture
CDM	Common Data Modal
DBMS	Database Management Systems
DICOM	Digital Image and Communication in Medicine
DIMSE	DICOM Message Service Element
EHR	Electronic Health Records
EHR4CR	Electronic Health Records for Clinical Research
EMIF	European Medical Information Framework
ETL	Extract, Transform, Load
HIS	Hospital Information System
HL7	Health Level seven
IaaS	Infrastructure-as-a-Service
IE	Information Extraction
IHE	Integration Healthcare Enterprise
IMI	Innovative Medicines Initiative
IOD	Information Object Definition
IR	Information Retrieval
ICT	Information and Communications Technology
ISO	International Organization for Standardization
IT	Information Technology
NEMA	The National Electrical Manufacturers Association
NIH	National Institute of Health (U.S.)
MR	Magnetic Resonance
MRI	Magnetic Resonance Imaging
OHDSI	Observational Health Data Sciences and Informatics
OMOP	Observational Medical Outcomes Partnership
PaaS	Platform-as-a-Service
PACS	Picture Archiving and Communication System
PDE	Primary Data Extraction
PET	Positron Emission Tomography
PHR	Personal Heath Record
PPRE	Private Research Environment
QRPH	Quality Research Public Health
RDBMS	Relational Database Management System
REDCap	Research Electronic Data Capture
RIM	Reference Information Model

RIS	Radiology Information System
SaaS	Software-as-a-Service
SCP	Service Class Provider
SCU	Service Class User
SDCP	Service Delivery Cloud Platform
SDK	Software Development Toolkit
SOA	Software Oriented Architecture
SOP	Service Object Pair
SR	Structured Report
TLV	Tag-Length-Value
UID	Unique Identifier
UMLS	Unified Medical Language System
US	Ultrasound
VR	Value Representation
WADO	Web Access to DICOM persistent Objects
XA	X-ray Angiography
XDS	Cross-Enterprise Document Sharing
XDS-I	Cross-Enterprise Document Sharing for Imaging
XML	Extensible Markup Language

1 Introduction

"Begin with the end in Mind."

The 7 Habits of Highly Effective, Stephen Covey

1.1 Motivation

During the last decades, healthcare providers, biomedical researchers, pharmaceutical companies, and many health-related services have been producing a huge and ever increasing amount of data, used mostly for a primary care service. Moreover, to maintain citizens' clinical history and due to legal requirements, data need to be stored for many years. As patients access a multitude of health services from distinct providers, their clinical information is, typically, kept fragmented in several information systems. This imposes new challenges to health institutions, which need to create efficient mechanisms to manage all these data, e.g. related to archiving, searching, sharing and privacy issues. As data have been gathered, the motivation to explore their value for secondary use has also increased. Healthcare professionals, researchers and providers started to realize their

potential value to help understanding better the efficacy of drugs, diseases trajectories and comorbidities. Moreover, these databases can contain clinical records related to rare cases that when combined in a larger scale, might help researchers to identify the causes and eventually develop new treatments. Medical researchers have been trying to extract new knowledge from patients' records and from cohort studies, e.g. analyzing risk factors for a specific health condition through population screening [1-3]. Moreover, besides the healthcare interest, their statistical analysis can also improve institutions' efficacy and productivity [4] [5].

In the last years, many researchers have been investigating and promoting translational medicine, a recent field that aims better exploring (i.e., translating) the *omics'* findings in clinical practice. Translational research in medicine is the combination of genotype-to-phenotype results made available to create novel diseases' treatments and, generically, for the benefit of mankind [6]. Biomedical informatics is a key part of this process, by joining areas and methodologies that are crucial for transitional medicine such as bioinformatics, medical imaging, clinical informatics, and public health informatics [7-9].

A huge amount of data and scientific results are being generated by genomics-related research, and there is a great expectation about how these outcomes can be integrated and used in clinical practice. A tremendous impact on clinical decisions, related to diagnosis, prognosis**,** treatment and epidemiology, can be obtained through the combination of information such as patient records, medical imaging, clinical reports, genes' mutations, and environmental data [10]. The integration of all these different sources is a great challenge that promises to improve patient care, create new guidelines for clinical decisions, potentiate the development of new treatments, and give insights towards the improvement of health services in general [11-13].

However, the access to an integrated view of this information is still difficult. Data are being gathered in distinct countries and institutions, creating many isolated silos, and there is no integrated view of the whole "catalogue" that can take advantage of those results in a holistic manner [14]. A huge part of information systems still stores the data in proprietary formats at the various repositories detained by hospitals and laboratories. Moreover, distinct privacy issues associated with patient data sharing still exist due to legal and ethical requirements. Besides the political and ethical issues that need to be addressed, but are out of the scope of this work, new biomedical data integration strategies are needed to integrate and correlate all these data, and to provide information in a unified way.

1.2 Objectives

The objective of this thesis is to investigate a new software solution to integrate and gather knowledge from heterogeneous and different biomedical data sources, stored in

multiple medical institutions and research centres (Figure 1.1). Generically, our research motivation was based on how to integrate and gather knowledge from *M* different types of databases among *N* different healthcare units.

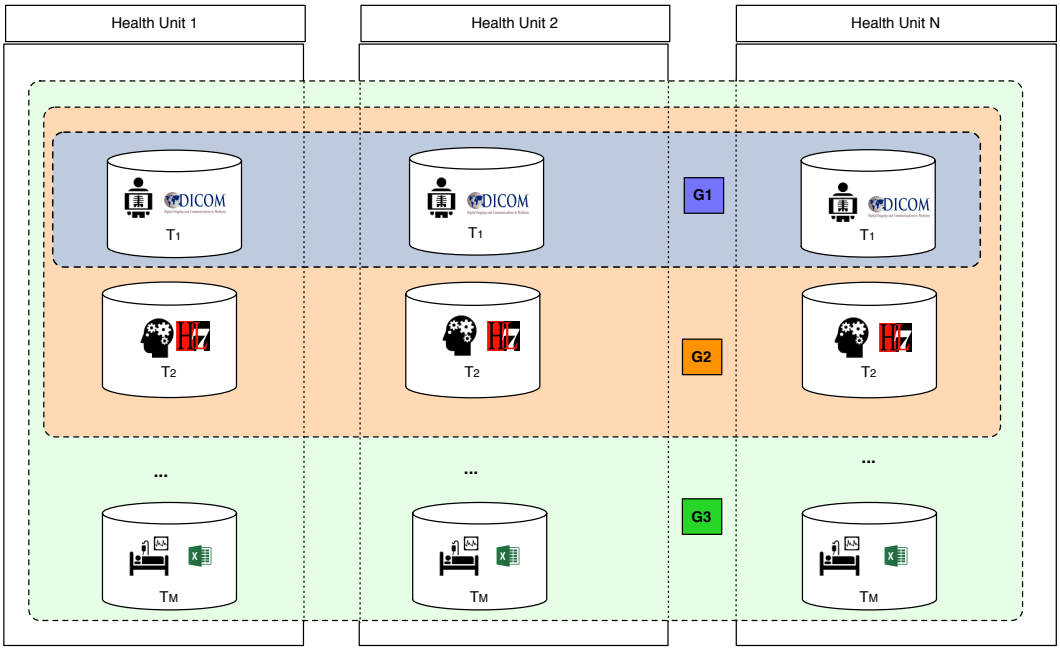

Figure 1.1: Problem statement definition.

This research can be addressed at multiple dimensions. Thus, we defined a plan that starts from a very specific problem up to a more broad and wide scenario. Therefore, three hypotheses have been proposed:

- A cloud computing architecture can solve the integration between different institutions and so, for *N* healthcare units we propose to do an integration of a type of database T_1 (Figure 1.1– G1).
- For *N* healthcare units, it is possible to create a federated strategy to inquire databases from a single and centralized platform in a scalable manner (Figure 1.1 – G2).
- A high level and summary view that integrates T_M databases will be possible for *N* healthcare units (Figure 1.1– G3).

1.3 Key contributions

This doctorate has contributed to biomedical integration and knowledge extraction in three different aspects:

1) Scientific contributions

- A common API to deliver services over multi-vendor cloud resources [15]: a framework to grant interoperability between different cloud providers, allowing a high level of integration, mainly focused in three types of services: storage, database and notification services.
- A DICOM relay over the cloud [16]: based on the cloud framework, we developed a solution to federate multiple medical imaging repositories.
- A centralized platform for geo-distributed PACS management [17]: A single monitoring and management system over multiple medical imaging centres.
- A sensor-based architecture for medical imaging workflow analysis [18]: a polling-based strategy to gather and integrate multiple sources. Workflow network, free text reports and medical images were considered, but the architecture is fully extensible to add other data sources.
- Normalization of heterogeneous medical imaging data [19]: an architecture to normalize dose measures that are supplied by different medical devices.
- An architecture to summarize patient-level data [20]: The main idea was to create a software solution that facilitates the aggregation of biomedical data at several layers of detail, from fingerprinting up to patient-level data. This is currently a major component of the EMIF Platform [21].

2) Clinical application of the developed solutions

- The deployment of scientific results in the real world scenarios often creates new problems. A key part of this work was the establishment of partnerships with clinical experts in areas such as medical imaging, structured and unstructured reports in cardiology, Alzheimer cohorts, and Electronic Health Records (EHR). The developed solutions were validated by hundreds of users using data from thousands of patients.
- Four invited articles have been published in professional-oriented magazines [22-25], contributing for the dissemination among the healthcare community.

3) Open source software

- A first contribution was in Dicoogle PACS [26], a repository for medical imaging. The main ones were many blueprints in the core architecture, which was relevant not only for this doctorate, but also for other projects. The development of a new software architecture allows Dicoogle to be extensible and following that, many contributions have been raised such as Dose Extractor [19], Semantic PACS [27] and Sensor-based architecture for federate medical repositories [18].

- The Catalogue [20] was developed, which is a web and online software to summarize patient level data across boundaries and countries.

1.4 Thesis outline

The thesis is organized in five more chapters described in Figure 1.2. The main scientific output is also shown for each contributed section. Chapter 2 aims to provide a state-of-the-art description of subjects that are most relevant to this work. It presents major biomedical data sources, and several standards proposed for storage and data transfer. In addition, it explores main computational methods that can be used to access, extract, store, retrieve, and search biomedical data. Moreover, it will introduce several techniques that can support the integration of biomedical data. Some examples and recent work on the area are presented and discussed.

Chapter 3 presents a framework to integrate medical images repositories between different healthcare units using cloud computing services. This solution creates an easy way to establish inter-institutional medical imaging services, namely shared processes and integration of repositories. The performance results have been measured and a real case study is discussed.

Chapter 4 presents an architecture to integrate multiple data sources in the medical imaging departments, using sensors for medical imaging and for semi-structured medical reports. Moreover, the presented solutions allow performing combined query across different sensors of information.

In Chapter 5 an architecture able to aggregate summarized data from disperse databases is presented. This allows keeping the medical records in each healthcare institution safeguarded and, at the same time, permits querying several characteristics of the databases with numerous layers of information details.

Chapter 6 presents final remarks of this thesis, and highlights directions for future work.

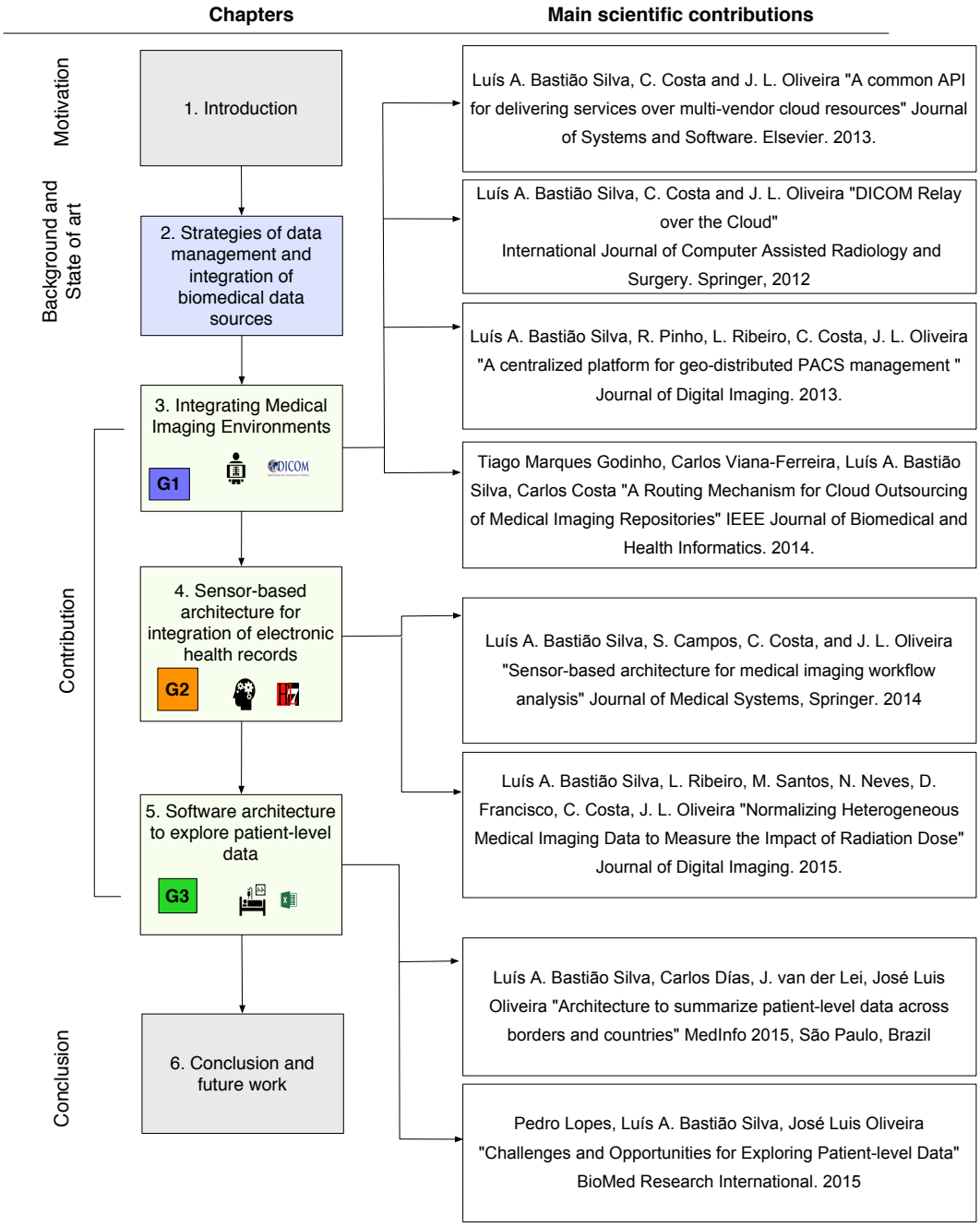

Figure 1.2: Thesis structure, highlighting the main scientific contributions.

2 Strategies for data management and integration of biomedical data sources

"Science never solves a problem without creating ten more."

George Bernard Shaw

Clinical practice is increasingly relying on synergies between different biomedical fields. This implies to cope with multiple data sources associated to medical institutions and biomedical research centres such as administrative, clinical data, biological data and population data. This information is frequently not readily available, and it is hard to find specific information across the boundaries of each administrative domain. Several methodologies can be established to manage biomedical data. However, the numerous dimensions of these data imply efficient strategies for data access, acquisition, storage, integration and search, as well information extraction over heterogeneous repositories (Figure 2.1).

This chapter describes distinct biomedical data sources, discussing the growing problem of health data fragmentation and the strategies to deal with such reality. Moreover, several software architectures will be presented, from the most commonly used and known, to the most complex. Finally, related work and models adopted by integration platforms were reviewed.

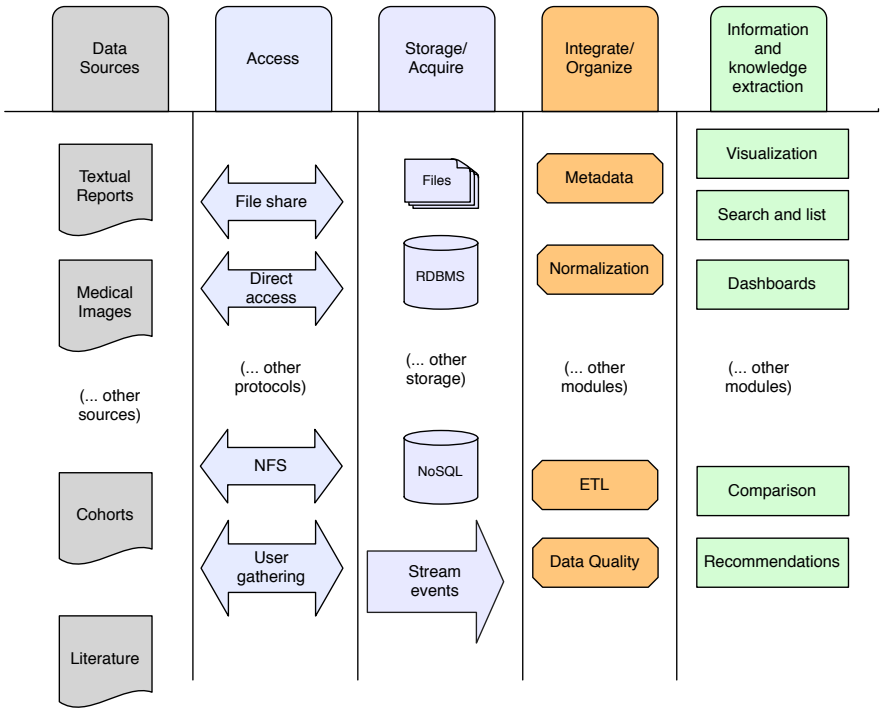

Figure 2.1: Pipeline of data management in biomedical informatics

2.1 Biomedical information

Citizens are assisted in many health care centres along their lives. They are also submitted to several medical tests such as blood analysis, medical imaging, and electrocardiogram. All this information contributes to the medical history of each individual, a valuable insight for future diagnostics or prognostics. To promote the storage of this history in digital format, several Electronic Health Record (EHR) systems have been developed, allowing gathering distinct information such as patient demographic, progress notes, symptoms, medications, vital signs, past medical history, laboratory data and radiology reports [28] (Figure 2.2). Despite this overall ambition, the reality often shows that most of these systems do not offer all these functionality.

Personal Health Record (PHR) [29] is a particular type of EHR, which describes a self-contained individual registry generated, maintained and kept by patients, primarily for his/her use and stored in a secure and trustable environment [30]. PHR data can be obtained by several EHR, typically from different healthcare providers. However, they can include also information supplied by the patient itself, which can be objective or subjective [30]. For instance, daily self-monitoring results (e.g. glucose), periodic weight monitoring and measurements generated by home monitoring devices are objective information that the healthcare professionals can rely on. Subjective information is related to self-assessment, and it may include details such as illness symptoms or fitness

information. There are also several types of PHR, i.e. they can be portable data devices, web or application entities.

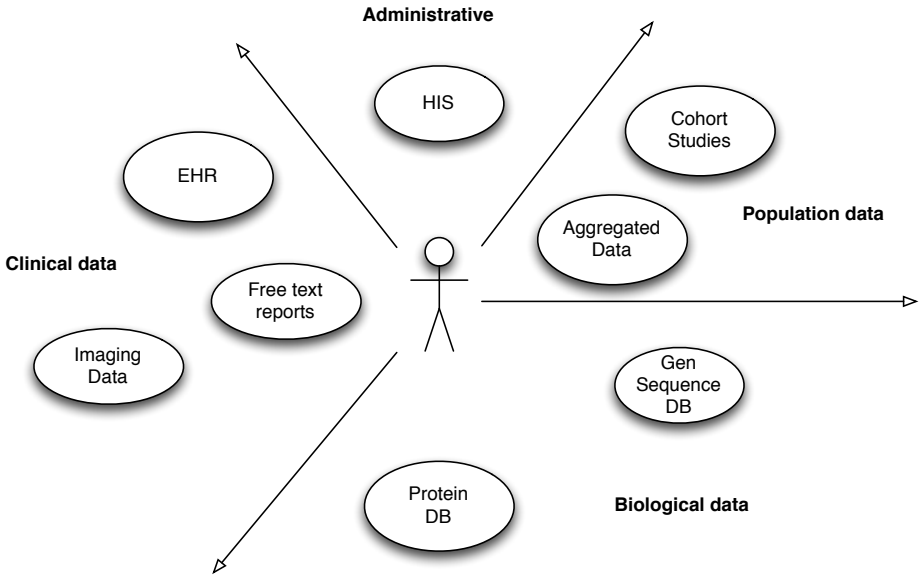

Figure 2.2: Biomedical data source: several information dimensions

Summarizing, EHR is a formal way for federating all biomedical data into a single integrated view. There are many ways to achieve this goal and some will be discussed in the next sections, having as a reference the current state of the art. Nevertheless, there are not many practical cases of fully integrated EHR and the patient information continues to be dispersed among distinct systems with very little integration and where the sharing processes are based on ad-hoc actions.

2.1.1 Hospital Information Systems

Hospital Information Systems (HIS) support three important aspects in healthcare units: all the clinical workflow from the admission up to discharge; hospital daily administrative transactions (financial, pay roll, etc.); and evaluation of hospital performances/costs. HIS provides automation for several tasks related to the interaction between patients and the healthcare institution, including registration, admission, discharge, transfer and accounting [31-33]. However, several other services are still lacking. For instance, access to the patient's clinical results such as laboratory, pathology, microbiology and many others could also be provided.

Many healthcare centres such as imaging centres, pharmacy, rehabilitation and clinical laboratories have their own system requirements. The integration of HIS with other information systems is very important to implement integrated workflows and provide automatic actions, avoiding manual updates and having multiple systems with redundant information. Most of HIS are developed through the integration of multiple information systems that need to collaborate and communicate among them. This implies the need for

standards and interfaces. The communication with other information systems can be performed through the HL7 (Health Level 7), a standard that intends to support all healthcare workflows. Nevertheless, in several cases, these information systems do not follow standards, and the data access is also an ad-hoc process for each institution or used software.

2.1.2 Radiology Information System

A Radiology Information System (RIS) was designed to support administrative and clinical operations of a radiology department. The idea is to create a system to store, manipulate, and distribute patient radiological data and images. The system manages patient registration and scheduling, patient list management (process patient and film folder records), provide interfacing with medical imaging acquisition devices, document scanning, reporting and printout, patient tracking, interactive user interface to record documents and management of modalities and material (Figure 2.3).

The RIS and medical imaging repositories work together to provide the necessary information for supporting the physician review and reporting of examinations. On the other hand, RIS complements the HIS for providing efficient workflow in radiology practices. Typically, the communication with RIS can also be performed through the HL7. Despite this standard, the reality shows that communication with these systems is difficult and the extraction of information needs ad-hoc actions for each healthcare unit.

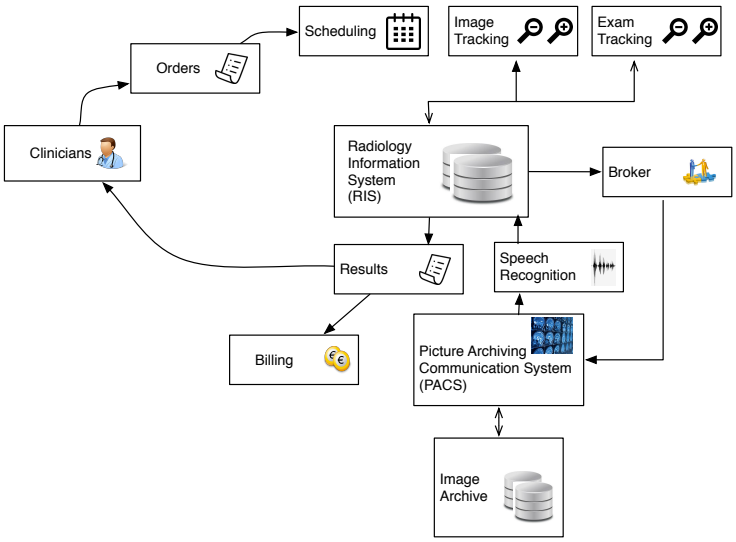

Figure 2.3: Radiology Information System - interaction (image adapted from [34])

2.1.3 Medical imaging systems

Picture Archive and Communication System (PACS)

Picture Archive and Communication System (PACS) defines a set of systems and encompasses hardware, communication networks and software technologies for the

acquisition, distribution, storage and analysis of digital images in distributed environments [35]. The main components are image acquisition and scanning devices, storage archive units, display workstations and databases with patient records. All those components communicate through the network, contributing to the integrated system.

Nowadays, the importance of medical imaging systems in healthcare institutions is unquestionable and even the small medical imaging centres are intensively using these tools. Thus, it is necessary to endow these medical institutions with digital storage and visualization devices to improve the physician's workflow. The tremendous evolution of Information Technology (IT) is promoting a revolution in the radiology sector like, for instance, new image acquisition techniques or Cloud-based repositories. Moreover, the workflows were speeded up and costs were reduced, with a significant financial impact on healthcare institutions. During the last two decades, health centres have made significant investment in IT to support medical imaging laboratories. Those technological advances are contributing for the improvement of diagnosis and supporting clinical decisions. One of the important changes was in the management of digital radiology modalities, e.g. X-Rays, Computed Tomography (CT), Magnetic Resonance (MR), Ultrasounds (US), X-Ray Angiography (XA) and so on. They proliferate in the last decades and even the small healthcare institutions are able now to purchase those acquisition devices.

The volume of generated data in digital medical imaging laboratories is tremendous, especially in modalities with high resolutions or dynamic image acquisition (i.e. cine-loops). Efficient storage and distribution mechanisms for ensuring permanent availability of produced studies are huge challenges [36, 37], at least without requiring major upgrades and overhauls that significantly increase the total cost of ownership over time [37]. As result, institutions must deal with several problems related with the planning and maintenance of IT infrastructure, including solution scalability, fault tolerance, performance issues, hardware maintenance costs, system obsolescence and migration. A redundancy and disaster/catastrophe plan are also important issues for patient data security.

Filmless radiology department refers to a unit where the majority of analogue films have been replaced by electronic systems, which acquire, store, distribute and visualize medical digital images. The operational symbiosis of PACS and RIS systems is fundamental to achieve this ambition.

DICOM Standard

The use of technologies and information systems in the medical imaging area started with the manufacture of equipment able to acquire, store and transfer data between medical stations. However, communication between all these devices did not follow a standard and many manufacturers developed their own communication protocols increasing the

difficult in accessing from different vendor devices. In the 80s, NEMA (National Electrical Manufactures' Association) and ACR (American College of Radiology) created a consortium for normalizing formats and communication processes in the medical imaging. This group developed a set of standardizations and guidelines, which allows communication and transfer of medical imaging data between different vendors' devices. This movement had a strong impact on the development and expansion of PACS.

In 1992, this consortium released a standard named DICOM (Digital Image and Communication in Medicine) version 3.0. The latest version of DICOM, at the time of writing this document, consists of 20 parts and 192 working supplements that affect specific issues of one or various parts of the standard. DICOM is an international standard that defines the file format and directory structure needed for offline communication. In addition, the communication protocols and packet semantics are described to contribute to operation between medical imaging devices.

DICOM standard not only grants basic connectivity between imaging devices, but also supplies guidelines for workflow in an imaging department. Nowadays, it is a major contributor to the exchange of structured medical imaging data and almost all medical imaging manufacturers are following the DICOM rules.

2.1.4 Cohort studies

A cohort study is a form of a longitudinal study used in medicine to analyse the risk factors that determine a subject's susceptibility to contract a specific disease. It is a distribution study of determinants health related state or events in a specified population. The cohort studies main goal is to understand how to control a health problem. The cohort studies can measure, for instance, disease frequency, distribution, determinants of a disease and many other metrics. Cohort study is undertaken to support the existence of an association between suspected cause and disease.

A cohort study follows two or more groups of people from exposure to a certain disease outcome. There are several kinds of cohort studies:

- General population: sample of population defined geographically or administratively.
- Restricted population group: a group of selected people who offer facilitated conditions to evaluate the exposure, for instance, healthcare professionals.
- Special exposure: a group of selected people who will be subjected to an unusual high level of exposure.

Cohort studies represent one of the most used methods to identify incidence and natural history of a disease. They can be used to examine multiple outcomes after a single

exposure. Nevertheless, they are less useful for examination of rare events or those that take a long time to develop.

The results of the cohort studies are generally published in scientific forums, but the data resources are typically stored in tabular databases by healthcare institutions. These data might be useful when integrated with other medical information sources. Figure 2.4 shows a cohort study to predict the risk factors associated to coronary heart disease, considering variables like, for instance, patient sex, age and blood pressure.

Table 1. Baseline characteristics of the subjects completing the 12-month follow-up.

Variables	*Baseline smoking status*				
	E-cigarettes only	Tobacco cigarettes only	Dual smoking	Overall	
	(n = 236)	*(n = 491)*	*(n = 232)*	*(N = 959)*	*p**
Mean age in years (SD)	45.2 (10.7)	44.2 (11.9)	44.3 (12.0)	44.5 (11.6)	
Male gender, %	62.7	48.7	64.2	55.9	*, **
Mean BMI (SD)	24.7 (3.9)	24.4 (4.0)	24.8 (4.0)	24.6 (4.0)	
Married, %	60.5	54.8	55.7	56.4	
Employed, %	78.9	79.5	74.4	78.2	
Educational level, %					
- Elementary / Middle	21.9	21.9	22.1	22.4	
- High school	54.5	42.5	46.7	46.3	*
- Bachelor or higher	23.6	35.6	31.2	31.3	*
Physical activity (69 missing)					
- At work, %	18.9	20.3	15.7	18.9	
- Weekly hours at work, mean (SD)	23.8 (18.4)	26.7 (16.6)	22.9 (19.0)	25.3 (17.5)	
- At home, %	48.0	48.0	51.3	48.8	
- Weekly hours at home, mean (SD)	5.3 (4.7)	5.2 (5.5)	5.3 (4.5)	5.3 (5.1)	
Alcohol use					
Regular alcohol intake, %	20.0	29.4	27.4	26.6	*
Mean alcohol units daily (SD)	2.1 (1.2)	2.1 (1.6)	2.1 (1.0)	2.1 (1.4)	
Cardiovascular risk and health					
- Hypertension, %	13.6	11.6	9.9	11.7	
- Diabetes, %	4.2	3.3	4.3	3.8	
- Hypercholesterolemia, %	8.1	8.8	10.3	9.0	
- Self-reported health, mean (SD) [€]	8.0 (1.3)	7.8 (1.3)	7.7 (1.2)	7.8 (1.3)	
- Low (<6) self-reported health [€], %	5.0	5.5	3.3	4.9	
Smoking pattern, mean (SD)					
- Years of tobacco smoking	21.4 (10.7)[¥]	22.3 (12.6)	25.2 (12.5)	22.9 (12.1)	**, ***
- N. tobacco cigarettes daily	—	14.1 (8.1)	14.9 (9.8)	14.4 (8.7)	
- Months of electronic smoking	8.8 (5.1)	—	8.4 (4.5)	8.6 (4.8)	
- N. e-cigarette daily puffs	162 (276)	—	96 (146)	130 (224)	***
- EC nicotine dose in mg	8.7 (5.2)	—	10.9 (5.6)	9.8 (5.5)	***
E-cigarettes by nicotine dose, %					
- No nicotine	12.8	—	5.6	9.3	***
- 3 to 8 mg	23.5	—	19.1	21.3	
- 9 mg	40.7	—	34.0	37.4	

Figure 2.4: Sample of a cohort study

2.1.5 Omics data

Omics research encompasses a set of areas such as genomics, proteomics, and metabolomics, related respectively to the study of the genome, the proteome and the metabolome. On the other side, genetic medicine is concerned with the connection between the genotype and phenotype. A genotype is defined as an individual's genetic makeup, i.e. his/her DNA sequence, while a phenotype can be defined as the visible traits of an organism, which are produced by the interaction of a genotype and the environment. For instance, a person who has brown eyes is their phenotype, while the genotype is the gene that is responsible for such characteristic.

Proteomics is related with the study of the structure and functions of proteins in a large-scale. The focus of proteomics is a biological group called the proteome. The proteome is dynamic, defined as the set of proteins expressed in a specific cell, given a particular set of conditions. Proteins are a vital part of living organisms, as they are the main

components of physiological metabolic pathways of cells, i.e. networks of reactions and protein-protein interactions.

Metabolomics is the scientific study of chemical processes involving small-molecule metabolites, and their changes. Specifically, metabolomics is the systematic study of the unique chemical fingerprints that specific cellular processes leave behind, the study of their small-molecule metabolite profiles. Examples include antibiotics, pigments, carbohydrates, fatty acids and amino acids.

In the last decade, much work has been done in genetics such as the Human Genome Project (HGP) and other genetic research projects. Despite of such efforts, the genetic information is not a unified concept. On one hand, it encompasses skin's or eye's colour, sex and height of an individual. On another hand, it also includes diseases genetic component and their all abnormalities.

A common denominator for omics research areas is that they produce large amounts of data. In fact, there are many databases with omic data. In most cases, the databases are accessible through the web services with a specific API or flat files that can be parsed. The most popular databases[1] contain disperse data of each omic component, i.e. genomics, proteomics, and metabolomics. A description of the main omics databases will be presented in the sequel, i.e. mainly focusing on pathway, genetic and literature.

The KEGG (Kyoto Encyclopedia of Genes and Genomes) is a bioinformatics database resource for understanding high-level functions and utilities of the biological system that links genomes to life and the environment. The main goal is to collect genomic information relevant to metabolic pathways and organism behaviours [38]. It deals with genomes, enzymatic pathways, and biological chemicals [39].

Reactome [40] is an open-source and manually curated pathway database that provides pathway analysis tools for life science researchers. The largest set of entries refers to human biology and a few other organisms as well. Pathway annotations are a collaborative work between biology experts. It contains much visual information from textbooks and scientific articles.

PID (Pathway Interaction Database) [41] is a US National Cancer Institute and Nature Publishing Group initiative containing a highly structured collection of information about known bio molecular interactions and key cellular processes assembled into signaling pathways. Their users are from cancer research community and other entities interested in cellular pathways such as neuroscientists, developmental biologists, and immunologists.

The Universal Protein Resource (UniProt) [42] provides the scientific community with a single, centralized, authoritative resource for protein sequences and functional

[1] http://www.oxfordjournals.org/nar/database/cat/3

information. Interpro [43] is a competitor of UniProt. It is an EBI (European Bioinformatics Institute) database with focus on proteins and the proteome. The dbSNP [44] stores information about Single Nucleotide Polymorphisms (SNP), particular changes in our genetic sequence that are relevant for the detection of anomalies in our genes.

The Mendelian Inheritance in Man (MIM) collects diseases that are mostly caused by genetic syndromes [45]. Medical Subject Headings (MeSH) [46] is also a disease database that is correlated with other databases such as the Medical Literature Analysis and Retrieval System (MEDLINE), a huge bibliographic database of published material referred to life sciences and biomedicine. The entire MEDLINE library is accessible through PubMed[1], an online search engine.

2.2 Data access

The first step to integrate biomedical information is the establishment of a channel for accessing to data. However, data sources provide information in different ways, depending on several factors such as data type, or the institution generating them. Furthermore, these records are stored in many distinguished ways, as described in section 2.1. Despite of the standards that are being developed to integrate health information systems, medical data access is still a very complex matter, mainly due to privacy reasons. The confidentiality of patients' records is a social and ethical-legal issue. Thus, medical images and reports are considered valuable information for many entities, including hospitals, doctors, researchers and insurances companies [47]. Considering the importance of these data, healthcare institutions must guarantee that medical records are safeguarded. Moreover, the access to clinical data depends on each countries' laws, hospital committees and even a high level of security implies dealing with bureaucracy and other political issues.

The growing importance of biomedical research in the discovery of new treatments shows that access to clinical data is of crucial importance to improve and develop new drugs. Data access for researching purposes is usually controlled according to a predefined agreement validated by an ethical-legal commission. It will allow exporting data, but with several restrictions like, for instance, the database access is only provided inside the institution, the process is monitored and granted for a short period of time.

Data sharing is also very important in health integration of data, though those systems have to guarantee data protection and privacy. There are several aspects that should be taken into account. Clinical trials, diagnostic data and medical reports must be stored with privacy and only accessible to authorized people, even for de-identified data. The transfer

[1] www.pubmed.com

of data has to be confidential, avoiding man-in-the-middle attacks or other ways to access the information.

2.3 Data storage

The biomedical data universe handles large number of documents. Thus, it becomes necessary to evaluate which is the best and most efficient solution to store them, and which features are required for our particular problem. NoSQL databases were designed to efficiently handle large amounts of data, at the expense of rich features available in relational databases. Despite of good methods to scale a solution, deployment of real solutions still need hardware to be tested and deployed.

With the advent of the network and global connectivity, there was an explosive growth in outsourcing of storage to remote locations. Nowadays, there are many providers offering these services, for instance, Amazon, Google and Microsoft. These data centres have large capacities, but dealing with ultra large scale has become a major challenge. There are enterprise solutions like NAS or SAN storage, although, the major problem with these solutions is the high maintenance costs and the heavy machinery requirements.

Computing devices and Internet access are now available anywhere and at anytime, creating new opportunities to share and use online resources. A tremendous amount of ubiquitous computational power, such as Google and Amazon, and an unprecedented number of Internet resources and services, such as email and storage, are used every day as a normal commodity. In addition, Internet bandwidth is plentiful, which allows online data storage and time-efficient remote access from anywhere.

A Cloud computing service consists on the aggregation of distributed resources into one single virtual system, aiming for virtualization, i.e. decoupling the business service from the infrastructure, and for scalability, i.e. the system capability grows as it is needed [48]. Besides, one of the great advantages of Cloud computing is its resilience. In theory, Cloud computing services are built in such a way that, if a machine fails, the system readjusts itself so the user will never know that one machine failed. Taking this into account, Cloud computing is a promising technology to ensure a level of stability that a single server cannot provide. Moreover, the costs saving on a local datacenter infrastructure (hardware, software, air conditioning, fire alarms, physical security, etc.) are tremendous, including continuous IT updates, licensing aspects and electricity consumption.

It is evident that the computing-as-utility business model is becoming prevalent in the electronic world and numerous institutions are adopting it. However, there are also some important weaknesses that must be considered when someone decides to migrate an existent solution (infrastructure and/or application) to a public Cloud provider [49]. One of the most important is the latency introduced with factor distance, a relevant aspect in

huge volume data transfers. It is true that broadband Internet links minimize this aspect. Nevertheless, Internet latency times cannot be compared with values obtained in scenarios where servers and clients are located in the same Intranet. Another problem is the lack of interoperability between providers, i.e. services are not interchangeable across Cloud providers. The current absence of interface normalization does not allow transparent migration of Cloud applications between providers. Finally, a critical concern is related to security, namely data privacy, durability and availability. The sharing and dynamic resource allocation of Cloud computing reduces user control over proprietary data and poses new security issues, when compared with a traditional application server hosted behind an institutional firewall. Cloud provider selection is important to reduce some security risks. For instance, the Amazon Simple Storage Service (S3) was planned to provide 99.99% availability of objects over a given year, excellent values when compared with Intranet datacenters. However, the quality of the client Internet link service is fundamental to ensure data availability.

There has been great investment in building Cloud computing infrastructures for health purposes. For instance, Harvard Medical School built an internal computing Cloud to enable collaborative research among several departments and partners. Another example, TC3Health Company, is already providing healthcare players with an integrated solution supported by Amazon Web Services, namely S3, EC2, and SQS technologies [50].

Dynamic scalability is one of the principles of Cloud computing, mainly focused on databases. The goal is to, automatically, scale our databases with low data access latency and providing an interface with an easy programming model. NoSQL database scales nearly linear with number of servers used. This is possible due to data partition, mainly using technologies like DHT (Distributed Hash Tables). This system is based on a couple of keys and values hashed into buckets and partial storage spaces, each one placed on a network node.

The horizontal scalability allows dividing the computation into concurrent processing tasks. In order to tackle this challenge, the industry started investing on distributed storage file systems to support large datacenters. Several examples of this reality can be identified like, for instance, the Google File System/Big Table, Amazon S3, Cassandra, Hadoop, and many others. The next subsections will explore the main techniques for storing data in large-scale.

2.3.1 Relational Database Management Systems (RDBMS)

Relational databases [51] are the most common solution to have persistence of structured data. The proposed model allows us to represent the data in tables with attributes and relationship between them, providing mechanisms for ensuring data integrity through the usage of primary (entity) and foreign keys (relational). A huge advantage of the relational

model is the minimization of redundancy, known as normalization. This technique permits the table to be decomposed in order to produce structured relations. Thus, data is decomposed into smaller tables, each uniquely identified by a primary key and linked through foreign keys.

Traditionally, RDBMSs have not achieved the scalability required for large systems. In the last decade, the applications had to deal with more quantity of resources and the queries answers should be given in a short period of time. Thus, database engines started to support distributed strategies to perform faster and support larger amounts of data. The first approach was based on data replication, i.e. duplication of records for more than one database. Therefore, the core database can increase the read performance, since the operations can be distributed over the cluster database nodes. Nevertheless, the write operations become more complex, as they need to be synchronized among all nodes. There are mainly three models of distributed databases: master-slave replication, multi-master replication and sharding [52]. The master-slave is the basic way of replication. The master owns all database objects and has exclusive permission to modify, add or remove them. The master replicates the operations to all the slaves and, if some operation fails, he will re-execute again. The main problem with this architecture is that the system cannot operate when the master fails or is inaccessible. This situation does not occur in the multi-master replication approach, because any node can be responsible for updating and propagating changes to other nodes. Nonetheless, this model has also some problems with synchronization, since keeping updated records to every single operation has performance costs. Normally, they use buffering or caching mechanisms with a defined number of operations. The major benefit of this model is that the system continues to operate when a node becomes offline. So, its objective is to have better data availability, not to improve the system scalability. Finally, sharding means that the contents are distributed between several nodes. In this model, the database is not in a single machine, but dispersed over a set of machines. The database content splitting has the advantages of offering data balancing between cluster nodes, each one with less storage requirements. The availability of data also increases and, when a node crashes, only one part of data is compromised. Nevertheless, sharded databases do not support join operations (i.e. tables data aggregation), because these are used to produce data from two data sets, connecting them by a common attribute. Sharded databases lead us to explore NoSQL databases, in section 2.3.2.

2.3.2 NoSQL

NoSQL stands for “Non SQL” or also as "Not Only SQL" and consists on databases that do not follow a traditional relational model. They have adopted different approaches to store data, which provide a more efficient way to handle unstructured data such as word-processing files, email, multimedia, social media [53].

A key point of NoSQL databases is the horizontal scaling, i.e. replication and partition of data over many servers [54]. Nevertheless, NoSQL is not compliant with the ACID (Atomicity, Consistency, Isolation, Durability) properties that are usually supported by relational databases to ensure the reliability, consistency and independency in concurrent database transactions. Some authors suggested that NoSQL are BASE: Basically, Available Soft state and Eventually consistent. [54]. The main idea is to achieve high performance and scalability. Nonetheless, there are already NoSQL systems that provide some degree of consistency based on multi-version concurrency control [54] [55].

There are several types of NoSQL databases, and the most important will be described next:

Key-Value Store: The simplest storage model that is similar to a hash-map memory, where the data (i.e. the Value) is stored and retrieved through the usage of a unique Key element. These systems generally provide replication, versioning, locking, transactions, sorting and many other features. The client's interface provides insert, delete and index lookups.

- **Column Store:** It is a storage system where the related information is grouped by columns, promoting a physically adjacent storage of similar information. For instance, the column Citizen may store the Name, SSN number and Birthdate attributes. The major benefit of this approach is when the database has a large number of rows and less columns. In the relational model, the information is grouped by rows. Thus, when updating or accessing a specific column on several rows, several disks might be seeking to find the correct column in each row. In this kind of data, one operation will find the column.

- **Document oriented:** It is a refinement of the key-value store model where the value (i.e. the data) can be a formatted document. These systems could store documents in the traditional sense. However, a document can also be a "pointless object". It supports multiple types of documents like, for instance, Microsoft Word, PDF and text files. Moreover, many implementations are inclusively supporting JSON and XML documents.

Despite of the concepts, many implementations of these types of databases have been raised in the last years. Project Voldemort [1] is an advanced key-value store, open source and written in Java. It supports sharding data and multi-version concurrency control for updates [54]. The major drawback is the asynchronous update of replicas, and not guaranteeing data consistence. Nevertheless, if an updated view for the majority of replicas is requested, a consistent view is assured. The sharding support is automatic, i.e.

[1] www.project-voldemort.com

the system adapts itself if nodes are added or removed from the cluster. This database is used by LinkedIn and receives many inputs from their team. Riak [1] is an open source key-value database written in Erlang (i.e. a functional programming language), which uses JSON as database objects with support for multiple fields (owever, queries are limited to the primary key, i.e. the hash key). Database access interface is through a RESTful API. Riak supports insert, delete and lookup operations and provides scalability through distribution of contents over distinct nodes (replication and sharding).

CouchDB is an example of an open source document store database that was incubated from Apache Foundation Software. It is also written in Erlang and the documents can have values in text, numeric, boolean and lists. Moreover, it supports the creation of indexes for values of the documents using B-Tree structures [56]. Thus, besides improving the performance of queries using those attributes, it can also include restrictions and the results can be ordered or value ranged. CouchDB does not guarantee consistence, but it implements a multi-version concurrency control on individual documents and, for each version of the document, a sequence identifier to avoid inconsistencies is created. Finally, the system provides durability even when it crashes. All updates associated to document operations are written to the disk by default. The interaction API is also through REStful webservices.

MongoDB is very similar to CouchDB with some slight differences. It supports automatic sharding and provides atopic operations for fields, while CouchDB only supports for the documents, but does not provide global consistency. Unlike the key-value stores, the document store provides a mechanism to query collections based on multiple values constrains. The documents are distributed over nodes, thus, achieving scalability by reading the replicas values [57].

Cassandra is an open-source columnar store database, written in Java. The column updates are cached in memory and then flushed to disk. Cassandra automatically brings new available nodes into a cluster [58]. The major drawback is the weak consistency, because they do not have any lock mechanism for transactions. HBase is another database written in Java and maintained by Apache Software Foundation. It is similar to Cassandra, but the big difference is the focus on strong concurrency [59].

In bioinformatics, there are several examples of applications that use NoSQL databases [60] [61] and the main justification is the solution scalability that fits well those environments with high throughputs.

[1] http://wiki.basho.com/Riak.html

2.3.3 Cloud Storage-as-a-Service

Storage-as-a-Service is the ability to offer virtual remote storage service for any operating system and application. Nowadays, Cloud providers are offering storage using the Blobstore concept, which, *per si*, is not new. In the past, these concepts were used in Database Management Systems (DBMS) in the storage and movement of large data blocks. Blobstores are associative memories, i.e. key-value storage providers, where the blob is unstructured data (value) stored in a container and the lookup is performed through a text key. A container is a namespace or domain for the blobs. A blob is always uploaded to a container. Blobstores have a list of containers where the developer can create, remove or rename them. The container holds content, which can be blobs, folders or virtual paths.

Cloud providers have released the blobstore service that allows their customers to store data in a container over the Cloud. For instance, Amazon S3, Microsoft Azure and OpenStack Swift have blobstore APIs. These services are considered Software-as-a-Service (SaaS), because they allow developers to take advantage of the remote storage service to support their application data in a transparent way, without worrying about scalability. There are many examples of SaaS use, for instance, the Dropbox application that stores customers' files in Amazon S3 or commercial web portals that store great quantities of pictures in cloud blobstores.

2.3.4 Cloud Database-as-a-Service

Database-as-a-Service (DaaS) is a new paradigm that outsources database to the Cloud. Therefore, the database is hosted in a remote datacenter and shared between users in a transparent way. For instance, Amazon AWS, Windows Azure [62] and Rackspace [50] offer this service in a pay-per-use business model. Common database operations are supported by these services, for instance, creating tables, loading, and accessing data in the tables. Cloud providers often supply an API to access the database, and execute operations through a web service API.

Furthermore, database maintainers do not need to worry about the server's redundancy, upgrades, back-up plan and recovery from disaster. Nonetheless, some enterprises are concerned about ensuring data privacy. In fact, this is one of the weaknesses of DaaS. Despite Server Level Agreements (SLA) by Cloud Providers, there is no guarantee that an intruder in the Cloud company does not access the clear data. In addition, store procedures and triggers might not be supported in the overwhelming majority of Cloud providers supplying DaaS. Finally, performance might deteriorate, since the applications may be running in a remote location.

SimpleDB is part of Amazon Cloud services offering, along with EC2 (Elastic Compute Cloud) and S3 (Simple Storage Service) on which it is based. It is a simpler database and

supports nested documents using pointers to Amazon S3 object locations and related metadata information. This database supports data consistency, but not transactional consistency. Moreover, it supports asynchronous replication, without any relevance for the end-user. This database is similar to other document oriented stores, and supports more than one group in database, i.e. the documents are organized in domains, and support multiple indexes. Domains and their specific metadata can be enumerated. Regarding the distribution of data, for instance, data from different domains might be stored in different nodes of Amazon. The Amazon Relational Database Service (RDS) is also a database service over the Cloud that easily allow us to create, manage, backup, and scale MySQL database instances.

Azure Table Storage is also a Database-as-a-Service provided by Windows Azure Cloud. The main differences between SimpleDB and Azure are the conditions. SimpleDB conditions are in each item, while in the Azure they only support Batch transactions in the same table and partition group.

2.4 Data integration architectures

The main goal of data integration is to fetch data from distributed and heterogeneous sources that may include normalized and proprietary systems, even unstructured formats, that will be organized and offered through a unified data view [63] [64]. For this doctoral program, there was a need to integrate different data sources, over different locations. This section will present the standard architectures that have been applied for data integration in many areas, including the biomedical.

2.4.1 Replication

The replication, or physical data integration, is a method that captures changes in the originator data sources, copy and integrate them in one or more destination databases. The replication can follow distinct strategies: data or operations replication. In the first case, an application reads the data changed in the original source and updates the destination databases. In the second one, the application captures the operations performed in the original system and executes them over the destination database data.

The main advantage is the availability of data for medical emergencies and epidemiological studies. Since the information is gathered in a single structure, the complexity of the implementation of security services is low. The main disadvantage of this approach is the high complexity of its technological implementation, involving the existence of a large-scale centralized infrastructure with high storage and communication requirements [65, 66]. Moreover, a possible issue with replication of data is guaranteeing their integrity. Data should be protected against changes when they are only a simple

copy. In bidirectional replication, a strategy of solving conflicts should be well implemented.

2.4.2 Data warehousing systems

The data warehouse systems allow integrating data from multiple databases. These techniques are a set of algorithms and architectures that permit selecting and incorporating data from multiple databases into a single one (Figure 2.5) [67].

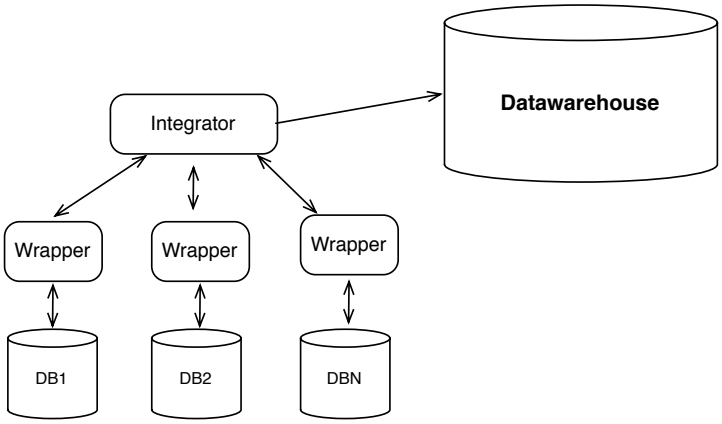

Figure 2.5: Data warehouse general architecture. The DB1, DB2 and DB3 are distinct datasources. The wrapper is responsible for do a translation of the data to the integrator. The Integrator sends the data to the data warehousing system.

The main advantage of these systems is its capacity to deal with large datasets and work in a centralized manner. However, there are also a few disadvantages, mainly regarding their space requirements, i.e., it might become hard to scale and the data that must be cleaned, transformed and loaded are often duplicated.

There are two important components in data warehouse systems: Wrappers and Integrator (Figure 2.5). The wrapper is responsible for the translation of the different databases and sending the information to the integrator [68]. The integrator receives the data and stores them in a normalized way. Moreover, the integrator is responsible to keep the information up-to-date and receive notifications from the wrappers when data changes occur.

ETL (Extract, Transform, Load) is a process involving the collection and aggregation of transactional data from multiple sources, to be used in databases for reporting and analytics. The idea is to extract data from several systems, transform these data into business rules and, finally, load them in a data warehouse.

The first step is the extraction of data from the source system. This process can acquire data from different systems, including different formats. Those sources are typically relational databases or flat files. During this process, data are converted to a more "standard" format. Next, the transform stage applies a set of rules to the extracted data and sends them to the load stage. The kind of functions or rules is dependent of the type

of data and the information supposed to be achieved. For instance, transform coded values into more standard ones.

Finally, the load stage loads the values to the Data Warehouse. There are many strategies to accomplish this step: some of them can replace the existent values; others can add new ones and do cumulative processing to calculate updated values.

2.4.3 Federation systems

Federation is a data integration model that combines disparate data into a common logical data structure, not by moving data, but by providing a uniform view. The main idea is connecting disparate database technologies through a "bridge" that provides a "virtual" view of multiple databases. The resources can be homogenous or heterogeneous, local or distributed, depending on the implementation [69]. The uniformed way can be achieved with an interoperable strategy, based on wrappers that solve integration technical issues (see Figure 2.6).

The main advantage of this model is the fact that there is no duplication of information, since all data is stored in the original data systems. This avoids duplication and data synchronism, while also limiting the need for data transfer, since, typically, only a limited part of the full record is accessed. The biggest problem of this model is associated with possible unavailability of part of the data, depending of the status of the remote data source system and communications network.

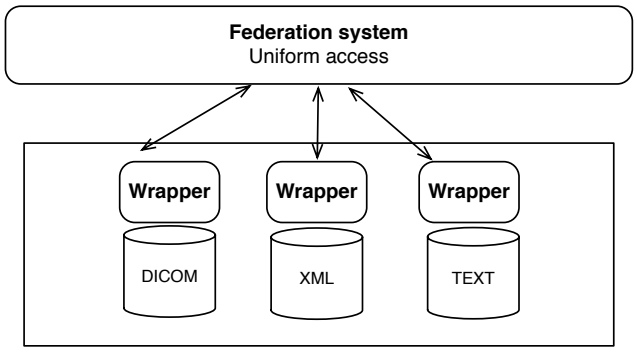

Figure 2.6: Federated integration architecture

2.4.4 Service Oriented Architecture

Service-oriented architecture (SOA) is a set of principles and methodologies to create solution logic units, which are independently shaped, so that they can be cooperatively and repeatedly used to support the achievement of specific strategic goals [70]. SOA is known as a transactional data integration pattern that composes messages to instantiate objects that will perform at different levels on a common network interface called a service bus [71]. These objects represent functional business components, which are created or instantiated at different layers of granularity. In fact, the business logic is very

well defined to simplify the reuse of a service for different purposes. The architectural model aims to improve the agility and cost-effectiveness of an enterprise, while reducing the IT burden on the organization. [72]. SOA enables an establishment of well-defined trust relationships, which define what is trustable, in what conditions, through which communication methods. These relationships are usually potentially passive of liability issues. Dynamic federation is the real-time establishment of such trust relationships on a need-to basis. Most telecommunication systems currently rely on static federation**,** an off-line process for the establishment of such trust relationships.

SOA supports many strategies to integrate multiple services (Figure 2.7). Service Composition is an aggregation of services collectively composed to automate a particular task or business process. An Orchestration establishes a business protocol that formally defines a business process. It can centralize and control multiple application logic through a standardized service model. They have several similarities with a federated system, previously described. Choreography is a complex activity comprised of a service composition and a series of Message Sequence Patterns. They involve multiple participants, who can assume different roles and have different relationships. They are both components for an effective coordination of multiple services.

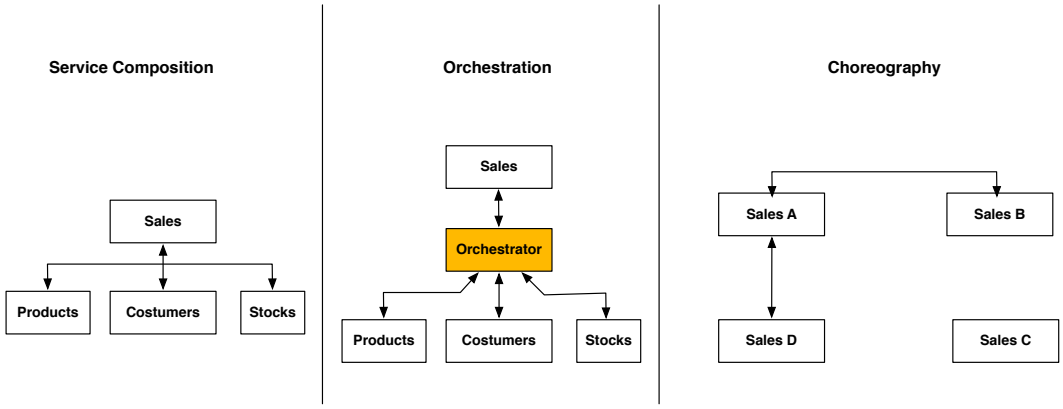

Figure 2.7: Service Oriented Architecture - different types of service integration

2.5 Knowledge extraction

During the past decades, impressive amounts of data have been gathered in healthcare centres, with the aim to store patients' clinical history and allow the extraction of new knowledge from existing information. Unfortunately, these data repositories are stored in distinct formats, with different terminology, taxonomies, languages and many others. The Information Retrieval plays a significant role in data management strategies, as a method for extracting, indexing and searching over biomedical data. On top of these systems, it will be possible to create mechanisms for knowledge extraction. Nevertheless, these processes are not trivial, due to large amount of data that need to be stored in a structure, with easy access, and ready to scale.

In the recent years, an increasing interest has been paid to semantic information systems. Search engines, for instance, have adopted semantic solutions to enhance their user interfaces. When querying for a specific concept (e.g. a personality, a city, or a country), these engines already provide several semantic elements associated to that concept. This type of mapping is possible due to the increasing number of available ontologies. An ontology describes a set of classes, which are terms or entities, and relationships among them. Several W3C initiatives (World Wide Web Consortium) have produced standards and recommendations for the semantic web, such as RDF (Resource Description Framework) [73], OWL2 (Ontology Web Language 2) [74]. Ontology represents knowledge from a set of concepts that belongs to a domain. In the medical domain, there are already a few ontologies that can be useful to map the concepts related to medical images content and their associated information. RadLex is an initiative from RSNA to help radiologists having a unified language to organize and retrieve images and reports. As previously discussed, radiologists and, in general, the medical community, use a variety of terminologies and standards, but no single lexicon serves all of their needs. RadLex is a lexicon, unified language, with radiology terms for standardized indexing and retrieval of radiology information resources. This ontology has more than 30,000 terms. The RadLex ontology also has the mapping to the IDs of the UMLS, FMA and SNOME-CT. For instance, the FMA (Foundational Model of Anatomy) is an ontology that contains several concepts regarding the human anatomy.

While semantics are important, there are also other important features required to allow the user to extract knowledge from the stored data. These include aggregation, visualization and dashboard. .

2.6 Biomedical data integration cases

The development of efficient systems to process and summarize information from multiple sources associated to distinct areas of medicine is an important and persistent subject in biomedical research [13]. Clinical and translational medicine is fundamental to promote the flow of information between basic and clinical scientists, to improve new biotechnologies and to enrich patients' health quality [75]. Biomedical informatics gathers a set of methodologies in translational medicine. It intends to support the transfer and integration of knowledge across the major areas of translational medicine, from molecules to populations [76].

The art of using science to create synergies between different medical disciplines promises to improve the patient treatment, to better understand the interventions used in clinical practice and to help in the development of new policies and guidelines.

There are many areas that fit on the translational medicine such as bioinformatics, imaging informatics, clinical informatics and public health informatics (Figure 2.8). The

bioinformaticians need to identify the molecular and cellular regions that can be targeted for specific clinical interventions and to provide treatment for specific diseases. Imaging informatics plays a significant role in understanding pathogenesis and identifying treatments from the molecular, cellular, organs level and tissue. New methods to visualize and analyse these data will be needed and are already being developed [77]. Clinical informatics innovations are necessary to improve the patient care through the availability and integration of relevant information. There are many methods being currently applied to medical reports information such as text mining and knowledge extraction. Public health informatics is the area that intends to analyse and study new techniques to investigate data in a large scale. Translational medicine teams need to address many challenges, focusing on Decision Support, Natural Language Processing, Standards, Information Retrieval and Electronic Health Records.

There are efforts to integrate health data from multiple institutions, correlate them with genomic information and integrate knowledge from scientific literature. Nevertheless, the integration of some biomedical data is still not applied in practice. In this section, a state of the art of the solutions to integrate biomedical data will be provided, discussing the pros and cons of each solution.

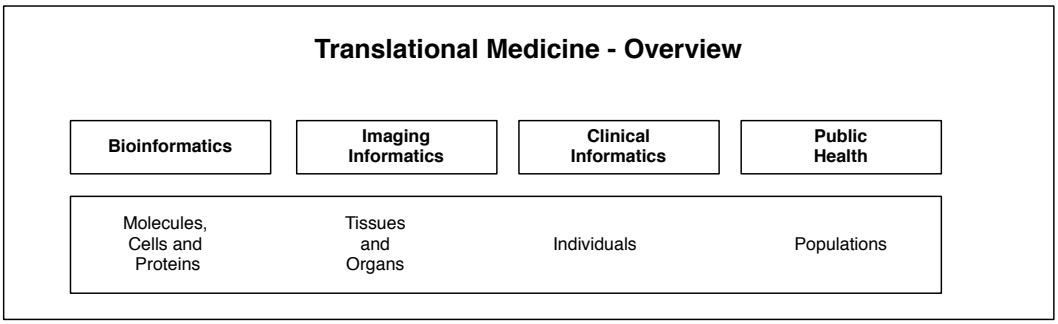

Figure 2.8: Synergies across biomedical informatics towards translational medicine

2.6.1 Integration of medical records

Although efforts have been made to create Electronic Health Records in medical institutions, the integration of those systems still is an open issue, especially in inter-institutional environments. Nevertheless, it is possible to identify several efforts aiming to provide integrated views of EHR, mainly for research purposes.

Pangea-LE [78] is a middleware intelligence that integrates biomedical information. It is an integrator between health data sources and health professionals. It allows a structured XML view over distributed repositories. The databases are designed according to information system's requirements and were already deployed to support RIS or HIS. The changes necessary to support the integration are independent of the database schema. The main components of this architecture (in Figure 2.9) are the adapters and biomedical data

extractors. The adapter is a wrapper that understands the organization of data at one specific source, extracts this information and offers it to the system core, using a common interface. The biomedical extractors are responsible for the translation of the biomedical definitions from the original sources to a common and unified way. Pangea-LE was deployed in the Consorcio Hospital General Universitario de Valencia (CHGUV), which is a medium sized hospital, with 592 beds, 21 operating rooms and over 500 physicians. According to the authors, the vast majority of patient health data is available electronically through Pangea-LE.

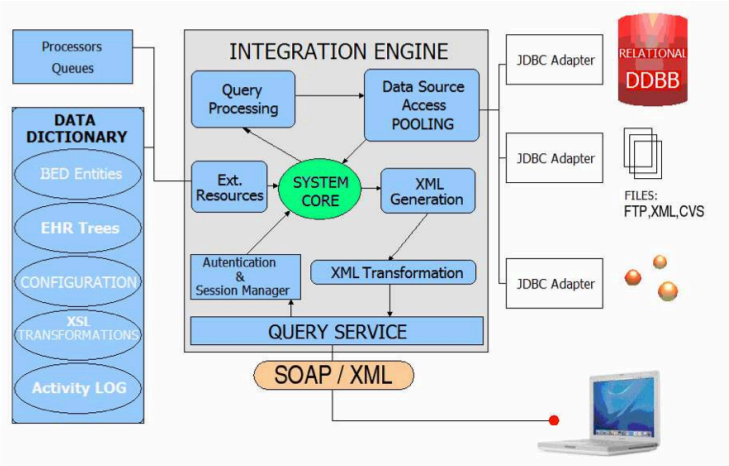

Figure 2.9: Pangea-LE architecture

ResearchEHR is another project [79] focused on the normalization and semantic aspects of EHR. The internal level deals with normalization and semantic upgrading of exiting EHR using archetypes. The external level uses Semantic Web technologies to specify clinical archetypes for advanced EHR architectures and systems. This solution is already being used in practical scenarios [80].

LinkEHR-ED [81] main idea is to transform data into knowledge, based on the OpenEHR standard [82]. It explores the use of archetypes as a mean to achieve standardization and semantic data integration of distributed health data. These archetypes are used to make clinical information publically available, in a standardized format.

The paper [83] describes a secure platform for knowledge discover from clinical data stored in distributed healthcare institutions. They use IHE standard to access the data, namely two important profiles: XDS and QRPH. The authors propose a portal where it is possible to access the clinical information, using a single-sign-on system and preserving the patient privacy. However, the authors only propose directions for privacy issues, not an implementation.

EHR4CR framework is playing a significant role on the integration of several medical data sources [84]. The objective was to create a solution with semantics interoperability for Clinical Research Document (CRD) context, i.e., a tool used in clinical research to

manage a clinical trial data. Its data sources are EHR. Fadly et al. [84] show some developed applications based on EHR4CR, point out some problems with semantic, and duplicate entries on the Clinical Data Management Systems (CDMS). They propose a single source entry in the clinical care context, which can be used in clinical research content. The solution reduces the efforts on data collection and elimination of double entry errors between two different contexts. Furthermore, they decrease the error rate in automatic matching of clinical data elements. The only drawback is that they focus on acquisition of information from EHR and not on other biomedical sources integration.

2.6.2 Medical imaging integration

Medical images are stored in PACS and, typically, serve one single institution. The data produced by digital modalities generate information on performed exams that can be used in different scenarios, for instance, to monitor patient dosimetry, radiographic procedures and image quality. Some important parameters such as image processing parameters, exposure index, patient dose and geometric information are generated by the modality and transferred to the PACS repository. Despite of this information being stored on the healthcare repositories, the traditional information systems are not able to access them. However, there are already tools for the integration of multiple institutional repositories, including the search over any DICOM attribute.

Dicoogle [26, 85, 86] is a free and open source alternative solution that allows extracting all textual information from the PACS archives, providing an enhanced query mechanism over DICOM metadata. Those query and retrieve facilities are transparently available over distributed Dicoogle repositories through a P2P network layer. Its analytics module provides a unique access point to search and query, to perform data mining and to extract knowledge, over distributed data sources. Finally, it follows a micro-kernel architecture, making it really simple to extend core functionalities through the development of new plugins.

CloudMed [87] is a Cloud integration platform that works in a federated way. The information is kept on the traditional PACS and, encapsulated on HTTP through the DICOM standard. It is possible to search and retrieve images from a centralized repository supported by a Cloud blobstore service.

In [88] the authors centralize the medical images in a common platform to allow the information extraction. The application is based on the Globus toolkit, an open source software to build GRID platforms, which final goal is to support clinical investigation.

2.6.3 Omic data integration

As expressed, there was an increase of datasets in clinical information and genomic databases, for instance, NCBI GEO, EBI Array Express, Molecular Brain Neoplasia Data

and caGRID infrastructure. Several tools to analyse genomic and clinical data have been developed for individual purposes. In the recent years, many efforts have been made to combine the biomedical data from different types of sources, for instance, biological and clinical data.

DiseaseCard [89] is an information retrieval tool for accessing and integrating distributed and heterogeneous medical and genomic databases, presenting it in a familiar visual paradigm. The main goal is to provide the user with an integrated view of the information available in the Internet for a specific disease, from the phenotype to the genotype, avoiding replication. It targets the rare diseases due to their high association between phenotype and genotype. It provides also a navigation protocol to guide the users during the process of retrieving information from the Internet.

Merck [13] developed a tool to study data from oncology trials between a cancer center and a researcher institute – Moffit Cancer Center and Research Institute. This proprietary system was built over SAS Drug Development, LabMatrix, MatLab, Microsoft BizTalk, Tibco and R. This platform addresses data sharing between two institutions.

Rembrandt [90] is a data warehouse platform to integrate genetic and clinical information. It was developed to support research, disease diagnostic and treatment, and is mainly focused on brain tumors. The system allows data query and visualization across different genomic databases. It is possible to perform analysis with data from distinct sources, including the possibility of using data provided by the researchers, for instance, to perform comparisons with the full dataset available in the system.

I2b2 [91] is a software platform to mine clinical data for cohort discovery and hypothesis generation in translational research, optimized for genomic studies. This system can integrate data from distinct systems and it is able to query and visualize data from several datasets. Thus, researchers will be able to inquire the existent clinical data to find more answers. This information can be combined with genomic data to create new research opportunities. The correlation of different data sources facilitates the design of therapies for patients with genetic diseases. It is an open source platform built over a pluggable architecture to support the extensibility of its functionalities.

GenePattern [92] is an analysis tool for genomic data that can perform gene expression analysis, proteomics, SNP analysis, flow cytometry, RNA-sequence analysis and data processing tasks [93]. It is oriented to genetics research and provides a web portal with many interesting tools that can be integrated to achieve a desired solution.

tranSMART is a data warehouse framework that allows efficient data access and mining associated with sample genomic databases [13]. The data warehouse contains structured data from internal clinical trials, experimental medical studies and a set of public sources. This platform allows clinicians, scientists and biologists to inquire aligned

phenotype/genotype data, enabling better clinical trials design or stratifying diseases into molecular subtypes. The scientists can create and test new hypothesis taking into account the biological processes, creating opportunities to develop better treatment options. The data modalities include clinical data and aligned high-content biomarker data such as gene expression profiles, genotypes, serum protein panels, metabolomics and proteomics data. It is based on i2b2, GenePattern and warehouse supported by Oracle databases. The integration with third tools is possible, mainly using web services technology to query the indexed data. It is possible to use R, a popular language on data and text mining area.

2.6.4 Final considerations

Despite several efforts that have been made to integrate biomedical data, there is still a need to integrate disperse health information with omic, literature and many other data sources. We presented several frameworks and projects that aim to facilitate the scientific research, using the medical imaging, reports, genomic information and clinical trials to extract knowledge from these sources.

As described in the state of art review, there are already several solutions developed to support biomedical research around the world, and that allow integrating several biomedical data sources. Therefore, many sub areas of research are able to grow, as well as the tools to improve the clinical decision in medical imaging, quality assurance, treatment improvement and drug discovery. One of the most common problems that researchers who want to explore biomedical data face is to the way to manage these data and the considerable amount of time it takes, mainly to find the proper data source and include it in their study. Moreover, not only if they consider access to *M* databases but also if they decide to go across information systems of *N* healthcare units. The patient clinical history is an important contribution to improve the scientific research and, therefore, it is fundamental that it is as much complete and normalized as possible. Translational research can have an important role in this scenario. It may promote the integration of tools available in the distinct areas such as medical imaging, EHR, omic, literature and many others. Following the highlighted problems in current state of the art, three proposals will be presented and discussed, as stated in section 1.2. Firstly, the integration will be mainly focused on medical imaging targets. Secondly, a broader approach will be considered, including patient information such as diagnosis report. Finally, a more high-level integration and aggregation that will be able to unify and drill down in *M* types of databases of *N* healthcare units will be discussed.

3 Integrating medical imaging repositories[1]

"We can only see a short distance ahead, but we can see plenty there that needs to be done."
Alan Turing

The initial problem statement of this thesis aims to study and propose architectures for the integration of *m* databases of *n* healthcare units. While the problem is far from simple, in this chapter we will address the integration of medical imaging repositories in *n* geo-distributed healthcare units. The medical imaging scenario is very complex and presents many specific challenges [16, 94, 95], reason why it is usually treated separately from the HIS.

Collaborative work environments have greatly increased within healthcare units, in the past decade. This trend has changed the procedures in healthcare institutions and the exchange of medical imaging across institutions has become common in several

[1] This chapter is mainly based in the publication *DICOM Relay over the Cloud, International Journal of Computer Assisted Radiology and Surgery. Springer, 2012 [16].*

modalities [96]. Their importance has increased due to cost-savings for medical institutions and growth of applications such as expert consultation, cooperative work and sharing of images between multiple image centres.

Nowadays, PACS is one of most valuable tools supporting medical decision and treatment procedures. A PACS is a key point in storing, retrieving and distributing medical images in the various steps of clinical practices. Despite several institutions use of DICOM standard to distribute medical images, the inter-institutional usage of this standard is mainly supported by VPN connections.

Although the DICOM standards support SSL/TLS layers, i.e. encrypted channels that allow privacy in the transfer of electronic data, many medical devices do not support these features. This discourages users located outside an institution from securely accessing the PACS archive, only using DICOM with direct connections. Medical institutions often use VPN to share medical resources. However, this solution requires point-to-point configurations, which are not scalable. Other ways of exchanging exams between medical institutions include, for instance, through CD/DVD, by conventional mail, shared link, or by email. These solutions rely on manual processes, which cannot be considered efficient in a normal diagnostic workflow. There are several proprietary solutions that do not follow standards, compromising the interoperability with other equipment.

Cloud computing is widely used to share files over the Internet and allows users to communicate with each other using external infrastructures. This technology permits the access to applications and data with minimal infrastructure inside medical institutions [97]. However, some important issues must be considered regarding the implementation of a solution (infrastructure and/or application) in a public Cloud provider [49]. Namely, there are critical concerns related to data security, privacy and availability.

The main purpose of this chapter is to study and propose an architecture for inter-institutional communication of medical imaging studies, allowing the integration of multi-source documents and the establishment of shared workflow. The proposed DICOM relay service aims to be a communication broker, allowing the search, storage and retrieval of medical images within a group of healthcare units, in different sites. This solution permits, for instance, remote access to the institutional PACS archive for storage or search/retrieve studies. Communication between different islands is supported on the public Cloud services and follows Web 2.0 paradigm, due to the use of the HTTP channel. Moreover, it keeps the interoperability of the devices adopted by the medical community. The proposed DICOM routing mechanism has a transparent application for the end-user, without any breaks in current standards used by medical imaging devices and repositories. Finally, the architecture provides several security services associated with connections.

3.1 Medical imaging laboratories

3.1.1 Data structures

As stated in section 2.1.3, there is already a standard to allow communication between the medical imaging devices supporting the data structure and communication. DICOM supports different kinds of information, including different modalities of images, reports and waveforms. Besides the image pixel data, a DICOM file contains a metadata header that includes information related to patient, clinical staff, medical institution, acquisition device, conditions of exam, clinical protocol and much other relevant clinical information.

An example of DICOM file is shown in Figure 3.1. The metadata header contains a varying number of fields according to the study modality. Several fields are identified in the standard DICOM Information Model [98], which outlines fields that are mandatory, optional and conditional in every DICOM file.

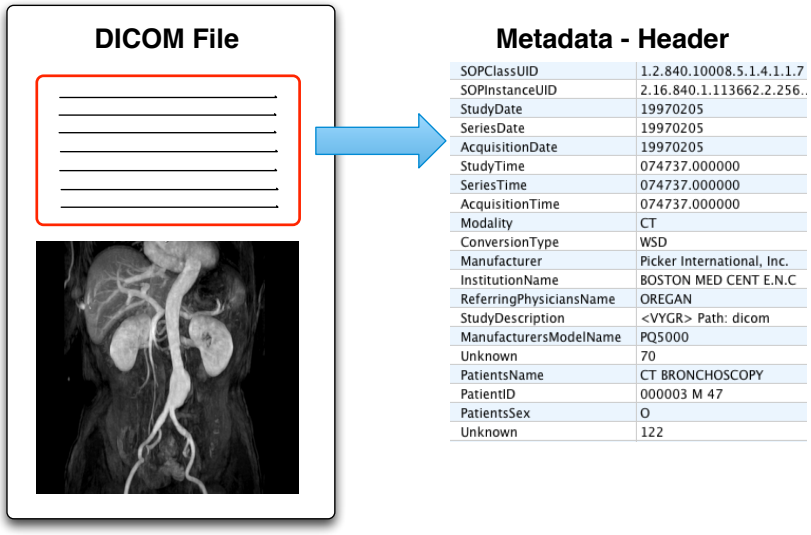

Figure 3.1: DICOM object with metadata and image. The header is zoom in (right hand side) and illustrates some attributes of the object metadata.

The DICOM objects and protocol communication messages follow a TLV structure (Tag, Length, Value), as shown in Figure 3.2. The tag is a pair of two values, a 16-bits unsigned integer, representing the group and element number. Each tag is unique and there are no duplicated tags in the same file. There is another field named Value Representation (VR) that covers how attributes are encoded. It contains the code that describes the type of element, e.g. PN for Person Name, DA for Date and OB for Object Binary. This label is optional, because the type of element can be reached using a DICOM Dictionary, i.e., for each normalized tag the dictionary defines what type it stands for. Length of attribute may vary for each tag, so, the field length defines the size of each attribute (value field). This size uses byte scale. Finally, the value field contains

the element used to store attribute contents (e.g. image pixel data, patient name, and many others).

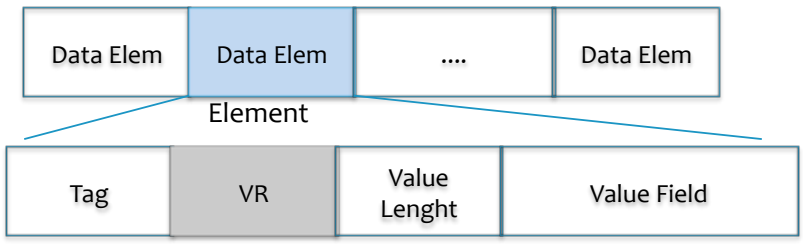

Figure 3.2: DICOM data structure - Tag-Length-Value

To encode or decode a DICOM object it is necessary to know the Information Object Definition (IOD). IOD is very similar to the templates approach, which means that each DICOM object is carried out by specific IODs in a high level definition. The IOD idea is to represent the most common data types in medical digital imaging. IODs are hierarchical structures and one single IOD may be a subset of other IODs. For instance, the IOD of a CT image modality contains few IOD like the "Patient Information Object Definition". However, this sub IOD is also used by other modalities. A SOP Instance is an instantiation of an SOP Class UID that contains IODs, which means that it has real attributes representing the definition. Moreover, DICOM objects may include also private tags defined, for instance, by modality manufactures, increasing the flexibility of file/communication configuration. Thus, its structure is flexible enough to be easily extended and the DICOM parsers works also for proprietary structure, decoding any kind of DICOM object even without understand the content of some attributes.

3.1.2 Communications

The networking specification of DICOM comprises an application layer protocol that uses TCP/IP to move data through the network and an addressing mechanism based on Application Entity (AE) [99]. It is a well-defined binary protocol that allows us to store, search and retrieve examination using the commands DICOM C-STORE, C-FIND and C-MOVE [100]. DICOM provides also a part that is complaint with Web 2.0, namely the WADO-RS, which allows images retrieval over the Web. This part was recently extended to include also storage and search mechanisms [101] based on REST (Representational State Transfer) Web services: the STOW-RS [102] for storage of DICOM objects; and the QIDO-RS [103] that supports queries based on DICOM Objects IDs. Thus, more web applications can access the information through the web resources API. Nevertheless, current infrastructures still rely on commands over TCP/IP and the Web 2.0 protocol layer is still not used. Overall, a wide domain composed of several medical institutions is still a challenge in what concerns data integration from multiple sources.

To communicate with a DICOM device, the first step is to purpose an exchange of information called DICOM association. In this procedure, devices negotiate several parameters, such as what kind of information will be transferred, how it is encoded and the duration of the communication. After the negotiation, the service commands are executed between SCU (i.e. client) and SCP (i.e. server) to perform the service goal.

Storage is a service that allows SCU to store images in a PACS Archive using the C-STORE command. Basically, the modality, or image generator, (i.e. Storage SCU) sends the images to the PACS archive (i.e. Storage SCP) - Figure 3.3. For each image, a C-STORE Request is invoked. All the contents of the DICOM objects are inside the C-STORE request message. A C-STORE response is sent from the Storage SCP after the file is received.

Query/Retrieve is a service composed of two commands. Query allows the SCU (i.e. workstation) to search for a study or patient, using the C-FIND command (Fig 3.4). The workstation can search over the image archive using several fields like, for instance, patient name, study date and modality. Fig 3.4 illustrates a query action, which looks for exams performed "today" with names starting with A, and two studies were retrieved (Antonio and Ana) in the response.

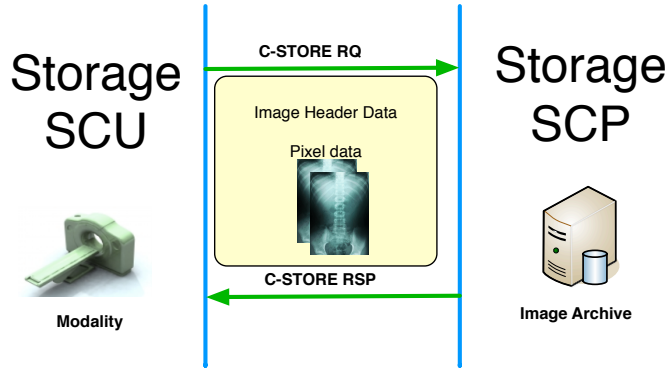

Figure 3.3: DICOM storage service. The Storage SCU is the client that is transferring a set of images to the PACS Archive (Storage SCP)

Finally, the retrieve method allows the SCU (e.g. workstation) to get/move image from the SCP (e.g. image archive) - Fig 3.5. The retrieve operation uses the C-MOVE or C-GET command. The C-MOVE is a retrieve command that uses a C-STORE to transfer the images. It does not download the images directly, instead, it performs an action that makes the image archive send the study to a specific location that typically is its own workstation.

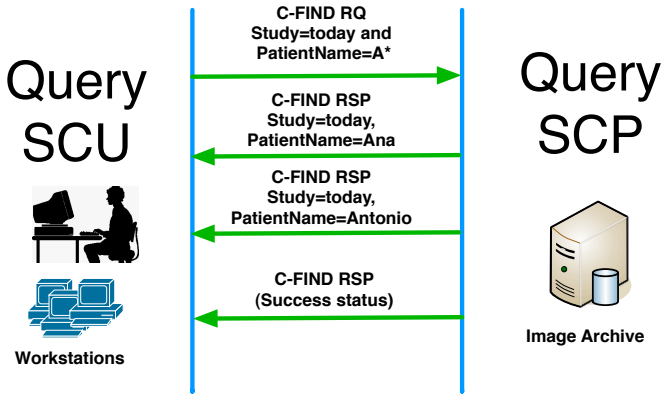

Fig 3.4: DICOM query (C-FIND). The Query SCU is the client that is searching the PACS repository (Query SCP).

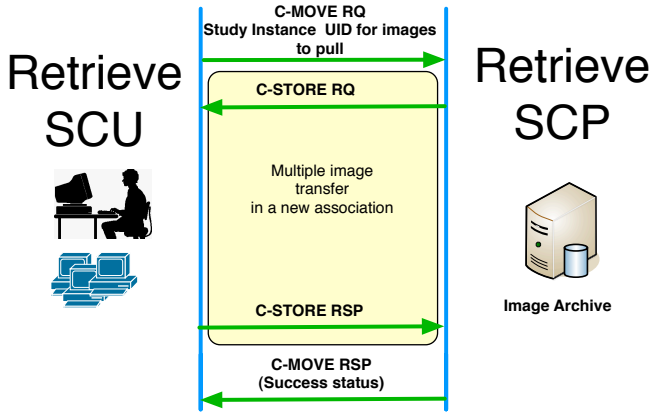

Fig 3.5: DICOM retrieve image. Retrieve SCU is the client that is fetching a set of images from PACS archive (Retrieve SCP).

3.1.3 Data integration cases

As previously mentioned, the technological challenges related to the sharing of medical imaging between multi-institutions are still not solved. In recent decades, several approaches have tried to tackle this issue (Figure 3.6). Many solutions have been developed, not all based on the standards. While DICOM standard has been widely adopted by the PACS vendor and several medical modalities equipment, other initiatives have been also exploring the area and creating guidelines to solve the problem, such as IHE (Integrating the Healthcare Enterprise) that started late in 2008 and have been developed over the years [104].

DICOM over email is an approach proposed in several papers between 2008 and 2009 [105, 106], as shown in Figure 3.6. Nevertheless, these solutions involve communication through email protocols, i.e. IMAP (Internet Message Access Protocol) or SMTP (Simple Mail Transfer Protocol), which may not be accessible in several institutions or networks, creating limitations in the use of these approaches. Moreover, these solutions must deal with mailbox message size limitations and still have some associated latency, due to the restrictions of email protocol.

Web PACS solutions have appeared in recent years [87, 107] and nowadays, many vendors offer private solutions to their customers. There is also WADO, Part 18 of the DICOM standard [108], which provides access to DICOM objects, using the HTTP protocol (see 3.1.2). However, it allows access only to the object level of the DICOM Information Model (DIM) hierarchy. Moreover, the biggest drawback associated with web technology is its limitations in terms of visualization, when compared to regular workstations for radiology diagnosis [35]. Nonetheless, the capability of having a web PACS and zero-footprint viewer allows easier access to the medical exams anytime and anywhere. This facilitates the clinicians' analysis and diagnosis, as they are able to use the medical images outside the hospital, without compromising the patients' privacy.

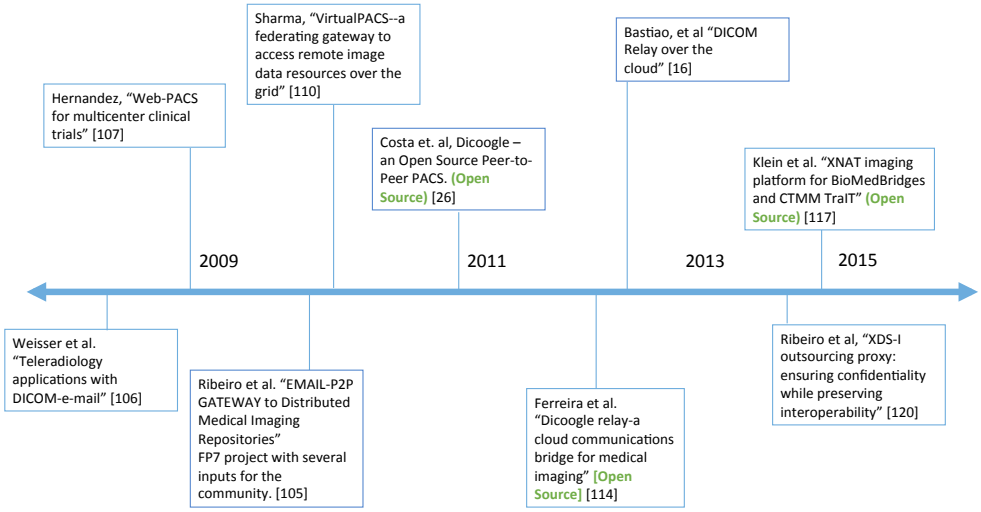

Figure 3.6: Most relevant scientific events in the medical imaging integration

GRID computing paradigm was also explored to provide federate access to distributed imaging repositories and a concept of Virtual PACS has been introduced [109-111]. These technologies were used mainly for large scale storage and processing of images, but also to allow the access between multiple sites [112]. Another approach, described in [113], presents a similar scenario based on Cloud computing. In this case, the solution focuses on exchanging, storing and sharing medical images across different hospitals. These approaches are relevant, but in both cases, a central repository is used and some institutions are not interested (or legally prohibited from doing so) in outsourcing the repository to a third entity. Costa et al. [26] proposed an approach to integrate several repositories based in P2P technology. Ferreira et al. have also been working in the integration of a medical imaging repository based on Cloud. However, both have as main target the exploration of DICOM metadata, essentially for research purposes [114]. Although being flexible to allow searching over any DICOM repository field and performing a fast query with a large amount of results following the philosophy of virtual PACS, they need to be the PACS archive or to have a replica of the data. XNAT [115] imaging platform is a web service for storing and organizing medical data mainly

developed to support sharing among researchers. For instance, BioMedBridges [116] is a project that aims to integrate several sources in biomedical informatics and is already using XNAT to support medical imaging share [117]. Nevertheless, it requires a setup of the software server infrastructure.

As already stated, a new initiative entitled Integrating Healthcare Enterprise (IHE) aims to improve the way healthcare institutions share information. IHE is a framework that takes advantage of well-accepted standards already implemented in most hospitals. It defined a profile named Cross-Enterprise Document Sharing for Imaging (XDS-I) [104], which intends to facilitate access to medical image repositories across multiple healthcare institutions. Those profiles make use of already accepted standards, such as DICOM and HL7 [118, 119]. Moreover, there are already studies and exploration of taking advantage of Cloud computing services do instance XDS infrastructure and consume it on-demand [120]. However, XDS-I is still a work in progress and is not implemented in most real scenarios. While this new trend is growing, institutions have already installed PACS infrastructure mainly supported on DICOM. In order to support XDS-I, many profiles have to be deployed in the institution, requiring great effort. A fully DICOM-compliant bridge solution to share standard services across healthcare institutions, without requiring complex changes in installed infrastructure, is still needed, at least for some small centres.

3.2 System proposal

3.2.1 Architecture

As explained earlier, the DICOM standard communications layer is not frequently used in inter-institutional interaction due to its own limitation. In practice, each hospital is an independent island, unable to establish standard communications with other hospital infrastructures. The integration of medical repositories from different institutions is an ad-hoc process, which has several barriers to surmount in deployment. Moreover, telework based on desktop diagnostic imaging software can be difficult, due to restrictions in accessing medical repositories outside the institution. The search over repositories of distinct organizational domains and the integrated visualization of results is extremely rare.

In this chapter, we propose the DICOM Cloud Routing that permits to create trustable share-point domains for medical imaging, where the institutions can easily provide and consume DICOM resources. It allows the establishment of inter-institutional medical imaging services, namely shared processes and integration of repositories. It is a solution supported on Cloud computing and provides a full stack of functionalities, including the registering of users and institutions, publishing of available DICOM services, automatic establishment of point-to-point DICOM services over HTTP protocol, control of quality-of-service, etc.

The solution architecture contains two main components: DICOM Bridge Router and DICOM Cloud Router, which we will explain in more detail further in this chapter (Figure 3.7).

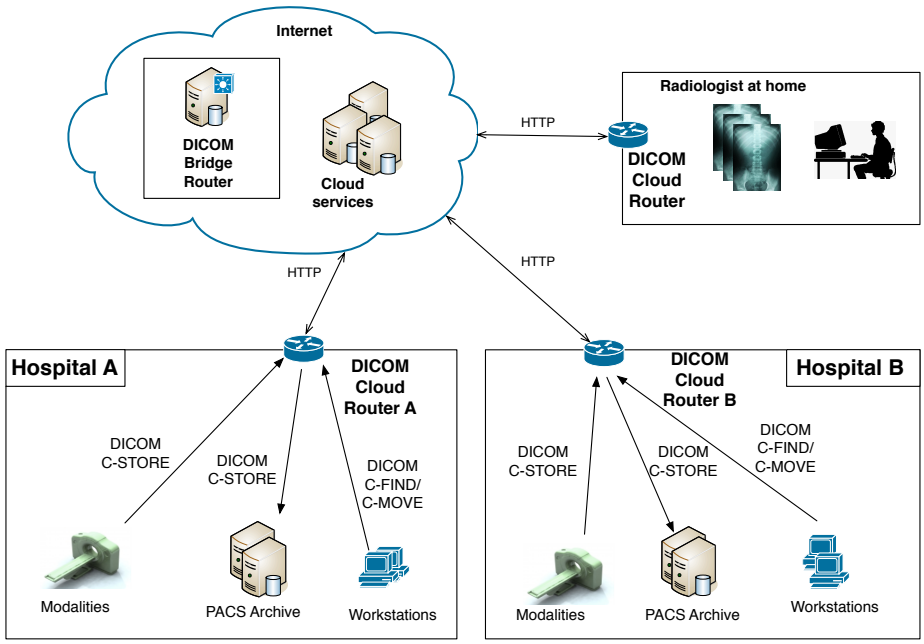

Figure 3.7: Architecture of the solution - Router in the boundary makes the communication between the DICOM devices and Cloud computing

Cloud offers a high quality of service, measured in uptime availability and long-term scalability. Our solution takes advantage of Cloud to provide remote search functionalities and exchange of medical imaging between different locations. However, communication between the components of digital medical laboratories is mainly carried out through DICOM. Nevertheless, Cloud resources are only accessible through web services. Thus, DICOM forwarding services between different sites is not straightforward. The main advantage of the proposed architecture is that it runs on top of Cloud services and follows the web 2.0 paradigm, i.e. services are available anywhere and anytime through HTTP connections.

DICOM protocol runs over TCP/IP protocol and contains its own addressing model through the AETitle that identifies the medical device [100]. Due to network filter restrictions (i.e. firewalls), this communication does not perform well in WAN (Wide Area Network) scenarios. Thus, it is not compatible with the Web 2.0 paradigms and some networks only support HTTP/HTTPS access, filtering the remaining protocols. To extend communication to different institutions, the proposed approach takes advantage of the DICOM addressing mechanism to route the information to the correct location (i.e. AETitle is the DICOM address mechanism).

3.2.2 Components

DICOM Cloud Router

The DICOM Cloud Router is a gateway that ensures interoperability between the Web platform (at Cloud) and the DICOM universe (devices at Intranet) and has the main responsibility of handling DICOM services and forwarding messages to the correct place. The forwarding of DICOM messages is based on the AETitle that identifies the medical device. Real world objects were mapped directly in the DICOM standard, for instance, DICOM equipment is represented as a "Device" in the defined concepts of the standard. The Router supports multiple devices (i.e. as many as are online in the WAN DICOM network), each one with a different AETitle and transfer syntaxes (i.e. data codification supported).

Furthermore, each medical institution or isolated DICOM network that wants to share services in the WAN DICOM network needs to run, at least, a Router inside the local network that will be seen as a standard DICOM node supporting several services (Figure 3.7). The communication between routers is carried out through the Bridge located on a Cloud provider. Routers communication does not require to open any additional firewall service. It only uses HTTPS communication from inside to outside, which is commonly available in institutions.

The routing mechanism also has a cache system integrated that aims to increase data availability and reduce the medical imaging communication's latency in distributed scenarios, for instance, an outsourced regional PACS archive. There are some problems related to the development of the cache mechanism to support medical imaging scenarios. The retrieval, caching and transfer of medical imaging studies must deal with huge amounts of data, but also with its heterogeneity. Different modalities produce data with distinct characteristics such as image matrix, number of frames and average image size [121]. Some modalities, such as CT, may produce thousands of image files per study up to 1GB. Other modalities, like cardiac US, can produce several cine-loop files with hundreds of Megabytes. Caching huge files is a complex issue, not only in terms of storage space, but also in data transfer latency. The strategy adopted to manage this problem, i.e. increasing the router's cache storage capacity and reducing latency, is based on splitting DICOM objects [121]. So, each file is logically divided into fragments of a pre-defined chunk size [122]. Moreover, the cache may not have all the study fragments and, so, it is possible to implement remote retrieval processes of only specific ones. Furthermore, if the information is available in more than one archive, the client router can retrieve blocks from multiple data sources (i.e. provider routers), increasing the system's performance in some network conditions [123].

Bridge Cloud Router

The Bridge is considered the main component of the architecture, since it manages all the processes and works as a relay mechanism between different DICOM Cloud Routers dispersed over several locations. It stores information about all the supported devices (i.e. routing tables - AETitles) and corresponding services. It has accounts from routers and a list of Cloud provider credentials that they can use to store temporary information, i.e. in-transit data. Moreover, it handles the session key used to cipher the DICOM messages of a point-to-point association.

System management is supported by a temporary information system and the platform is accessible through the web service mechanism (RESTful). It provides the credentials to validate authorized routers, AETitle of the DICOM networks and credentials to access the Cloud provider. Only validated users registered in this entity can access the DICOM WAN Network.

The platform communications are Web 2.0 compliant, the processes are supported by Web Services and the messages are encapsulated in the HTTPS protocol. This provides communications security and ensures the setup of DICOM services in restrictive private LAN environments, even without IT network expertise or administrator privileges. In this way, our system is able to run on most network configurations present in healthcare institutions.

The vast amount of information that flows in the WAN network is uploaded/downloaded to the Cloud providers. However, this component does not store medical data, it only has a cache mechanism to support the forwarding of ciphered data between routers.

The Bridge needs to be always available over the Internet, because routers need to write information in the Bridge to support communications. However, it can be deployed in other places like, for instance, a private server or Cloud detained by a medical institution or a public Cloud provider.

3.2.3 Services and workflow

Initialization process

The association of a Router to an affinity domain, i.e. a centralized share-point of medical imaging, and the publishing of its DICOM services (available in Intranet devices) follows several steps:

1. Validation of Router credentials;
2. Load the list of internal DICOM services to share with WAN;
3. Load available services from other private networks (accessible via WAN);

4. Start DICOM services;

5. Subscribe to the Cloud association channel and wait for new associations.

DICOM relay service only works effectively after Router account validation, to avoid unauthorized users accessing shared resources (step 1). Each working Router needs to be authenticated by the Bridge. The validation of the Router is performed with a username and a password. Once successfully authenticated, it will retrieve a session token that will be used to support network operations, i.e. forward DICOM messages. The token has an associated time-life and messages with expired tokens will be refused. Each Router needs to register (i.e. publish) the services to be shared with other institutions (step 2). This information relating to DICOM services available in the Intranet must be configured by the local PACS manager and stored in the routing table of the local Router. Next, this information is widely spread to other routers via the Bridge. Receiving reference to a new service X provided by a Router Y, forces all other Routers to launch this service X on the respective Intranet.

In the synchronization process, a Router provides its information to the shared DICOM domain through the Bridge, and receives the list of AETitles and transfer syntaxes from all other routers connected (step 3). This procedure unites the local routing tables with the external tables. Afterwards, the Router identifies the AETitles providing services (i.e. servers) and will start to provide all those services to the local network (step 4). The Router runs a DICOM device per AETitle and one single AETitle can support more than one DICOM service, e.g. Storage (i.e. upload of DICOM studies), C-FIND (i.e. search over DICOM repositories) or C-MOVE (i.e. download of DICOM studies). Thus, the Router can be contacted by the same IP-Port address and it will distinguish requests by the destination AETitle (step 5). Subscription is carried out through the notification systems that Cloud providers are offering nowadays. This service allows Routers waiting for new connections from remote institutions to be notified.

Storage service

The DICOM storage service (Figure 3.8) is responsible for moving persistent objects (image, structure reports, waveforms) between different medical devices and is based on the standard C-STORE command [124]. A DICOM association must be initiated, including the negotiation of transfer parameters, before sending any sort of service commands. Thus, for each client request received, the Router needs to create a new association in the Bridge (Figure 3.9). This has an associated session key that will be used to cipher the DICOM objects to be transferred through the public Cloud providers. This key is shared between the two Routers involved.

When a DICOM study is sent to another medical institution, external Cloud providers are used to store information in transit, reducing the Bridge overload (Figure 3.8). For each

C-STORE operation (step 1) they store the SOP Instance UID, i.e. identifier of the set of images, and the storage location of in-transit resources (step 2 and step 3), i.e. references to external Cloud location. Finally, the remote Router takes the images and sends them to the remote archive (step 5). An acknowledgment message is returned from the archive, signalizing the end of the transfer (step 6). In addition, this message indicates the association closing the Storage service. The control messages are sent over the notification systems, but they do not expose any medical data, only messages to control the dataflow.

Storage is the most complex process because it supports a multi-thread mechanism during the uploading of files to the Cloud providers. When a DICOM C-STORE is invoked, the remote Router opens an association with the target server, even without having all files in its safeguard. Meanwhile, those files are downloaded from the Cloud via a multi-thread process and using the splitting technique, in order to enhance the upload/download times [122]. For each downloaded file, the remote Router sends a notification to the other source Router that triggered the initial action, meaning that the file has already been transferred.

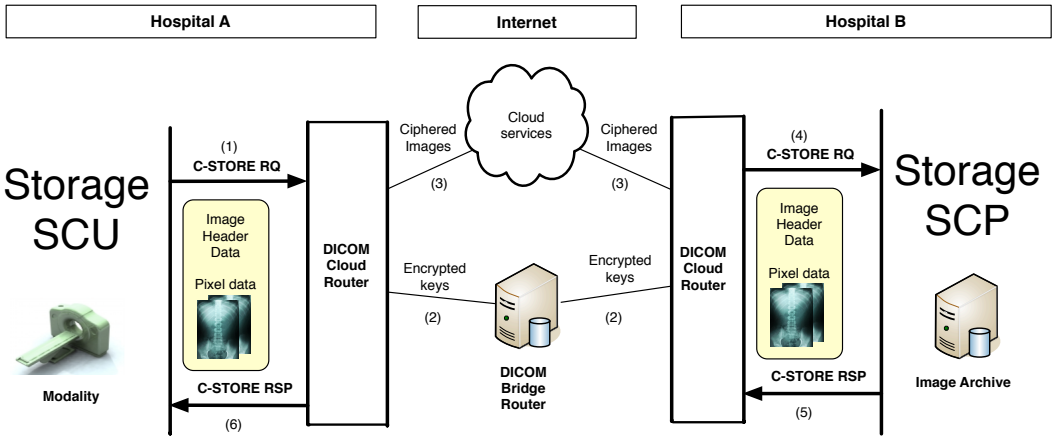

Figure 3.8: DICOM storage process. The Router forwards DICOM C-STORE objects via cloud and DICOM Bridge Router and the remote router receives the images and sends to the PACS archive

Remote searching

The C-FIND command allows the user to perform search operations over a PACS archive. C-FIND message has an IOD (Information Object Definition) that refers to several DIM (DICOM Information Model) fields normally used by clinical staff. The Router receives DICOM C-FIND requests (step 1) and translates them to a non-DICOM message (Figure 3.9). Meanwhile, the query is exported to XML, ciphered and uploaded to the Cloud blobstore (step 2 and 3). A session key is also generated to encrypt the association data, similarly to the storage process. On the other side, the remote Router receives the query message notification and gets the resources from the Cloud. The XML query-message is deciphered and the respective PACS archive is interrogated, using the routing table (step 4). The PACS archive will return a message with the responses

matching the query (step 5). The response is also ciphered and put in the Cloud blobstore. The Router initiating the process receives an asynchronous notification via Cloud and downloads the XML responses. After deciphering the answers, it will create the DIMSE C-FIND Responses and send them back to the workstation (step 6). Afterwards, the DICOM associations are closed and, meanwhile, the association on the remote side has also been closed.

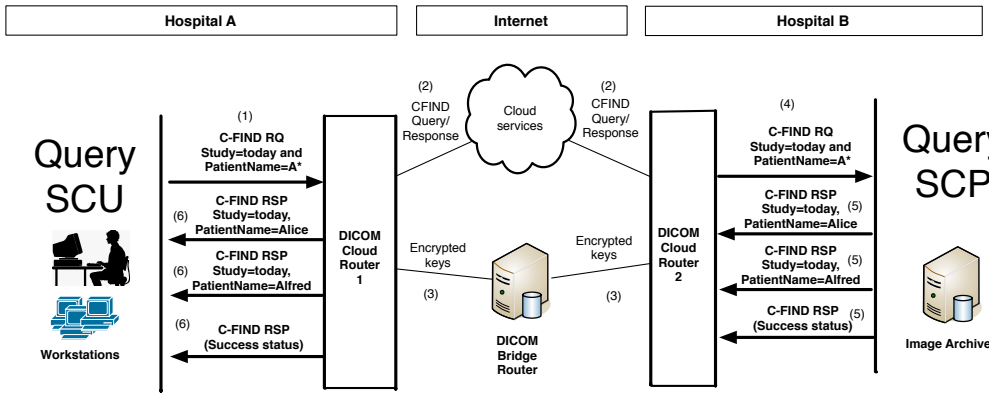

Figure 3.9: DICOM query process. The Router forwards DICOM C-FIND query via Cloud and DICOM Bridge Router. The remote router (Router 2) receives the query and inquires the PACS archive with the same query. The PACS archive sends the DICOM C-FIND responses to Router 2. Router 2 sends the responses back to Router 1 via Cloud. Router 1 answers the workstations with the C-FIND responses.

Moving studies

Looking at the DICOM C-MOVE, it is important to mention that this command uses the C-STORE to transfer DICOM objects from the server to the client (Figure 3.10). In the retrieval process, C-MOVE action is performed by the workstations. They send a request to move a patient study, a series, or more rarely an image, from a remote archive to a local computer.

The DICOM C-MOVE operation is asynchronous. For instance, device A wants to retrieve a study from device B (repository) and requests some object(s). Subsequently, device B invokes a storage operation from device A. Like in C-FIND, our C-MOVE implementation also creates an instance of DICOM association via Cloud. This instance also shares the session key to cipher/decipher the messages flowing over the Cloud. The workstation sends the C-MOVE request to the local Router (step 1), which will forward it to the remote destination (Figure 3.10). To do this, the first Router uploads the message to the Cloud and then, sends a notification to the remote Router, which will receive the message to move an image, series or study (step 2 and step 3). Thus, it will perform the DICOM C-MOVE command according to its routing table (step 4). On the other side, the archive server starts moving through the storage service to send images to the requester (common in the DICOM retrieval action). When the study is delivered to the requesting workstation, an acknowledgment is sent to the original router and retrieve C-MOVE response is delivered (step 5 and 6). Finally, the associations are closed.

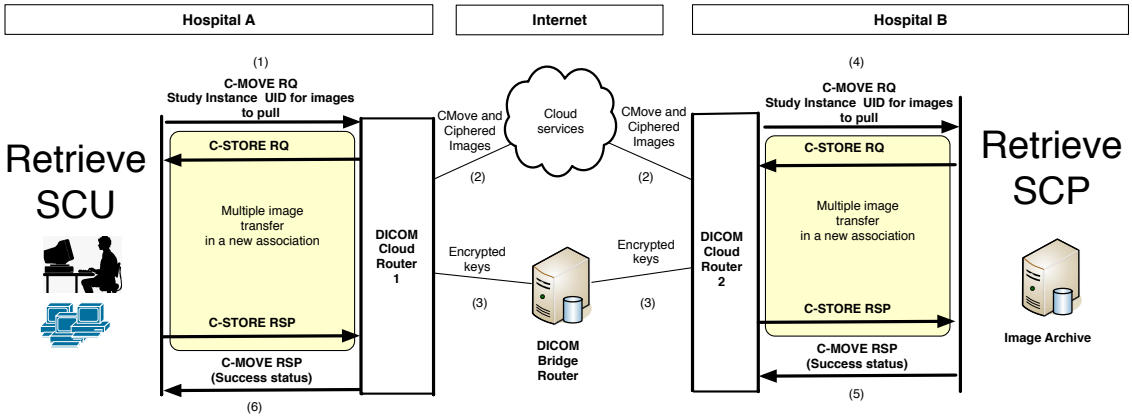

Figure 3.10: DICOM C-MOVE service. The workstation invokes the C-MOVE command that is sent to Router 1. The message is forwarded to Router 2 via Cloud. On the other side, Router 2 performs the command to the PACS archive. This action will trigger a C-STORE action executed in the PACS archive. Finally, the C-MOVE response is sent from Router 2 to Router 1 which will finish the operation.

3.2.4 Cloud infrastructure – abstraction layer

The public Cloud infrastructure is used for supporting the proposed Bridge Cloud Router. Nonetheless, several Cloud services provide different API (Application Programming Interface), and they are not compatible with each other. Thus, the initial assumption depends on a particular provider. In order to tackle this issue, we developed an abstraction layer that allows us to deploy the proposed solution in other Cloud service providers, avoiding being locked to any specific seller or deal with availability problems. It provides a common API for delivering services over multi-vendor Cloud resources**,** entitled Service Delivery Cloud Platform (SDCP) [125]. This middleware supports two Cloud services: Cloud blobstore (storage - associative memories) and notification systems (asynchronous message-based communication); with a plugin-based mechanism that provides a transparent interface to the Cloud providers.

SDCP platform has three main goals: 1) grant interoperability between different Cloud providers, creating an abstract layer for three Cloud services; 2) deliver services using multiple Cloud resources, including storage, database management and notification systems. 3) provide service combination, decoration and orchestration. The first goal (1) consists of granting interoperability between Cloud players in a transparent way. Basically, an application can work with as many vendors as desired, taking advantage of existing Cloud providers. The SDCP allows creating a Cloud provider poll. For instance, it can store data in multiple Cloud vendors or Cloud free services, creating a federate view of all containers. In addition, it enables the developer to have interoperability with other protocols (2) inside private networks. Cloud services of distinct providers can bundle and decorate it with extra functionalities like, for instance, data ciphering on-the-fly (3). Moreover, cache and pre-fetching mechanisms are other examples of value-added SDCP services, extremely important to reduce latency.

System architecture consists of a hybrid infrastructure that allows "Enterprise to the Cloud" and "Enterprise to the Cloud to Enterprise" applications, i.e. communication between two, or more, different enterprises, using multiple resources from different Cloud vendors [126]. The architecture has, basically, two main components: the Cloud Controller and the Cloud Gateway (Figure 3.11). The Controller contains sensitive information and must, therefore, be deployed in a trustable provider. Within the SDCP architecture we have also built a SDK (Software Development Toolkit) that simplifies the development of SDCP-based applications [127].

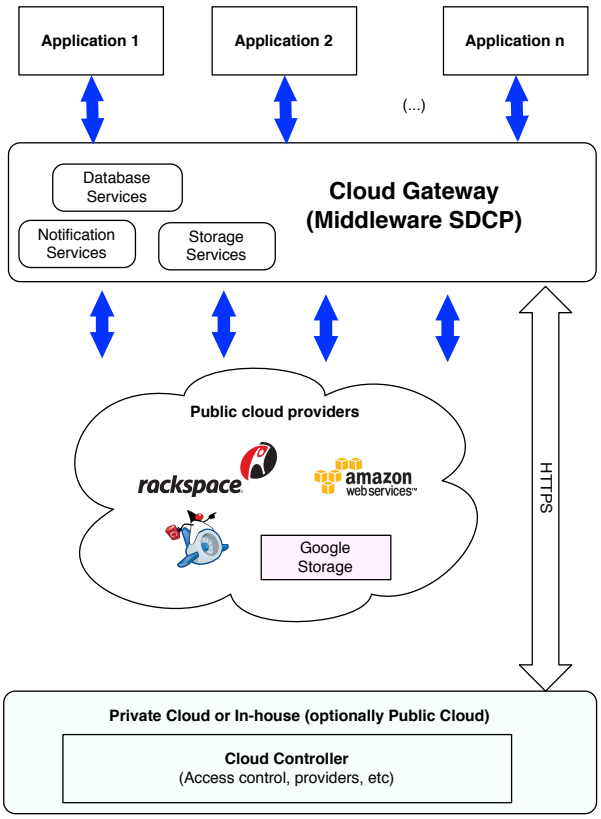

Figure 3.11: SDCP general overview

The SDCP was designed to simplify the development and loading of new application modules, using a plugin approach or web service API. As it is possible to see in Figure 3.11, the applications are on top of SDCP. The platform is able to deliver new services using the Cloud facilities: data store, databases and communication using Cloud providers. In order to extend the platform to support different providers and services (e.g. Google, Amazon, Azure, Rakespace, etc), we have built a specific model whose structural design sustains the use of different modules under the same interface. We have considered a distributed architecture to support multiple accesses to these data, from distinct points.

A more detailed description of SDCP architecture, components and dataflow can be found here [125] [128].

3.3 Assessment

3.3.1 Performance

In order to assess the performance of the proposed system, a testbed was performed with two different, geographically separate, networks. Each location has bandwidth available around 24 Mbps. We accomplished the tests with an Intel Core 2 Duo 2.2GHz with 2GB RAM and an AMD Athlon 1.6 GHz with 2GB of RAM. These two machines contain the DICOM Cloud Router and are located in two different locations. There was another machine containing the PACS Server archive, i.e. in the remote location. We used several third party client workstations to test the Cloud relay service, namely OsiriX [129] [130], dcm4che2, dcmtk [131] and Conquest [132].

In this section, we will perform the trials supported on Amazon S3 Cloud provider. The values with DICOM direct TCP/IP connections between the two sites were also analysed and compared with the Cloud relay solution. Our Bridge was deployed in Google AppEngine and the notification provider was PubNub [133]. A dataset with 4 studies was used, containing 851 DICOM files.

We analyzed the moving of medical imaging studies from one institution to another, which we considered a key point in regional communications. We compared the values with DICOM direct connection over a VPN (Table 3.1). In addition, the direct connection is frequently blocked or needs to be set up in the network and this can be a barrier to accessing the repository, unlike our transparent Web 2.0 approaches.

In the first approach, [16], no optimization has been made or cache used. The time obtained seems to be acceptable and the proposed Routing process may be optimized. The solution with splitting files into optimized chunks and study thread parallelization changed the behavior [121]. The average time measurements of the storage are presented in Table 3.1. In the results, no cache has been used, so that we have a fair metric with direct connection. The proposed solution considerably speeds up study retrieval times, on average 1.55 times, with the best case study up to 2.4 times (MR-3 study). DICOM protocol in WAN introduces severe performance penalties.

Table 3.1: Remote studies transfer - time measurements

			Retrieval Times (s)	
Modality	Number of Files	Volume (MB)	VPN Connection	DICOM Router Platform
PT-1	244	16.3	14.3	13.8
MR-2	223	47.1	31.1	19.1
MR-3	369	206.1	115.5	48.1
XA-4	15	401.6	228.7	202.8

This test was performed using the Dicoogle PACS [26] to execute remote queries over the network. Dicoogle is able to search, simultaneously, over distinct repositories and integrate the results in a unique view with a reference to the studies' remote location. In the trials, we search the remote repository using a considered amount of results, intending to illustrate the typical workflow of a workstation. In this case, the query and search in the remote environments are slower than direct connections, but still accepted, taking into considerations that shows 243 results in 1.92 seconds compared with 1.1 second in direct connection. However, the direct connection does not supply any security mechanism and is very difficult to implement in a real scenario.

3.3.2 Integration case study

The presented solution was deployed in a geo-distributed PACS, for physically federated medical imaging studies of two distinct health units, belonging to the same administrative domain. DICOM Cloud Routing was used to deploy a Regional PACS archive hosted in a private Cloud, including transparent communication processes with several distributed modalities and workstations. With the distributed PACS in place, both sites started to use and store all DICOM objects produced in a single common archive. The central archive relies on Dicoogle [134], an open source PACS archive compliant with DICOM standard.

The archive was placed in a machine with 8 Intel cores i7-2700K with 3.50GHz and 16GB of RAM. This machine contains a 6 TB of storage with RAID 1 and one disk with SSD to support reading and writing at high speed. At the other site, the modalities store the exams on the site's Cloud Router. The Cloud Router has a short-term cache that stores the most recent studies to serve the site more efficiently. However, when the study is stored in the short-term cache, the router forwards it in parallel to the central archive. The Cloud Router was placed in a machine with 4 Atom cores with 1.8GHz and 2GB of RAM and a hard-drive with 500GB. Both machines archive and Cloud Router are supported over Linux with Ubuntu Server 12.04 LTS.

The healthcare institutions, in this scenario, provide three acquisition services: Computer Radiology (CR), Ultrasound (US) and Computer Tomography (CT). The management system was assessed during 3 months, between November 2012 and January 2013, corresponding to a median of 5000 studies per month. In each healthcare unit there are 1 computer topographies, 1 mammography, 2 digital computer radiology scanners and 6 echocardiographs. During all process 4 medical doctors, 6 people from clinical staff, 2 people from IT and 2 supporting technical staff are present. Both sites have the same modalities. In order to reduce the number of hardware resources and their associated costs, it was decided to only store medical images in one datacenter.

3.4 Final considerations

The presented solution promotes the establishment of DICOM standard services in distributed environments, involving distinct medical imaging devices. The proposed architecture supports the creation of federated DICOM networks, establishing a unique view of all resources that can be searched and retrieved. The results proved that it is a good solution to support the sharing and remote access to medical imaging studies in distributed environment, promoting the access to data "anytime, anywhere" through Web 2.0 compliant services.

The solution validation was done in two perspectives: functional and performance. On the one hand, it was presented a real world use case that validates the platform usage to integrate medical imaging studies from distinct sources. On the other hand, an experimental testbed has evaluated the solution and the results show a good performance, when compared to a direct connection through VPN channel.

The stability of the system regarding the workload within the DICOM Cloud Router might have peaks when most image diagnostics are performed (i.e. late morning to early afternoon). However, the Router component only forwards messages, a light operation when compared with a "real" PACS archive SCP that needs to store images on disk, update a database and sometimes compress the data.

The presented solution does not require complex setups. Regarding the end-user, i.e. radiologist at home, or medical institutions that want to share their PACS servers, they need to install the DICOM Cloud Router software that does not require any complexity to setup. It is only needed to add the DICOM services, i.e. the AETitle, IP and listening port for service. Regarding the Bridge, which is the most complex component, just needs to be deployed once in an application server. Then, it supplies a web interface to setup the Cloud providers API keys and also register the login to the routers. Only validated routers over the Bridge will be able to exchange DICOM services.

The exchange of medical data across multiple institutions can have other problems like the existence of different patient identifiers. These problems are common to other shared environments and their analysis is out of the scope of this proposal. Moreover, there are standards already taking care of that, for instance, PIX (Patient Identifier Cross Referencing). Our solution can implement this, or another conversion mechanism, as a Bridge component.

The integration of medical imaging scenarios created many challenges despite of what has been proposed in this chapter. For instance, to ensure quality of service in real use cases, an architecture to monitor platforms for DICOM services [17] was created. The proposed platform can remotely monitor DICOM nodes in a geo-distributed way, like for instance, Cloud PACS solutions, as the one proposed in this chapter.

The quality of the integrated medical imaging environments also creates new challenges when extracting knowledge from repositories between several institutions. Nevertheless, there was a need to acquire more information, coming from other sources, and have a flexible and easy way to inquire those information systems. The next chapter will focus on the integration of the medical images databases and describe how the information can be linked.

4 Sensor-based architecture for integration of electronic health records[1]

"A good plan today is better than a perfect plan tomorrow"

George Patton

The use of healthcare information systems has increased in many countries, leading to productivity gains and improvements in the quality of services [135]. Several types of computer information systems are currently used within these institutions such as HIS, RIS and PACS. In medical imaging laboratories, every day, a huge amount of imaging data and examination reports is stored in dedicated repositories. The use of those repositories is mainly oriented to support patient-related processes. However, if adequately used, they also allow extraction of relevant metrics to evaluate department workflows. By correlating medical imaging information provided by different sources – e.g. the archive, the network DICOM traffic and the RIS database - it is possible to obtain efficiency metrics [4], monitor processes and provide alerts based on DICOM tags and

[1] This chapter is mainly based in the publication *Sensor-based architecture for medical imaging workflow analysis, Journal of Medical Systems, Springer 2014. [16].*

workflows [136], ensure processes' compliance with internal rules, and use data mining techniques to support knowledge extraction [86, 137]. These metrics are important for service planning and optimization, at a time when the rationalization of resources in healthcare is of the utmost importance in many countries [138-140].

Despite all the advantages that new information systems have brought to medical institutions, the proliferation of distinct data sources often results in isolated islands of information [141]. Moreover, data collection and information production are distinct processes. For instance, PACS archives generally do not index all the information present in a DICOM file meta-data, because this protocol only supports queries based on textual template matching over a limited number of fields, such as patient name or UID [142].

This chapter proposes a solution to correlate and integrate several medical imaging information sources through distributed intelligent sensors, which are controlled by a web platform integrator. It allows users to search and analyse the information available in disperse sources, namely in DICOM-based archives, in the medical imaging workflow extracted from traffic flows, and in the RIS examination reports. To assess the solution we created a case study based on an echocardiogram repository with 84063 examinations. Besides this, the developed platform has great potential for use in many distinct scenarios such as performance metrics extraction, optimization of resource usage and extraction of relevant datasets for clinical research.

4.1 Background

In spite of the great evolution on data integration standards, the integration of health information systems is still an unsolved issue, due to many technical, ethical and political issues. The task of transmitting and linking data across multiple biomedical data sources is complex and difficult considering the different data formats, distinct institutions and non-aligned policies [84].

While there are already hospitals starting to implement Data Warehouses based on solutions such as Pentaho [1] and Open Talend [2], they still rely mainly on the integration of single hospitals. Moreover, they also face some difficulties integrating data sources. The developed solutions are part of the information systems that will feed the Data Warehouses (Figure 4.1). In this section, we will discuss some standards that have been proposed to simplify the exchange of medical records between several institutions and an analysis of the related work will also be made.

[1] http://www.pentaho.com/
[2] https://www.talend.com

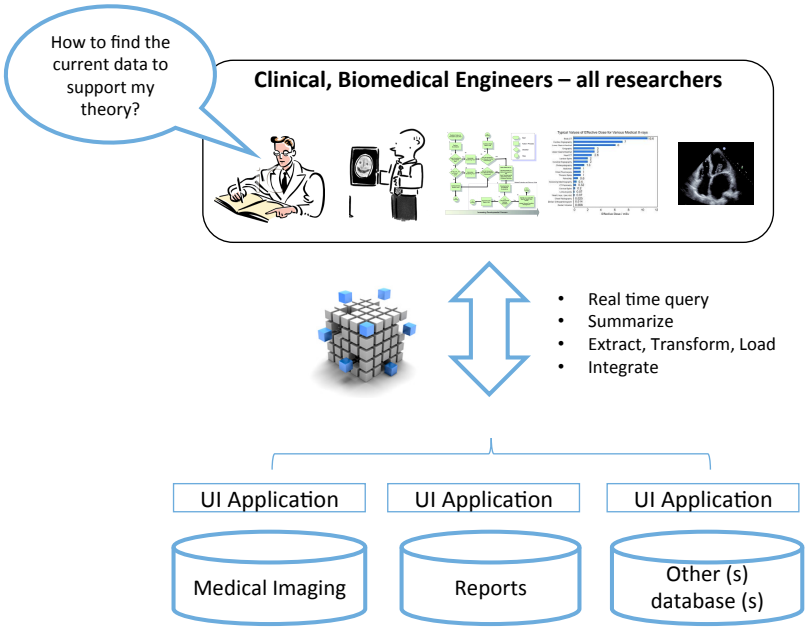

Figure 4.1: Data integration: the answer to solve the gap between the databases and the researchers

4.1.1 Standards for medical data interoperability

Several standards have been proposed to allow the access, sharing, and storage of different biomedical information sources.

The most used standard for exchanging electronic healthcare information is the HL7. It follows an event-driven approach, also known as triggered, i.e., for each event, the exchange of messages between two applications is triggered. There are several types of triggers defined in the standard, for instance, admitting a new patient, entering a new order and inserting new clinical results.

The HL7 message protocol [143] plays an extremely important role in the healthcare services, offering a mechanism to exchange information between medical applications. Nonetheless, this standard does not offer fully interoperability, because it only defines the messages structure, and does not specify the kind of information contained in transacted messages. The most recent HL7 standard is the version 3, which is not compatible with version 2 and its use is not so widespread [144]. The Reference Information Model (RIM) is the base of information model specified in HL7, and it is related with the exchange of data, instead of its storage. The main objective associated to the RIM creation is to support the interoperability between the systems used in different healthcare institutions. It follows an object-oriented model based on a generic template. The HL7 version 3 also includes the Clinical Document Architecture (CDA) [143], a standard to organize the digital documents produced in the healthcare services. The CDA extends the RIM and uses the same type set of HL7 v3.

OpenEHR is another open standard, which main goal is to create an independent architecture that permits a technical and functional expansion, without expensive costs and efforts. OpenEHR model gives special importance to ontologies aspects. Its approach is based on different levels of information following a real world nomenclatures, i.e. described by the classification and appropriated terms. As long as other standards focus on one specific area, such as DICOM on digital imaging or HL7 on patient management, the OpenEHR standard can be used to describe any kind of data, since any information structure can be defined with the help of an archetype. An archetype represents a clinical concept. It is used to constrain instances of the OpenEHR information model by defining a valid structure, data types and values. This standard allows using international standards such as SNOMED, ICDx and LOINC and is also compliant with HL7 messages. OpenEHR is available in open source code.

A new initiative entitled Integrating Healthcare Enterprise (IHE) appeared in the recent years, also aiming to improve the way healthcare institutions share information. IHE is a framework that takes advantage of well-accepted standards, already used in most hospitals. Instead of defining new standards, IHE promotes the use of the existing ones (e.g. DICOM, HL7, ISO, IETF, etc). It works as a document to normalize, discuss and undertake problems of information integration between several health information systems. It intends to grant interoperability providing integration profiles - guidelines to describe real-world scenarios or specific characteristics for building integration ready systems. The IHE profiles provide a precise definition of how standards can be implemented to meet specific clinical needs. Each profile is composed by several actors and transactions, which are based on real world healthcare environment, i.e. the health information systems. While some of the transactions are traditionally performed by specific product categories (e.g. HIS, Electronic Patient Record, RIS, PACS, Clinical Information Systems or imaging modalities), IHE intentionally avoids associating functions or actors with such categories. For each actor, the IHE defines only the functions associated with integrating information systems. The IHE definition of actors and transactions provides the basis for defining the interactions among functional components of the healthcare information system environment [63]. For instance, a profile named Cross-Enterprise Document Sharing for Imaging (XDS-I) was defined [104], which intends to facilitate access to, and distribution of, medical image repositories across multiple healthcare institutions. Those profiles make use of already accepted standards, such as DICOM or HL7 [118, 119].

4.1.2 Related work

Many efforts have been made to create Electronic Health Records solutions in medical institutions. Moreover, there have been also several efforts to provide integrated views of EHR [145-148]. Nevertheless, one of the common problems is how to search for specific

fields that are not recorded in EHR, as detailed in section 2.6.1. This section describes the several methodologies that allow extracting knowledge from the medical imaging records, proposed by numerous authors in the literature. Moreover, their flexibility and extensibility to gather and integrate data from other sources will also be included criteria.

The integration of medical records permits extracting information that is not usually available in the current information systems, or if so, that is very hard to extract and analyze in large scale. For instance, in the most part of medical archives, PACS, the DICOM metadata are usually recorded in a SQL table, but it is not easy to search on images features. Moreover, due to the wide variety of IOD in DICOM, it is very complex to create a relational schema to store all fields [85]. While some followed static schemas approaches, for instance data warehouses based on open source software, they are limited when we intend to explore the metadata of different medical imaging archives, such as measure the quality of information, extracting dose analyses or productive metric [65, 149, 150]. Figure 4.2 provides an overview of the most important trends regarding data integration since 2008, mainly focused on the data gathering in medical imaging laboratories and their related departments.

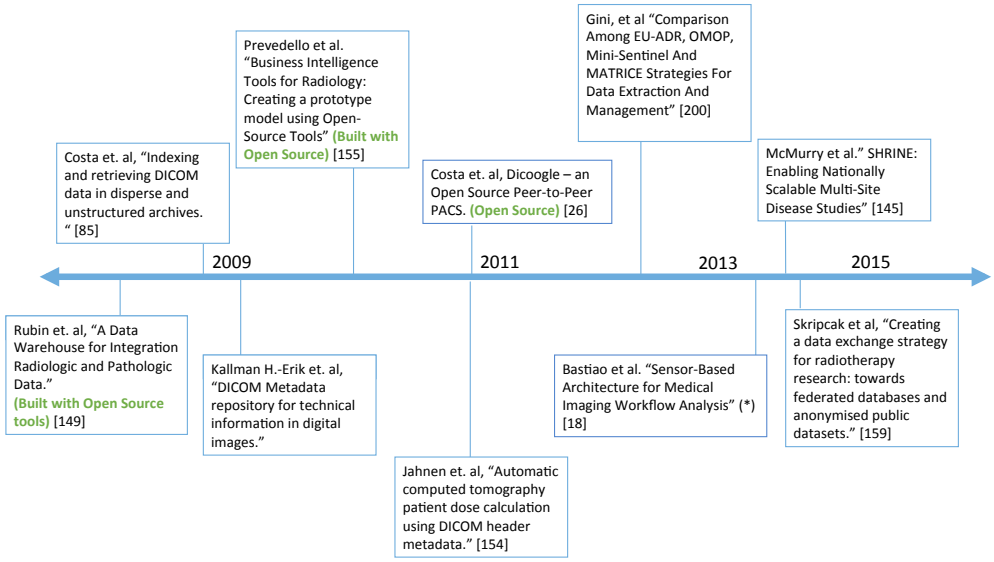

Figure 4.2: Evolution of the data gathering in medical imaging (images, reports and related information). (*) – this chapter is based in this article.

Kallman et al. [151] developed a DICOM metadata repository for the optimization of the dose radiation in computer tomography (CT) allowing to detect malfunctioning devices. They developed a daily routine that accesses the PACS Archive server and extracts the metadata. This process is running since 2004 and has collected around 18 million images metadata, until the date of publication. Their limitation lies in the database, which only contains approximately 200 DICOM static fields. They also establish direct communication between the medical images and the RIS database. Nevertheless, the system was mainly developed to support dosimetry information and analyze their current

problems, for a single institution. In 2008, the adoption of a data warehouse for data integration in radiology has gained importance. Rubin et al. [149] developed a data warehouse for integrating radiologic and pathologic data into a centralized location. They built the data warehouse on top of open source solutions and they explain the difficulties regarding the regulatory approval and technical issues, mainly regarding systems' connecting problems. Moreover, following the authors, these integration tasks usually requires a huge effort to build the database schema that fulfills both areas, due to the access to the standard services (which usually do not follow the correct implementation), terminology, languages and used codifications. In terms of the Radiology Reports, the department system described by the authors used the HL7 standard. However, this was not enough to make them easily accessible and only a small subset of particular fields has been gathered, without the possibility to create an automatic and vendor neutral system to extract such information.

While more data warehouse based approaches have continuously been proposed [150, 152, 153], the problem with DICOM heterogeneity was not yet solved. Other authors kept exploring different approaches to gather more information from the medical imaging repositories. In 2011, Jahnen et al. [154] propose a system to acquire data, based on the C-STORE service. They want to keep the CT dose information in a duplicate information system and their engine was supported by a relational database. The constructed repository needs, then, to be processed by an external data-analysis tool such as Microsoft Excel.. In order to create a better analysis tool and optimize healthcare quality, a tool to measure performance and quality was created, named Performance And Monitoring Server For Medical Data (PerMoS). Thus, the analysis are executed in an integrated way, allowing the medical staff and clinicians to extract data directly from the developed application, mainly related to the research application.

On the same line, Prevedello et al. [155] developed a data warehouse target to obtain key performance indicators (KPI) and enhance the productivity. A multidimensional database format has been developed (data cube) organized by modality, time, number of exams per patients and several others attributes. The results are exportable to a tabular format allowing statistical analysis through other packages. Nonetheless, the data warehouse in this case was not used to integrate more information sources, but instead to pre-calculate several data to be accessed more quickly.

An extensible and open source tool to explore medical imaging metadata is Dicoogle [26]. A large amount of space is required in medical institutions to store imaging metadata. While the information is still there, it is mainly used for clinical practice and to analyse a particular patient, most part of the times, only one time. Dicoogle allows accessing to a large medical image repository and performing very flexible queries and

extracting the information for population studies. This service has a huge unexplored potential, considering what has been firstly presented in 2011 [26].

In 2010, the Dicoogle architecture had high coupling with the service that they supply, such as indexing medical imaging files and share files across the network [26]. The extensibility of the solution was required in order to develop new knowledge extraction tools, without penalizing the main architecture. Following the micro-kernel architecture, several plugins have been developed to improve the metadata extraction, mainly in terms of performance. The requirements raised while collecting information in real-environment were also considered [18, 156]. Nevertheless, the software is mainly focused in gather information from only PACS archive and is not developed to collect data from other sources.

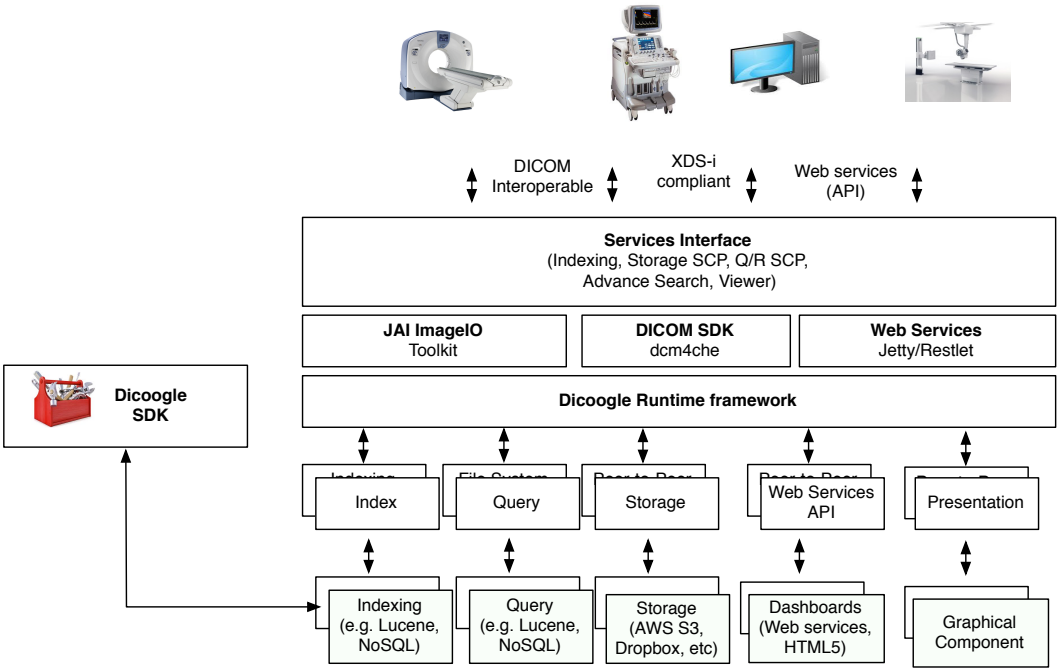

Figure 4.3: Dicoogle blueprint

Despite of a wide range of methodologies, developed applications, and strategies to build integrated data warehouse, the effort to build an extensible solution to gather information to the medical staff and clinical researchers is still complex and most of the proposed approaches are not portable. The information is not often available for medical researchers in an easy way, and much collaboration is required to perform a given study over a repository [157]. Moreover, the queries that they might perform to a single system cannot be distributed across different information systems, due to the lack of standardization and communication. However, there are already some standards to promote the interexchange of quality and clinical research documents such as IHE named Quality Research and Public Health (QRPH) [158], although still being a work in progress, not used in real environment. Achieving a real-time query across multi information sources is still a hard challenge, and the use of integration modality specific

sources is still common in hospitals and clinical research centers with complex infrastructures [159].

Extracting and correlating information in medical imaging environments is still a challenge, mostly due to the diversity and complexity of existing systems. Firstly, a methodology to extract the data from these source systems is needed. The diversity of data formats and the lack of standardization in some sources is problematic, e.g. the echocardiogram reports (in our case study, which will be described in the following sections). Secondly, the integration platform must be able to work with very large repositories, which makes scalability and flexibility to query the overall system main requirements. Finally, patient data privacy is a crucial issue and any solution must provide, for instance, data anonymization services. Although this problem has been covered in other projects [160-162], in the current proposal we are dealing with multiple data sources, where the same patient must be mapped into the same anonymous registry. What follows in the sections in sequel will present our approaches to solve those issues.

4.2 Sensor-based integration approach

4.2.1 Architecture

To integrate disperse data from different information systems, several approaches can be used such as a centralized database to collect all the data, or a federation engine, where a middleware provides uniform access to remote heterogeneous data sources. The centralized solution duplicates data and overloads the network, especially when acquiring large repositories, as is the case of medical imaging repositories. The federated solution also has some disadvantages. For instance, if the network connection is unstable, the search and retrieval of data will not work properly. On the other hand, the federated architecture does not overload the network, as in the centralized model, and it can supply a service autonomously, following the SOA (Service Oriented Architecture) guidelines. Our framework proposal, named Medical Workflow Imaging Analyser, follows a hybrid integration architecture, where only part of the data is collected by the central system, while the other part is performed in other data collectors.

The framework is a distributed system, formed of several components deployed within a hospital department (Figure 4.4). The core component of this system is a web application running on a private and secure server. Using the appropriate credentials, any user within the medical institution can access the system through a web browser. Besides the core component, there are several specialized sensors to allow the system to connect to each information source. These sensors provide an interface for the core component based on REST web services and each has unique credentials to authenticate each invocation. The system uses HTTPS to ensure confidentiality in communications between all components.

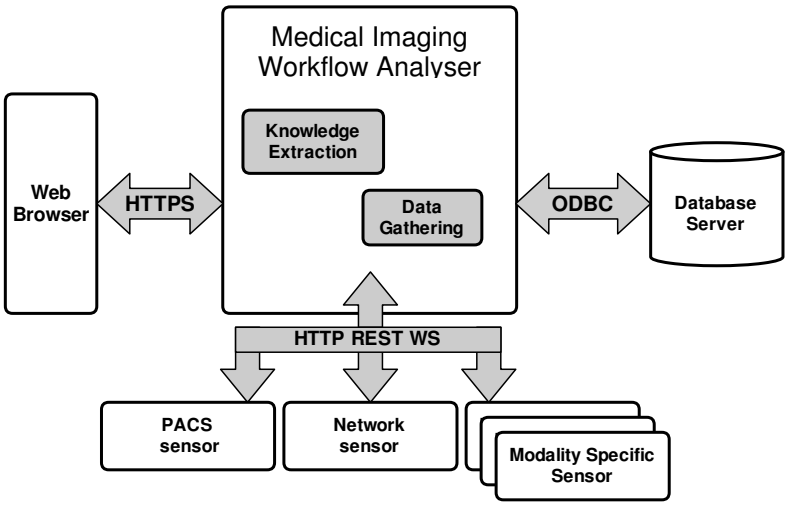

Figure 4.4. Top-level architecture and communication mechanisms.

4.2.2 Services API

The Service-Oriented Architecture followed makes the system more flexible to meet the needs of specific scenarios. We can, simultaneously, instantiate only some types of sensors and multiple replicas of each one. The web application orchestrates multiple services exposed by the sensors and delivers them to the user as a single aggregate service. In the context of this thesis, three sensors were developed: DICOM Network, PACS Archive and a modality oriented HIS.

The sensor should provide a RESTful API able to manipulate XML or JSON input or output parameters depending on the HTTP request headers. Four service modules were defined and can be implemented in each sensor: management, location service, search and pre-processing.

The search service is very important for the orchestration and it is used not only to find information in each sensor, but also to connect multiple data sources. It supports two types of queries: free text and advanced mode. To improve service performance, the search response only includes the patient id, name, exam type, report id and access number. However, the user can access all information by exporting the report fields using a specific API. There are several use cases where this API and filtering mechanism are used, such as knowing the geographical origin of exams in order to perform a demographic study. The geographical location service followed the standard ISO-3166 [163].

4.2.3 Privacy and security

Medical data is an extremely delicate subject and it is often necessary to ensure that patients cannot be identified. The developed framework will be used in different

scenarios and not only for integration of *m* type of databases, but also for *m* healthcare units. Thus, we assured that the privacy of data was taken into account.

The information collected by sensors can contain patient-identifying data. Fields such as patient's name, access number and demographic information are de-identified. Nevertheless, the anonymization process should allow clinical records of the same patient to be related, even if they were collected through distinct data sources. The adopted strategy always replaces the real name or patient ID with the same anonymous name or ID, preserving, in this way, the relation between the medical records of the same patient. The anonymization process is based on a mapping table that hides word patterns. This matching table is ciphered in a database. After concluding the data collection process, it is possible to discard the matching table and, thus, the connection with the real patients is lost.

This strategy works well for a single sensor, but we have multiple sensors collecting information from different data sources with anonymized data, and we intended to keep the relation with the origin of the patient information. Therefore, a centralized mechanism was developed, based on a master sensor that will anonymize the data. The remaining sensors will be slaves that will use the anonymization service available on the Master node. Thus, only one mapping table is held by the master sensor. This table will be kept safeguarded inside a hospital that can share the information through HTTPS access.

4.3 Data collectors

The architecture defines the interface to orchestrate the API. In this section, three developed sensors will be presented. Their main target is to collect information from medical departments, not only from images, but also from medical reports databases, or any other source that could be related with electronic health record.

4.3.1 PACS sensor

PACS sensor needs to handle great quantities of data produced by digital modalities. This information is stored in DICOM objects metadata, as already mentioned in section 4.1.2. Although this information is being stored in the healthcare repositories, traditional information systems are not able to access them in a large scale. These data could be useful in many scenarios, for example, in monitoring patient dosimetry, radiographic procedures and image quality. Some important parameters such as image processing parameters, exposure index, patient dose and geometric information, are generated by the modality and transferred to the PACS repository. Typically, PACS archives supply an interface to consult exams, but to access only the metadata we need to retrieve all the studies, which easily represents gigabytes or terabytes of data. To support the knowledge

extraction process Dicoogle was used as developing framework [26, 85, 86] and has been extended.

Blueprint extension

While Dicoogle was already proved to be an extensible platform to index and share DICOM metadata over the network, its extensibility was not fully developed, mainly due to the lack of SDK (Software Development Framework) support to extend these three dimensions: index, query and store. Moreover, if the Dicoogle web service API needs to be extended, there was no way. The flexibility to implement different strategies of index, search, storage and new services API was a requirement to build a polished PACS sensor based on Dicoogle (Figure 4.5). Many contributions have been given to the user requirements, software architecture design, and implementation, with results available not only in scientific community [156, 164], but also in Dicoogle github [1].

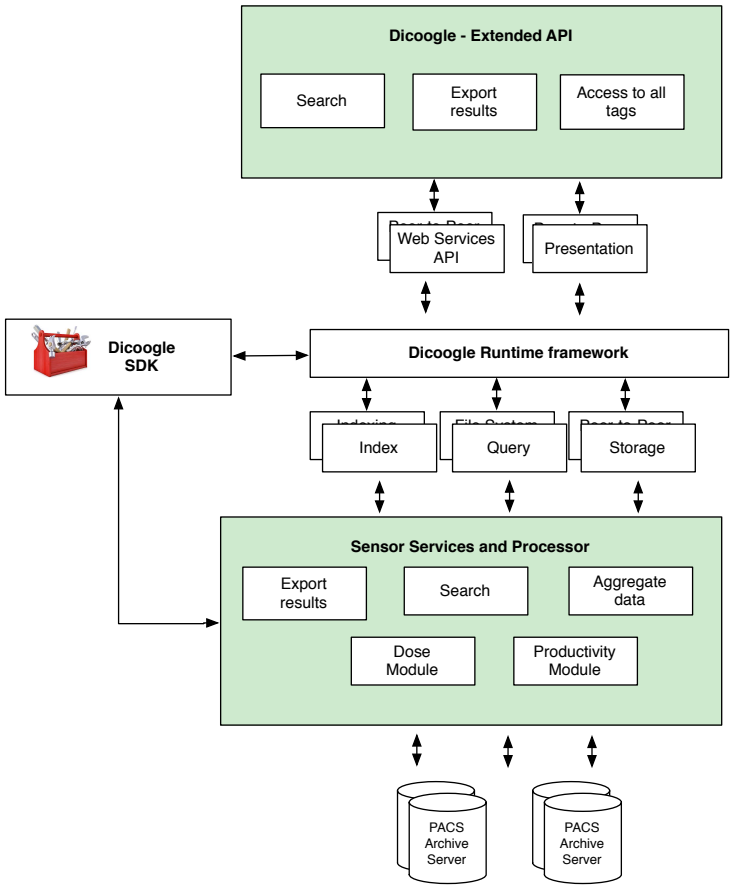

Figure 4.5: Access to the PACS archive repositories in healthcare units

On top of this new architecture design, several index optimizations have been implemented and new approaches followed to have more flexibility in the indexing system, namely due to proprietary formats, different PACS vendors and the capability to

[1] https://github.com/bioinformatics-ua/dicoogle

store the DICOM metadata in multiple data structures [156, 164]. The initial indexing process was an extremely intensive task that could take days, or weeks, to finish, depending on the repository number of DICOM objects. Thus, Dicoogle has been optimized to adjust this process intensity and not perturb regular PACS operation.

Regarding the export data process, the application did not initially allow data exportation in Excel format. Dicoogle also permits analysing the repository according to the user defined values to DICOM attributes and export the data in Excel format, enabling the indexed data post-processing, (e.g. perform quantitative analysis over the retrieved data) – see Figure 4.5. Moreover, the aggregation of several fields such as number of exams and series was also pre-processed and implemented to allow a fast retrieval of this information.

Extract results service

In order to support federation and easy access to the data by external services and end-users/external applications, new services have been developed, as described in Table 4.1.

Table 4.1: Dicoole web services API extension

Method	***Input Parameters***	***HTTP Method***	***Description***
/tags	-	GET	Get all list of DICOM tags
/image?uid=<uid>&height =<height>	UID of image and height required	GET	Generate a thumbnail or image JPEG based on DICOM image.
/enumField?field=<field> &type=<type>	Name and type of the field (string or float)	GET	Enumerate all the fields that they have stored.
/countResuls?q=<query>	Query of the search	GET	Count the number of results for a specific query.
/examTime?action=<action >	Pre-processing action	GET	Used to do pre-processing related to the exam time and duration-

Some of the indicators require more time, since it is necessary to process all the indexed data. For instance, the *examTime* service analyses the entire repository in order to extract the total duration and number of medical images that each exam contains. The results are stored in a separate file, which can be used later to faster extract these metrics. The reason to create this pre-processing method is due to the excessive time consumption of the data processing. The following actions can be executed:

- **getState:** returns the state of the service. It can be EMPTY, RUNNING or READY;
- **startCalc:** this action creates a thread with the pre-processing;
- **stopCalc:** stops the processing thread;

- **percentage:** returns the percentage of the task;
- **download:** allows to download the pre-processing file in CSV format;

4.3.2 Network sensor

In digital imaging laboratories there are several types of information flowing between different equipment, workstations and devices. The workflow differs from institution to institution and some processes can be optimized. Nevertheless, this optimization is only possible if there is a way to record the actions of the different actors. There are many approaches to execute the logging mechanism such as network traffic analyser, logging each device, or logging the main servers and record each user action. In order to achieve this, a network sensor able to log all DICOM Traffic was developed.

To analyse medical imaging workflows, a DICOM Network sensor [165] was developed. It captures the network traffic, parses DICOM packets and registers relevant events between PACS components. It allows to collects the network events and enables the user to extract knowledge from them. This sensor is a standalone application written in Java that can run in several platforms and support multiple database engines to store the collected information. With this sensor, it is possible to study the workflows of imaging departments, follow up clinical protocols or imaging quality parameters, trigger email alerts based on specific events or DICOM tag values, and obtain real-time metrics of network performance.

Figure 4.6 shows the main components used for this sensor. To capture the network packets we used Jpcap library [15]. Captured packets are parsed with a developed DICOM sniffer. Information extracted is saved in the database and detected DICOM objects (e.g. C-Store transfers) are parsed with DCM4CHE2 library [16]. This library is used primarily to extract the elements present in the DICOM objects, for instance, dose radiation [166].To store the collected information, we used an SQLite database with sqlite4java SQLite wrapper or MySQL.

Jersey[1] is an implementation of the JAX-RS API to build RESTful Web services. Grizzly [2] is a simple Web server container that runs the Jersey Web services. These components were used to create Web API to allow external tools to communicate, namely the Web integrator platform. HTTP with SSL was used to bring confidentiality and authenticate communications.

Summarizing, this component is able to assemble the DICOM traffic and register the C-STORE, C-FIND and C-MOVE commands in an SQL database. Moreover, it can recreate medical image objects from the captured traffic. Thus, the actions of the digital

[1] https://github.com/jersey/jersey/
[2] https://grizzly.java.net/

imaging laboratory workflow can be easily logged in a standard way. The developed sensor can also be orchestrated in the API of the centralized center and the sensor will be able to supply data that can be explored and correlated with multiple data sources, in order to optimize existing resources and ensure the quality of the services provided.

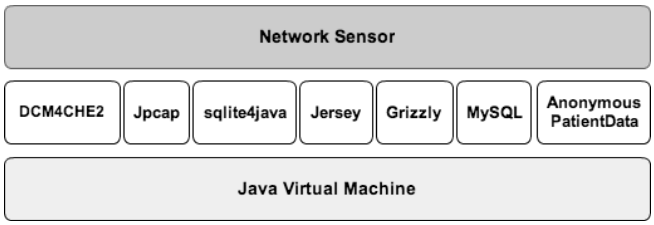

Figure 4.6: Sensor block diagram

4.3.3 Modality specific sensor

It is possible to create specific sensors for medical reports. Nevertheless, the RIS/HIS store the information in different formats and storage ways. Moreover, the stored data have a large set of fields. Modality specific sensors are supported by common data elements (CDE) that were inspired on the HL7 RIM (Reference Information Model) [144]. Due to the integration requirements between different types of sensors, there are specific data to each specialty implementation (Figure 4.7). To overcome this heterogeneity, and relying on the entities of the HL7 RIM, we created a minimal dataset that is supplied to all sensors' instantiations. Moreover, besides the variety of implementations and modalities, there are also other fields that are only stored in the medical reports and clinical databases. For those fields, modality specific sensors include an extension mechanism that supplies the aggregator with a subset of fields that will be retrieved in that particular sensor.

Web services API

The modality specific sensor provides a RESTful API that is able to manipulate XML or JSON input or output parameters, depending on the HTTP request headers. Four service modules were developed: management, location service, search and pre-processing.

The search service is very important to the echocardiogram sensor. There are two types of queries that can be done, the free text and the advanced query. To improve the performance of the search service, the response only includes the patients id, name, exam type, report id and accession number. The user can access the complete report information by exporting all the fields using a specific API.

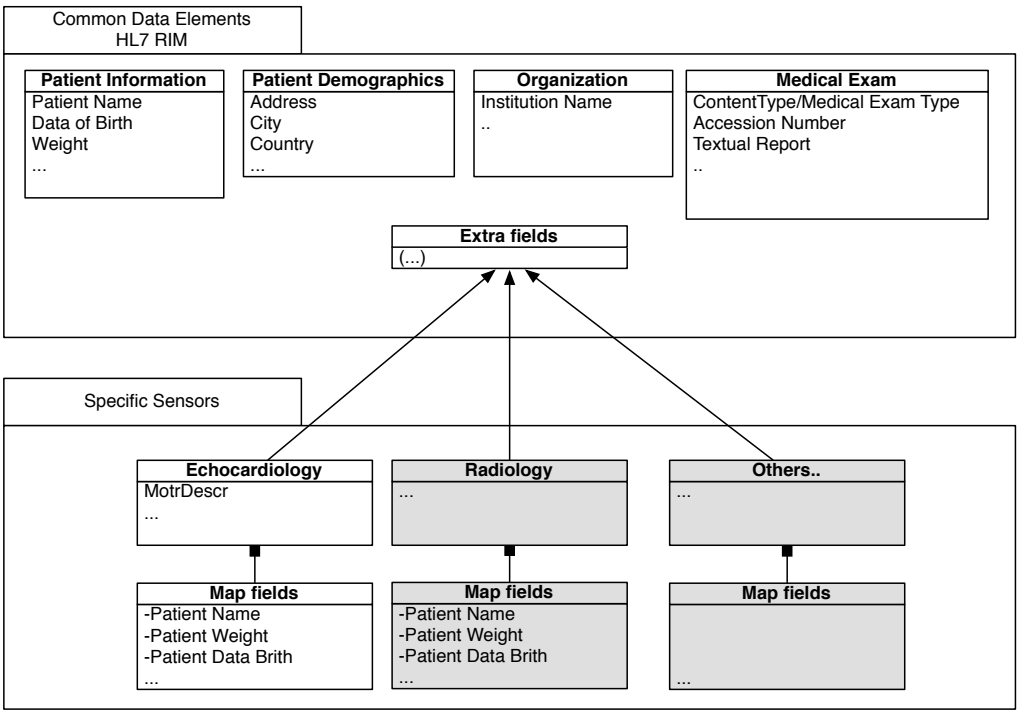

Figure 4.7: Modality Specific Sensors: entity mapping

Cardiology sensor

A specialized sensor was developed to integrate echocardiography data with other sources, such as medical images and network traffic related to medical imaging transferences. The cardiology sensor was developed as an extension of modality specific sensor. This integration will allow an assessment and an audit in a cardiology department, which will be presented and discussed in section 4.4.

There are several types of echocardiogram examinations such as transthoracic, transesophageal and stress [167]. The images produced by the acquisition equipment are stored in the PACS archive. Physicians revise these medical studies using visualization workstations and produce a report in the RIS or in the HIS. The produced report corresponds to a medical exam, but they are not stored in the same repository. The medical exams are stored in the PACS archive, while the textual report/patient episode is stored in the RIS/HIS. In our case study, reports are supported by a commercial HIS, following a non-standard format. These reports contain different structures for each type, i.e. transthoracic, transesophageal or stress, due to their different requirements.

For this particular case, it was necessary to develop a specific sensor to access the echocardiogram reports stored in non-standard format, which will not work straightforwardly in other reporting systems. However, the proposed interface supports attributes-value pair elements and, therefore, the costs of adapting the sensors to new systems are reduced. Access to the repository, an Oracle database, was through an ODBC connection with read-only access (Figure 4.8). This database contains a large amount of

clinical reports, but only three types of echocardiography reports were extracted: transthoracic, transesophageal and stress echocardiogram. These reports have different structures, requiring the development of distinct parsers.

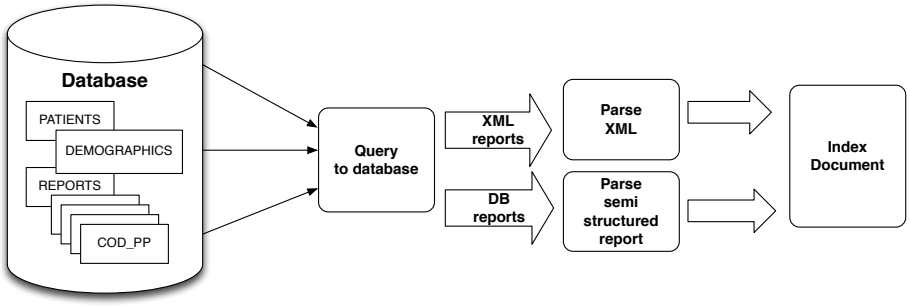

Figure 4.8: Access and indexing process of echocardiogram reports

Gathering information from this kind of information system means we must understand the database data model and build a specific ETL process to aggregate data. Each report contains different data structures and many have database field instances with an XML structure inside. For instance, the transesophageal echocardiogram generates a report with 60 fields and a stress generates one with 140 fields. Therefore, it was necessary to create several parsers and regular expressions to create normalized structured reports. Moreover, the report structure has changed over the years.

This situation forced us to develop a flexible architecture to support distinct reports. New parsers are defined as new classes in XML files. The idea is requiring the minimum effort to accommodate a new report. To enforce normalization, a translation table was also used for the specific fields numerated in the information system, without a match to a human readable value. In the case study, the classes "TransesofacicXML", "OverloadEcoXML" and "TransthoracicXML" were specified.

The sensor was developed in Java and the component diagram is presented in Figure 4.9. Jersey and Grizzly were used to supply the necessary web services to communicate with external tools. The MySQL database is used in the indexing process to obtain information from the reports.

The indexing of all reports may be a time-costly process, due to the high amount of I/O (Input and Output) and processing tasks. To take advantage of multi-core systems, a multi-thread indexing mechanism was developed.

The indexing mechanism relies on a Lucene engine and it supports queries with wildcards and logic operators. Below are some examples of queries that can be explored in the echocardiogram sensor:

- "Alex*": search in free text that returns all documents containing fields starting with "Alex".

- "PatientName:Steve": returns all documents of the patient "Steve".
- "ContentType:ETE AND Weight:Numeric:[70 TO 90]": Returns all transesophageal reports of patients weighing between 70 and 90 kg.
- "ContentType:ETT AND ExamDate:199907* AND NOT MotDescr:Arritmia": Returns all transthoracic echocardiogram reports performed in July 1999 whose subject does not match arrhythmia.

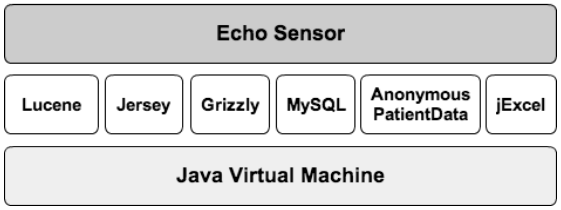

Figure 4.9: Software components of Echocardiogram sensor

The presented sensor can access data from a cardiology department. In this scenario, the stored data format changed along the time, due to several software upgrades. To deal with this issue, a normalization process was adopted and several data parsers were developed.

Not all clinical data are stored and managed by the PACS. Each time medical images are analysed by physicians, a medical report is produced and medical images are classified according to the diagnosis (e.g. arrhythmia, cardiomyopathy). New sensors can be developed taking advantage of a baseline plugin that already has the necessary methods to supply the web service API, handle queries and the documents' index. Nevertheless, depending on the data source, the list of fields available in the source is different, such as ContentType, MotDescr and many others. Thus, a list of fields should be provided through a well-defined web service, which will be presented in the graphical interface of the aggregator system.

4.4 Results and discussion

In this section, several results and case studies that support the main requirements of the proposed approach are discussed. The presented solution allows the clinical researchers to apply a straightforward methodology, while collecting data and extracting knowledge from them. The proposed solution was evaluated and validated in hospital centres, while collecting real data to be used by the clinical and biomedical researchers.

4.4.1 User interface

The Medical Workflow Analyser is an application developed to aggregate the information from several information sensors from one or more hospital departments. This application allows users to perform searches and relate the results from disperse data

sources. Users can build queries using free text or a set of keywords. Moreover, they can search in one source or over federated resources. For instance, in Figure 4.10, there is a query to the echocardiograms and a button to "Search images" that will propagate the query to the PACS archive sensors. Several indicators of performance and quality of service are provided by the applications. Health professionals suggested the metrics supported by the platform.

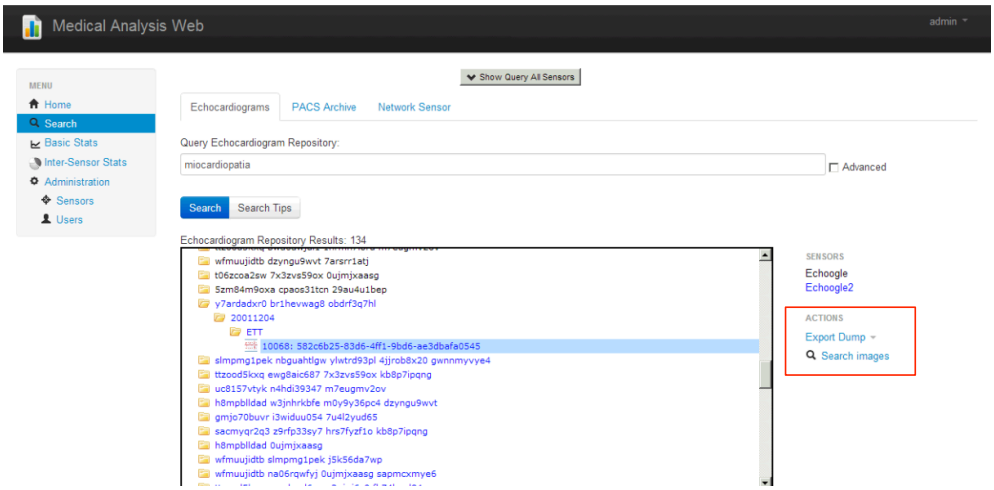

Figure 4.10: Search example of the Medical Imaging Workflow Analyser

The administrators can manage the set of sensors that are centralized in one application. This may be accessible in the authorized computers over the institution's network, according to the access control policies.

The analyser also supports actions over specific sensors. For instance, using the network sensor it is possible to monitor, in real time, the PACS network activity. Figure 4.11 presents part of the user's dashboard showing the number of DICOM commands executed in an hour. The scale of the various indicators is automatically adjusted, if the number of each type of command exceeds the limit. It is possible to see that, during this period, only three searches using the C-Find command were performed, 9 images were transferred with C-Store commands and 8 retrievals with C-Move commands. No C-Get commands were executed.

4.4.2 Network DICOM sensor: performance measurements

To evaluate the performance and reliability of the network sensor, a test scenario with the dcm4che2 tools and a script that contains 3 different C-FIND requests in a loop was created. We used a network of 100 Mbps between the client and the PACS archive server that provides the Query/Retrieve service.

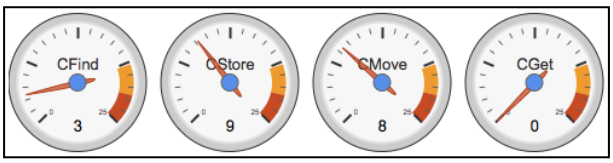

Figure 4.11: Indicator of the traffic rate for each DICOM service

The network sensor was running in a machine with Windows XP inside a virtual machine using VirtualBox with Inter Core 2 Duo a 2.2 GHz. The virtual machine has been limited to 1 core and 768 MB of RAM to limit the performance of the test machine.

In total, 60.000 C-FIND requests were performed against the storage server with 20 simultaneous associations of DICOM. These requests generated a total of 80.000 C-FIND responses. Taking into account the DICOM standard, around 580.000 packets have been transmitted over the network, ignoring the TCP packets to establish the session,

All of the 60.000 have been captured and analysed with success by the network sensor. In the response of the 80.000 average answers, only one was not captured by the sensor. The success rate to analyze the C-FIND answer was of 99.99875%. The tests have been executed in 13 hours, 48 minutes. The sensor needed 5 hours, 58 minutes to process the packets, analyze and store them into a MySQL table.

4.4.3 Auditing a cardiology department: a case study

The Medical Imaging Workflow Analyser was used to extract relevant metrics regarding the quality and productivity of the case study laboratory. For privacy and policy reasons, we will not reveal all details of the results obtained. However, we will explore some high level summaries of these data, which have clinical relevance for the improvement of the workflow in this institution.

The medical informatics scenarios empowered by index engines are promising and this technology supports the sensors described on this chapter. The sensor of echocardiogram reports allows collecting information of an Oracle or MySQL database and indexing it in Lucene. In the current case study, 84063 reports of 3 different schemas were considered using the same indexing step. This test used a machine with 1 core and 2.67 GHz with 8GB of RAM, implemented in Java. The produced index was anonymized and the supporting database was ciphered with an AES algorithm. The 84063 reports were indexed in 4 hours and 21 minutes. The result index volume was 261 Mbytes.

In Figure 4.12, it is possible to see the distribution of imaging quality according to patients' age. We can conclude this is a factor influencing the imaging quality of an echocardiogram. The same trend is also observed in other examination techniques based on ultrasound (Figure 4.13). Many other analysis are available, for instance, distribution

of the quality per patient weight. We conclude that patients with a higher body mass index tend to have an exam of poorer quality.

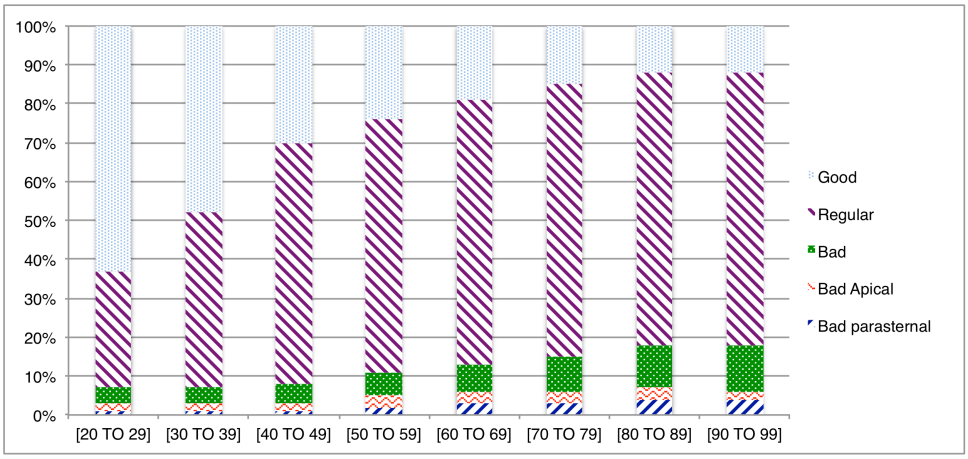

Figure 4.12: Distribution of patient age and quality metrics of medical images

Figure 4.14 shows metrics relating information obtained by the reports sensor with DICOM repository sensor. It was possible to relate various imaging studies with appropriate reports of 10,463 echocardiography examinations. The left-hand graph relates the perceived quality of the exam with its average duration. The right-hand graph shows the average number of digital medical images acquired for each type of perceived quality. These results are only obtainable, because we combine information extracted from both sensors. The duration of the exam was extracted with the medical imaging sensor, i.e., Dicoogle, while the quality is extracted from echocardiogram reports. Finally, it is possible to assess that the poor quality exams have a longer duration, than good quality ones. In addition, good quality exams have, on average, more images acquired.

This framework allows the extraction of many types of metrics that are useful to medical administrators for, for instance, planning and managing when new equipment is acquired, or analysing the laboratory's history with a qualitative assessment. Moreover, it can be used to improve examination scheduling and quality of care in patients' allocation, taking into account their physical condition parameters such as their weight.

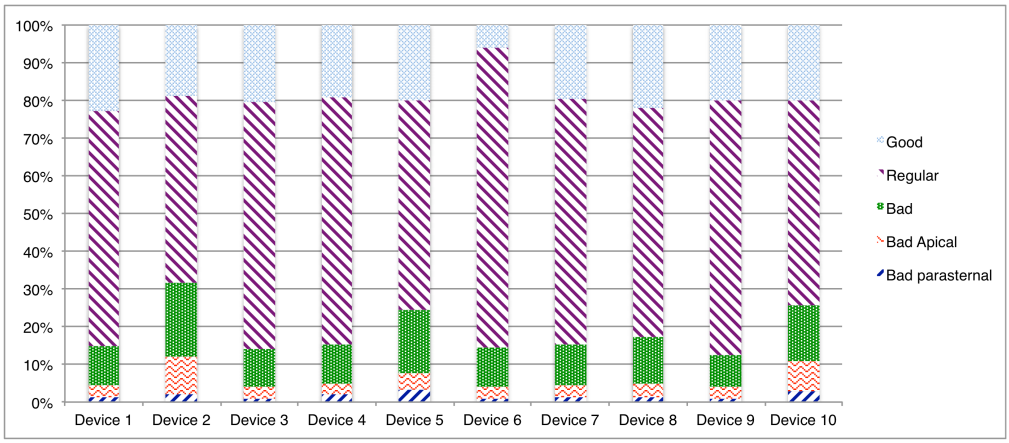

Figure 4.13: Quality of each echocardiograph

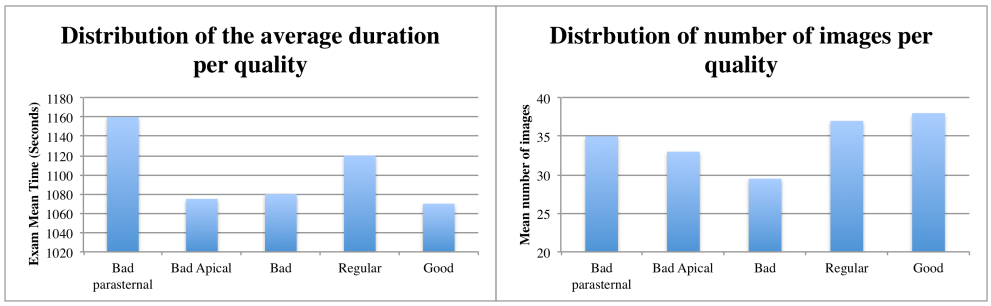

Figure 4.14: Average duration of exams and quality, and average number of images acquired with their quality

The de-identified indexing of the echocardiogram reports allowed us to do some metrics that were not possible in the original information systems. For instance, measure the quality of the exams per patient age. It is possible to see, that the quality decreases with the age of the patient (blue is the best quality and purple the worst). The probability of an echocardiogram having "regular" or "good" quality is smaller for an older patient. For instance, a patient with age between 90 and 99 has only 25% probability of having a "good" or "regular" exam.

In Figure 4.15, we verified the quality of medical images for all types of echocardiograms. Measuring the quality for each type of echocardiogram is easy with our system and is only necessary to change the query to include a filter to the report type. It is only possible to do a query over data from the PACS archive or echocardiogram reports to achieve the produced result.

These results use the location service provided by the echocardiogram sensor. The results' distribution can be designed in the map. Due to the flexibility of the proposed system, the results can be filtered by a structured query. For instance, only patients with an age between a range; or with an exam with particular parameters such as weight or any echocardiogram value.

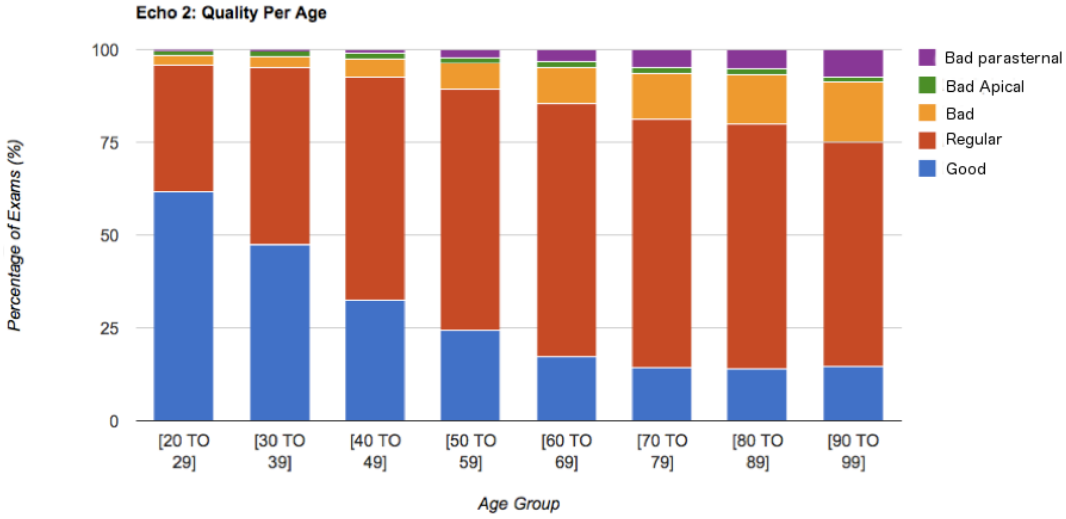

Figure 4.15: Distribution of the patient age and quality metrics of of medical images

Finally, we decided to integrate a geographic tool in the proposed framework. For instance, Figure 4.16 shows the distribution of patients according to their residence. This example uses the location provided by the echocardiogram sensor. Due to the flexibility of the proposed system, the results can be filtered by a query enriched by structure filtering, e.g. only patients in a specific age-range; or patients who have an exam with particular parameters such as weight or any echocardiogram value.

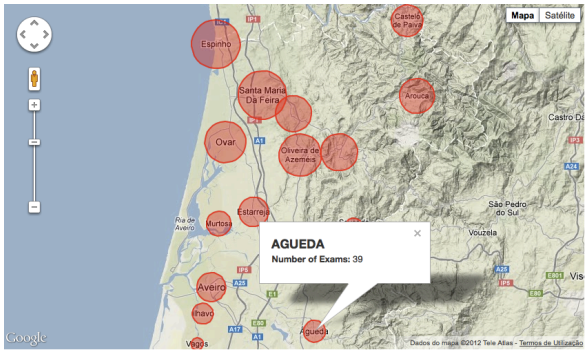

Figure 4.16: Demographic patient distribution in overall of the case study repository

4.4.4 Collaboration case studies

Radiologist clinical researcher

In medical imaging, one potential use cases for DICOM metadata was brought by Milton Santos, one researcher from data mining in medical imaging based on DICOM Metadata. One of the problems that he faced in his PhD was how to access the real medical imaging over the PACS, and how to analyse the images and their metadata content in order to suggest metrics and changes in the healthcare unit that have real impact in the patient daily life. One of his initial approaches was to use CD/DVD of the institution and analyse

a small subset, which are not relevant in a statistical point of view. This project has been developed in partnership with researchers from ESSUA (Escola Superior de Saúde da Universidade de Aveiro), and several healthcare institutions and around 30 millions of medical images were collected.

Echocardiography based left-ventricle dynamics assessment

Another project was to analyze the movement of the left-ventricle based on echocardiography with specific pathologies, such as myocardiopathy. The problem was to have the images from a specific pathology, which was difficult to achieve in a massive way. A new partnership was established to collect case studies and datasets with specific pathologies were created. New contributions in echocardiography have been assessed [168].

4.5 Final considerations

Our proposed framework can extract information from distinct data sources within a medical imaging laboratory, using a network of workflow sensors, DICOM objects' metadata and examination reports. This information is correlated and integrated in a centralized unit. The result is a federated information system, available through a Web portal, which allows free text queries similarly to popular search engines. The platform offers monitoring tools and can be used as an auditing system, measuring the efficiency and quality of healthcare services.

This framework is unique because it provides an integrated view over data sources that are usually unavailable. Using this tool in a real environment can result in gathering important data for professional practice improvement and service administrators. It permits the identification of factors that may contribute to a healthcare deficit. Moreover, it can contribute to improvements in practices such as patient safety programs and image quality control initiatives. In our case study, the solution allowed us to identify some characteristics and behaviors not perceptive until now, and confirm others that were empirically inferred. The accession number can be used in the integration of our case study, because it exists both in the DICOM images and in the echocardiogram reports. However, depending on the clinical workflow, the merge of this number can be challenging.

In the case studies explored in this chapter it was possible to understand that they adopt standards such as DICOM and HL7, but the real implementation in the clinical hospitals creates a gap between the researchers and the real data. Not only the access to the data sources is needed, but also to understand the dynamic of the network flows that are happening, the peaks of data access, the network paths where most traffic flows, the dimension of the network and the machines that have access to the data. The risk management is needed to take into account. The developed algorithms to gather the data

should measure the impact of the processes in the continuous work of the practitioner to avoid slow connections to the archive or any other potential issues. It is one of the major difficulties while accessing data sources. There are several constrains that need to be assessed before starting the process, and some of them regard privacy issues, risk management and study ways to take action without an intrusive method.

After all, to access real data, administrative issues and direct contact with all the commission boards with the required permissions is needed. For such process, the communication with IT staff is the first step, because they will provide the credentials along with other required assets. One of the important issues is the view that other people have over the data, as it is needed to certify that only read-only permission were given over the data repositories (even if they are in the file system, SQL engine, or any other data structure).

The presented results are only for demonstration of the developed system, but other metrics are being extracted from the integrated repository to support clinical research. With the developed system, cardiologists can perform other studies without the need to change any component, due to the flexibility and features that allow exporting data to CSV. Our method can be generalized to other RIS and HIS, by creating other parsers for each distinct clinical report solution.

The integration of this application in a healthcare institution is effortless, transparent and has a great potential to optimize workflows and services provided. Nevertheless, there are many issues to solve, namely in semantics and information normalization [4, 5, 166]. For instance, different metric measurements and different nomenclatures can represent the same information. Moreover, it is also possible to have data stored in private attributes, which creates major problems when implementing a vendor-neutral solution [169, 170]. Thus, normalization processes are usually manual, executed by clinical researchers and published in several scientific articles [166, 171-175]. Furthermore, the challenge was also explored in the satellite of this doctorate addressed in [19, 27].

To conclude, the proposed architecture to integrate healthcare information in this chapter works real scenarios, as already validated, for clinical researchers. However, it can only be applied to regional studies, closed consortium or agreements partners. Strategies to integrate the aggregated data for a wide range variety of institutions are still needed.

5 Software architecture to explore patient-level data

"Not only does God definitely play dice, but He sometimes confuses us by throwing them where they can't be seen."
Stephen Hawking

The integration of biomedical databases is in the root of our problem statement. While in the previous chapters the integration of data between different silos has been achieved for specific data types, in this chapter we intend to integrate undefined types of data sources over an undefined number of healthcare units[1].

The quantity of clinical information and disease-specific data has steadily increased over the last decade. This information is fragmented over dispersed databases in different clinical silos around the world. However, as the awareness about the potential of these data for clinical research increases, the need for solutions for secure sharing of

[1] This chapter is mainly based in the following publications: *Architecture to summarize patient-level data across borders and countries, MedInfo 2015, São Paulo, Brazil [20]*, and *Challenges and Opportunities for Exploring Patient-level Data, BioMed Research International [176]*.

information across different databases also grows. This reuse of data will lead to many benefits, mainly for clinical and pharmacologic researchers [13, 76, 145].

Several partial solutions have been proposed to solve the problem, from centralized to federated approaches with distinct security levels and different access roles [7, 177-179]. However, clinical researchers struggle to find datasets or clinical trial candidate subjects that fulfill their specific research questions [180]. Moreover, this problem is not only due to privacy issues, but also because, most of the times, appropriate datasets are not easy to locate, or their existence is unknown.

In this chapter, we propose a software framework that is able to summarize and aggregate databases content, which is not dependent on their type or application. With this solution, patient health data can be kept privately in each healthcare institution, but, at the same time, it allows clinical researchers to query databases at several layers of information detail, from high-level to aggregated characteristics.

5.1 Related work

Several projects have been created all over the world with the aim of integrating, organizing, and extracting information from biomedical databases.

5.1.1 Initiatives to explore biomedical databases

In this section we cover the most significant initiatives, considering our research goal, looking for large-scale international projects in the literature. The sponsors of these projects are mainly NHI (National Institutes of Health), IMI (Innovative Medicines Initiative) and/or European Commission. The analyzed projects are large-scale international projects containing partners from academia and business sector in U.S. and Europe. Figure 5.1 shows a simplified version of the main scientific contributions in the exploration patient level data relevant for this work and also points out the most important key projects.

In U.S. REDCap (Research Electronic Data Capture) [181], for instance, provides a scientific research workspace for translational research allowing users to manage online surveys and databases. This web-based application is mainly a data repository that supports exporting the data to SPSS, R, SAS and Stata formats. It is being used by more that 90,000 users, for the management of more than 30,000 research studies. Mini-Sentinel [182, 183] is another project that aims to create an active surveillance to monitor the safety of FDA-regulated medical products. It uses pre-existing electronic healthcare data from multiple sources. The developed tools provide features to identify and validate medical products' exposure and potentially associated health outcomes. There are isolated software tools, used mainly inside an institution, that require a great expertise for setup. Another U.S. based initiative is Bridge-To-Data [184] which offers services that allow

users to identify key features and compare database profiles. This resource also serves as an educational tool for public health research and as a template for health systems planners to design or refine their healthcare data. It provides a unique searchable and comprehensive compendium of information on population healthcare databases worldwide, allowing the healthcare professional to obtain profiles on various population datasets on a single website.

In Europe, there also many projects over the years that contributed to the patient databases exploration. BBMRI (Biobanking and Biomolecular Resources Research Infrastructure) was a project between 2008 and 2011. The aim of this project is to build a European and internationally biobanking research infrastructure [185]. The infrastructure includes samples from patients and healthy persons, which represent different European populations with molecular and genomic resources. The key point of BBMRI is to help increasing scientific excellence and efficacy of European research, while expanding competitiveness between research units and global industries. In 2013, a new BBMRI-ERIC Inauguration Conference was held. BBMRI-ERIC was created to facilitate access to quality-defined human health/disease-relevant biological resources and to provide quality data in an efficient, ethically and legally compliant way [186]. Following the authors, it will only be achieved by reducing the fragmentation of the bio-medical research landscape through harmonization of procedures, implementation of common standards and fostering high-level collaboration, contributing in the end towards a stronger Europe's cohesion policy.

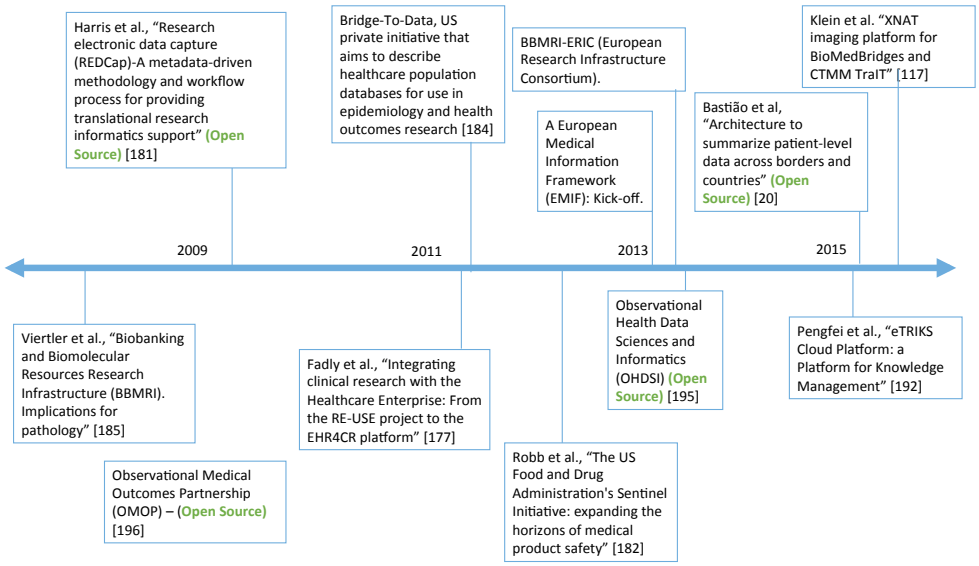

Figure 5.1: Related project initiatives timeline: mainly their important scientific outputs in the literature

There is a growing realization that the development and integration of EHRs for medical research can enable substantial efficiency gains, making Europe more attractive for R&D investment, whilst providing patients with facilitated access to innovative medicines and improved health outcomes [187]. The key challenge to achieve the integration between

EHRs, existing research platforms and healthcare networks are the countless legal, ethical and privacy requirements, when providing a platform that works across many EHR systems, especially when trying to be sustainable within a scalable business model [188-191]. EHR4CR [177] is one of the projects with that target, which aims to build a platform (systems, organizational structure, data interoperability, governance model, etc.) to demonstrate the viability and scalability of the EHR integration as a business model through pilots (e.g. platform services designed for protocol feasibility, patient recruitment, and clinical trial data capture). EHR4CR supports the feasibility, exploration, design and execution of clinical studies, while enabling trial eligibility and recruitment criteria to be expressed in ways that permit searching for relevant patients across distributed EHR systems, and initiate confidentially participation requests via the patients' authorized clinicians. It aims also to provide harmonized access to multiple heterogeneous and distributed clinical (EHR) systems; and integration with existing clinical trials infrastructure products, guaranteeing improvements of data quality to enable routine clinical data to contribute to clinical trials, and, more importantly, vice versa, thereby reducing redundant data capture.

BioMedBridges [116, 117] is a still a on-going project that empower the creation of an e-infrastructure to allow interoperability between data and services in the biological, medical, translational and clinical domains and thus strengthen biomedical resources in Europe. With this, BioMedBridges intends to launch a new environment, where the data and services exchanges between biological and medical sciences can be of critical value to clinical and industry stakeholders. To ensure access to relevant information across all biomedical sciences research infrastructures it enables scientists to conduct and share cutting-edge research. BioMedBridges may act as a major catalyst for realizations of the European Research Area (ERA). This initiative will provide access to all the biomedical sciences data and tools across Europe, allowing the full potential of research in all countries across the ERA to be realized and will propel Europe to the forefront of Biological and Medical Science research globally, while simultaneously improving Europe's competitiveness in the biomedical sciences and health-related economies. Finally, the projects have also several outputs in the medical imaging research structure to share medical imaging across other projects and countries [117].

The eTRIKS project arises in the context of IMI's launching of 30 projects, many of them involving the integration and analysis of diverse types of biological and medical data from a range of sources. This implies to create a Knowledge Management platforms that not only store data, but also facilitate the comparative analysis of different data types and the use of advanced analytical and modelling tools [192]. Within IMI there is currently no common Knowledge Management platform and no provision of services that can support data intensive translational research. The project has already outputs through the

exploration of Platform-as-a-Service and Software-as-a-Service to supply a Knowledge Management over Cloud computing services [192].

In the sequel, two important projects are described in more detail: the European Medical Information Framework (EMIF) and Observational Health Data Sciences and Informatics (OHDSI).

5.1.2 European Medical Information Framework (EMIF)

EMIF (European Medical Information Framework) is a joint IMI project supported by the EU and the European pharmaceutical industry (through EPFIA), which has the overall ambition of enabling efficient re-use of health data for research purposes. It will allow the creation of new features to re-shape the way researchers answer key questions and also figure out possible future research directions [193].

The framework that is being developed (EMIF-Platform) is supporting two important research topics (RT) that serve as exemplars and test cases: biomarkers for obesity prognostic and its metabolic complications (EMIF-Metabolic), and biomarkers that may explain the development of Alzheimer's disease and other dementias (EMIF-AD) [194]. Through the participating data sources, the project also aims to explore how the massive data available in pan-European EHR systems and research cohorts can be optimally leveraged to improve biomedical research. In this context, there is a need to integrate medical records, clinical and other omics information from different sources. The interaction between the EMIF Platform and RT/EHR is crucial to achieve success. It is expected that the EMIF Platform will facilitate the access and the effective use of data, while the RT provides focus and user guidance, fundamental to achieve a sustainable platform development.

The EMIF Platform intends to be an integrated platform to allow users to browse information at different conceptual levels. It will allow rapid exploration of a wealth of readily available information. The browsing capabilities of the EMIF-Platform system will allow the user three distinct levels of data "zoom" (Figure 5.2). The level 1 is from the perspective of available data sources based on the data catalogue. It is populated with DB fingerprints, i.e. a general characterization of the DB for indexing and retrieval purposes. The level 2 drills down to the level of aggregated counts in databases. The PDE (Primary Data Extraction) will extract aggregated data of several DBs and provide a first level of a Private Remote Research Environment (PRRE). Finally, the level 3 allows drill down to the level of individual patients in those databases. That is, the user should be able to "zoom in", from the whole populations to de-identified patients. The PRRE provides a secure place for handling patient level data. Three is still a security framework that is transversal to all the other modules and will be used according to the security constraints for each scenario.

By the end of the project, it is expected a new platform environment which will enable this EMIF ecosystem, allowing dedicated software, such as the clinical information browser or the project knowledge base to be built.

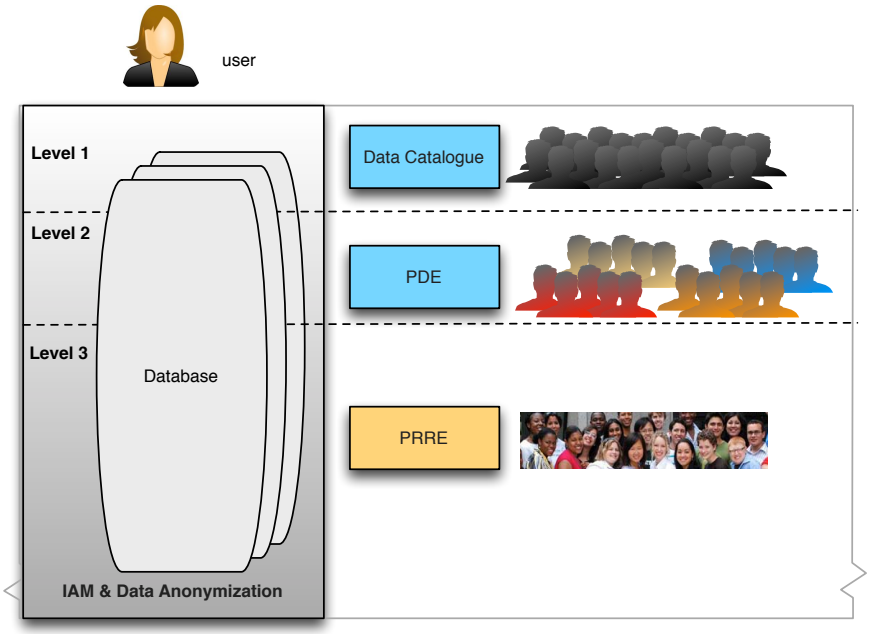

Figure 5.2 – A simplified schema of EMIF Platform main components.

5.1.3 Observational Health Data Sciences and Informatics (OHDSI)

The Observational Health Data Sciences and Informatics (OHDSI, a.k.a. *Odyssey*) is an open under joint initiative program, which main idea is to facilitate the analyses of large-scale datasets [195]. This is an active worldwide initiative that proposes new solutions for data gathering and aggregation, promoting a Common Data Model (CDM) for the patient-level databases representation [196]. The idea is to provide a standardized data model that allows performing queries across a set of databases [197].

Besides the common data model, the OHDSI community have been also developing several analytic tools, such as Achilles, Achilles Web, HERMES, CIRCE, to simplify the exploration of data (Figure 5.3). HERMES is a vocabulary browser over the CDM. CIRCE is a web application that allows defining the population of interest, using specific inclusion/exclusion criteria. Achilles main goal is the characterization of observational databases by aggregating patient-level data. It provides powerful statistic features based on R scripts. OHDSI also developed Achilles Web, a web application that presents summaries and statistics in a navigation/dashboard model, in order to simplify the researchers interface to Achilles.

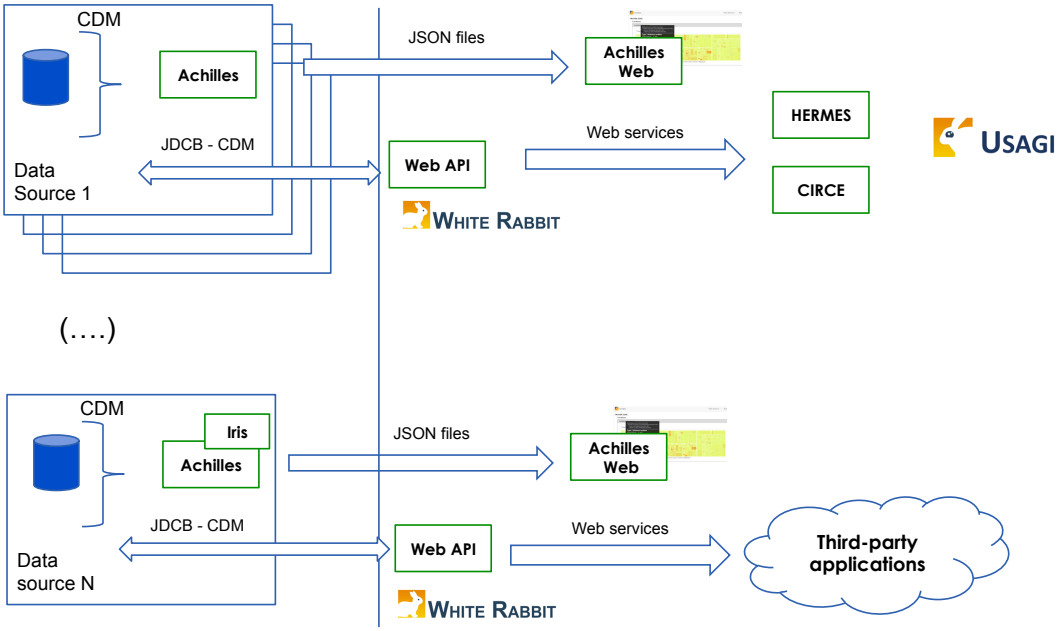

Figure 5.3: OHDSI components architecture

Finally, to complete and extend the communication through SQL, which is not adequate for modern application, they are also developing a Web API, which will allow an easier communication with third-party applications. All solutions being developed by OHDSI are open-source and available for free[1].

5.1.4 Challenges and opportunities

Although the previous discussed projects are examples of already developed solutions for database profiling, they do not perform a deep summarization and exploration of automatic tools to extract patient-level information [198]. Moreover, there is still a gap between clinical research communities, which blocks the scientific researchers to continue improve the treatment improvement, drug discovery and healthcare quality assurance.

The development of ICT (Information and Communication Technologies) infrastructures for the federation of resources, including dynamic data models to support various analysis and visualization procedures, is crucial for the evolution of the medical research. A system to collect detailed summarized views, and ask for high level questions regarding the databases, is still needed. Moreover, the re-use of already existent tools should be possible and also the capability to drill-down in several levels of patient data. Europe and world wide citizens may then benefit from improved health-care and medical research conditions [176].

[1] https://github.com/OHDSI

To address these requirements, our main goal was the development of an information framework capable of integrating patient health information from different healthcare and research communities, at a level that is currently not available.

5.2 Requirements to summarize health databases

Following the literature and project analyses, the patient-level exploration is indeed a need and a trend nowadays. Moreover, as partners of EMIF, mainly responsible for the EMIF Platform, we needed to understand all the requirements in detail. From the evaluation of the state-of-art and the interactions with various EMIF end-users, we uncovered a set of requirements that drove the definition of a new methodology to summarize health databases. The development of the EMIF Platform follows a double approach (see Figure 5.2): the blue track focuses on "population based datasets" and the orange track on "research cohorts". The final challenge is to combine these two tracks in a single architecture.

While the summarization of patient databases is needed, there is still no clear vision how to collect them. In the EMIF project, with collaborations with several partners, it was possible to define a methodology and a workflow process to provide new database summary schemas. Moreover, it was also necessary to define how to populate the summaries, taking always in account privacy and confidentiality issues that are often barriers for data sharing. The integration of fingerprints from different databases allows researchers to find databases suitable to their needs from a single access point.

5.2.1 Functional requirements

Fingerprint definition

The system to be developed needs to support dynamic creation of fingerprint schema, i.e., a set of questions that should adequately characterize each database type (e.g. EHR data sources) – see Figure 5.4. The main idea in the conception of the fingerprint schema was to have a summarized overview of a number of geographically scattered healthcare databases, with the express aim of facilitating the initial assessment and selection of databases for specific research questions. Thus, this system will allow database owners to fill out a questionnaire, creating their own database fingerprint was required.

The template schema of the fingerprint needed to be defined for each database type, such as EHR, cohorts, medical imaging repositories, observational databases, or many others. The system also needs to allow system administrators to dynamically modify the questionnaires of each data source type.

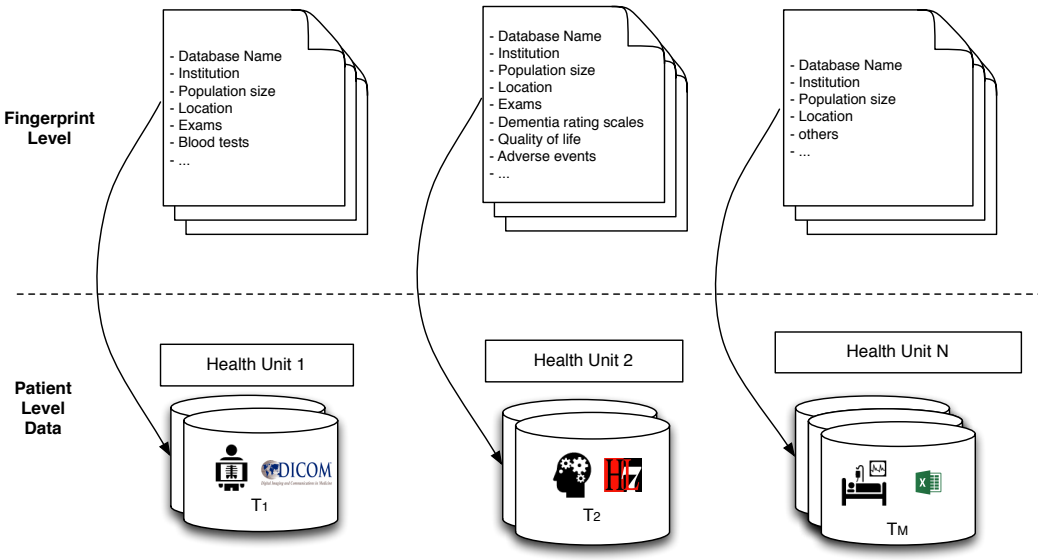

Figure 5.4: Fingerprint Concept

User Profiles

The definition of user profiles and their role in the Catalogue is an important issue in the system design. The user must have access to the features and interfaces of interest according to the research question, and, therefore, flexibility was a priority. We considered three types of users: Researcher, Data custodian and System administrator.

The researcher is able to search over a set of fingerprints. A free text search feature must be provided, so that the regular user can access all contents provided by database owners. Comparison of search results, tabular views and searching for 'similar' databases are also needed. This required the definition and agreement of measures of similarity that trigger filters for specific property matches or various other properties, such as location proximity.

Data custodians are able to manage and access the databases that they have added to the system, and navigate through the corresponding fingerprint data. Due to the complexity of the database information, the user is also able to save the database fingerprint at any point and to edit it over time.

The administrator is able to create and edit fingerprints' schemas, create group questions in the templates, define questions, validate user registrations, and several other management tasks.

Data collection process

While real data are not shared and do not flow out of healthcare institutions' boundaries, clinical researchers are able to query high-level information of the databases. We presented the concept of fingerprint, which is a database characterization schema with a set of questions, such as database population, coding systems, geographical distribution and other relevant information.

Each template may contain several groups of questions. Within these, each question can be individually defined, including the answer type (for instance, multiple-choice, plain-text, yes or no, and many others). Administrators can change each catalogue entry in the administration panel (accessible for users with administration profile), and the application will be automatically updated.

Users can create fingerprint templates using an online assisting tool, or a tabular textual schema (in Microsoft Excel, for instance). The process of building this kind of schemas and allowing collaborative members to contribute without a steep learning curve was a priority when developing the Catalogue.

Based on the appropriate template, database owners have the capability to fill out their database fingerprint, i.e., answers to the questionnaire corresponding to their database type (Figure 5.5).

A key idea behind the creation of the Catalogue is allowing researchers to find specific databases, which are aligned with their research purposes.

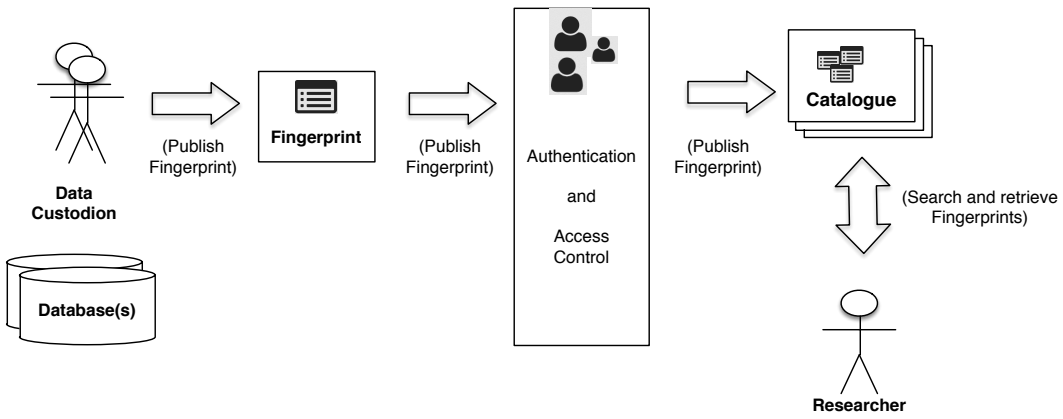

Figure 5.5: Fingerprint description workflow

5.2.2 Non functional requirements

To build an integrated eco-system it is needed to guarantee that all the pieces are communicating together and that all components have the right requirements to work and to correctly use the platform. Furthermore, a security strategy will be applied, the interactions between the components decided, and Single Sign On (SSO) implemented over the components. Finally, the exchange of the information normalization and the harmonization process (captured from heterogeneous databases) needs to be taken into account through a knowledge object library, where the access levels for each user are semantically described.

Database aggregation

As described previously, the idea of a Common Data Modal (CDM) is to provide a standardized data model that allows performing queries across a set of databases [197]. This OHDSI model is gaining increasing importance and it has been adopted by several pharmaceutical companies, and in large-scale projects [179, 196, 198]. The idea is to build tools able to aggregate data from different databases and which can contribute to a global summary view without compromise patient privacy. Finally, real-time queries may be integrated followed the same strategy, but instead of being static aggregations, the system may do aggregations in real-time.

5.2.3 Workflow to define a fingerprint template

The process of introducing a new database schema is described in the Figure 5.6. First, it is required to identify the experts in the field, such as Observational Data, Alzheimer Cohort or any other type of specialist. The requirements should be highlighted and it will be imperative to decide what kind of database will be required. Then, a tabular fingerprint schema should be developed and introduced in the Catalogue, which will automatically generate a Fingerprint Schema for a new community. After being deployed and already in production stage, upgrades to the fingerprint schema could occur, so, merge strategies need to be in place.

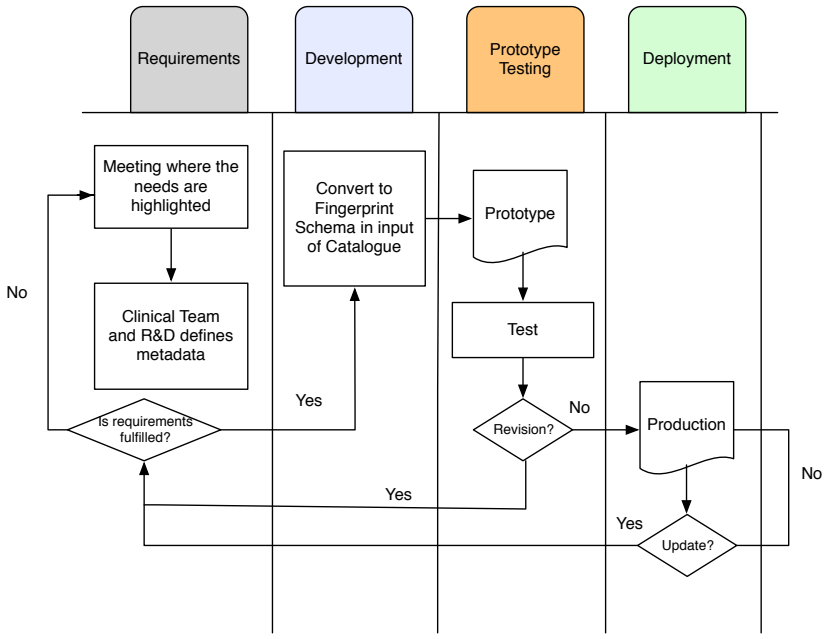

Figure 5.6: A methodology to create new fingerprint schema

5.3 A stratified approach to explore patient-level data

This “drill-down” architecture is presented with a conceptual strategy, according to the EMIF architecture described in section 5.1.2. Nevertheless, a clear vision of the

architecture is needed about how it can be instantiated in real concepts and entities (Figure 5.7). Our proposal is split in four layers, from the higher (no patient level data, only fingerprint data), to the deeper level (patient-level data). These will be detailed in the sequel. In Figure 5.7, the left-hand side shows the general concepts of the EMIF platform, as detailed in previous section while describing EMIF project. In the right hand side, it shows how it can be solved and how the information can flows between different levels of the architecture.

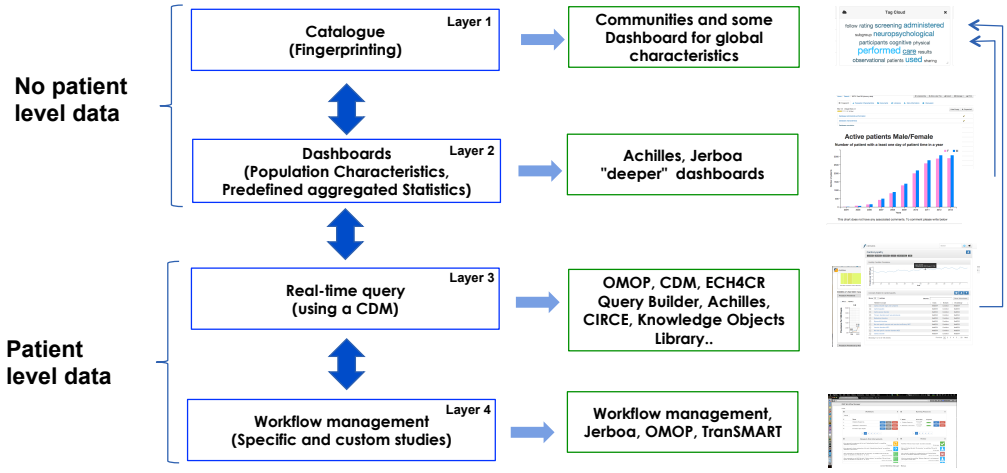

Figure 5.7: Hierarchical proposal to explore patient-level data

5.3.1 Layer 1: Catalogue and global database statistics and dashboards

The Data Catalogue will be used as an entry point of the EMIF Platform. It should have dashboards with global information about the databases that have already been catalogued. It should also have the capability to provide global statistics, not only from each database, but also from multiple databases. Moreover, the idea of the community need to be implemented, because the main goal of the EMIF is to create a European Medical Informatics Framework where not only the EMIF partners can take advantage of, but also new projects can joint and create their own fingerprints templates and insert new databases without mixing the data and fingerprints between communities.

The dashboard capability needs to be extended to support the information provided by third-party components. In this layer, it is possible to aggregate new information coming from low layers. The core component allows creating extensions to the dashboard, and evinces the results from different layers (applications). The low layers will be able to feed the entry point presenting some global statistics and dashboard that could be dynamically updated, taking into account, for instance, the fingerprints and the CDM/WebAPI provided by the OHDSI.

5.3.2 Layer 2: Dashboard for the aggregated data

While the global information collected in the database fingerprint is crucial, it is also important to have more specific information for each database. In this layer, it should be possible to show database meta-information, collected through sets of questions, which may include a set of documents, associated literature, and also population characteristics. This dashboard layer must be fed from external tools that process directly the original databases and provide aggregated data regarding general or specific research questions. One such tool is Jerboa[1], which provides extracted population descriptions according to the patient-level databases' content. Moreover, due to the adoption of the OMOP CDM, other tools to extract the aggregated data from OMOP CDM also need to be integrated: ACHILLES and AchillesWeb. Basically, the proposal is to create an extension of the Fingerprint browser (layer 1) to support the data that can be provided by Achilles. Specific, from the Catalogue, it should support the upload and visualization of the both TSV or JSON files (or any common standard).

Besides the integration capability of ACHILLES, the extension for seeding from third-party components should be taken into consideration. An API should be provided to add extensions to the Catalogue/Fingerprint Dashboard (Figure 5.8).

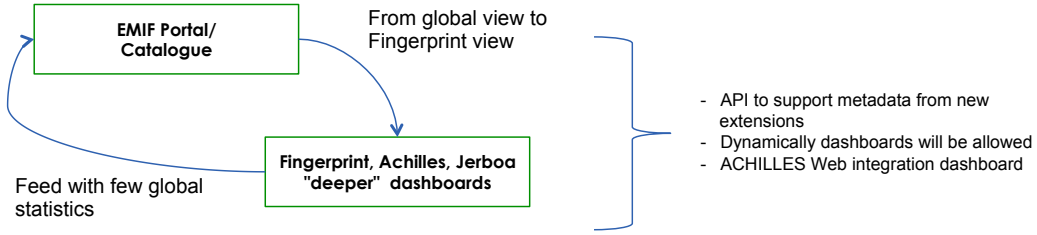

Figure 5.8: Layer 2: communication between the tools

5.3.3 Layer 3: Real-time query

Layer 3 will provide the real-time aggregated studies, already involving patient-level data. For this to be possible the databases that belong to this ecosystem must apply internally and expose data through the OMOP CDM. On one hand, there are also open source tools built in OHDSI that can be used as part of the platform, in a way that allows also the EMIF community to give several contributions, such as WebAPI, Circe and Hermes. On another hand, the EHR4CR project [177] has already a query builder tools that is able to query patient-level data. The adaptation to support the CDM will be developed and instantiated to become part of the whole EMIF Platform. These tools should be easily integrated in Layers 1 and 2 of this stack and thus, an extensible architecture are required.

[1] Jerboa is a tool built by ErasmusMC, Netherlands.

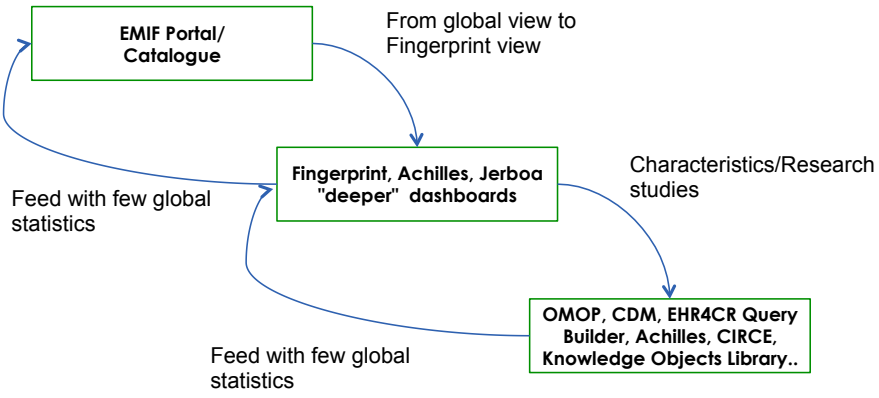

Figure 5.9: Layer 3: Real-time query

5.3.4 Layer 4: Workflow management

While the real-time query is important to have access to patient-level data, the authorization process, the confidentiality agreements, and the whole workflow process need to be tackled. Basically, we need a component that mediates the federation of data sources from distinct database providers and allows the query builder to do queries over different databases. This involves a workflow and process management, mainly dealing with confidentiality agreements. The supplied model needs be adapted to the CDM and integrated with the platform so that the user is able to navigate in the platform, starting in Layer 1.

5.4 Catalogue: A web framework to explore patient-level data

The Catalogue is an online application that enables biomedical data custodians to publish information about their databases, and researchers to browse, search, query and submit requests for databases they are interested in. Its architecture is extensible allowing other developer and communities to add their own extensions. The implementation is based on the proposed architecture and it supports both layer 1 and layer 2, being compliant and ready to adapt third party components of layer 3 and layer 4.

5.4.1 Software architecture

The Catalogue is built in three-tier software architecture (Figure 5.10). The top layer is open for interactions with both end-users, through the user interface, and third party applications, through direct data exchanges with the Catalogue Web services.

At this level, four main modules were developed: Browse Catalogue, Fingerprint Template, Administration management and API Web services.

The idea of the Browse Catalogue module was to create a workspace where the user is able to browse the fingerprints and also search for researchers' queries. Furthermore, the system allows the user to query in the Catalogue search engine and also compare results,

e.g., compare two databases and return the similarity for each field in the fingerprint. Moreover, third-party components should be easily integrated (we will describe this part of the architecture in section 5.5).

Fingerprint Templates is a module to create and edit the templates. Furthermore, an Excel template was created to allow specific types of databases to easily define new fingerprint templates, so they can be effortlessly imported into the system. In addition, data custodians can submit information about their database to the Catalogue. After the database fingerprint is submitted to the system, it should be searchable in the Catalogue.

The Administration Management module is available to control user registrations, to and to manage accounts and roles in the system.

The Web Services API provides a set of programmatic endpoints that can be consulted by third party applications and by the Catalogue presentation layer. The main idea is that other applications can send data to the Catalogue, in the format of key-value pairs, containing, for instance, population characteristics. Moreover, it is a simple mechanism to dynamically add metadata information in each database.

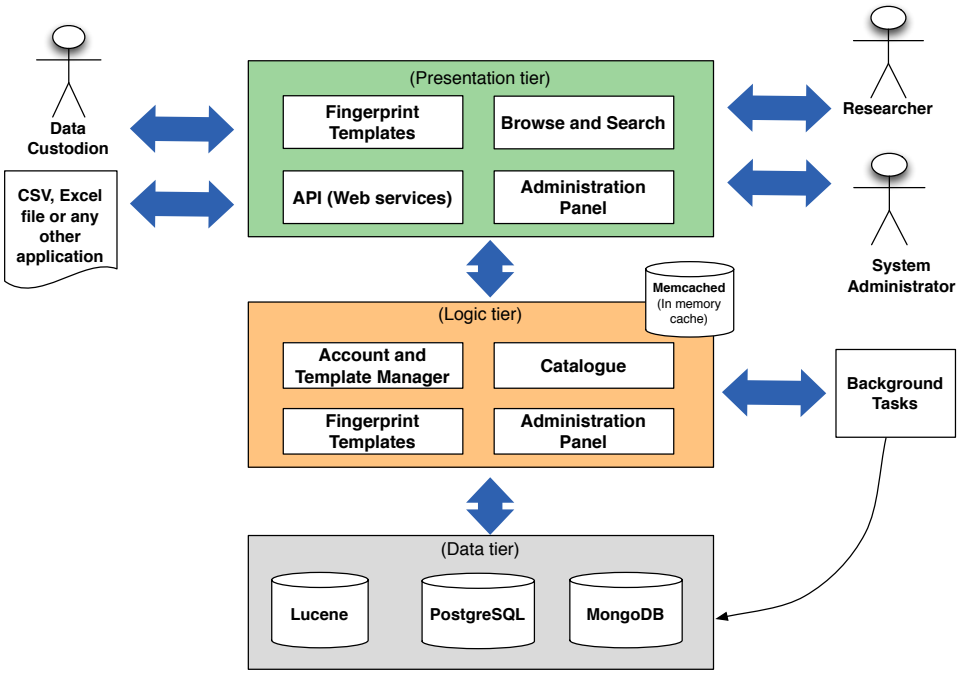

Figure 5.10: Catalogue - Software architecture

The logic layer contains the models that determine the Catalogue's entities, such as the users with different profiles and permissions, the definition of the template schemas, fingerprint instances and many others. The business logic is also defined in this layer. Finally, the data layer is where the information is stored. For example, the information regarding the Catalogue templates and dynamic groups of questions, as well as users

accounts, are stored in a PostgreSQL database. Furthermore, each database is also indexed in an Apache Solr instance. This open source search engine allows full-text search and is used in large-scale information retrieval projects. Moreover, specific aggregated views of the database, such as population characteristics, are uploaded in a MongoDB instance, which allows higher flexibility in the stored values, while maintaining good filtering features over stored data.

5.4.2 Software technologies

The Catalogue was developed in Python, using Django[1], a framework that encourages rapid development and clean programmatic design. However, a considerable part of the development was made in HTML5, CSS and JavaScript, namely the interface and the end-user interaction. Furthermore, in order to improve the web design quality, we have adopted the Bootstrap[2] framework, a front-end framework for web development.

Due to the high number of tasks that the Catalogue is subject to, we realized that several processes could slow down the system, when dealing with numerous concurrent interactions. Due to this issue, we adopted several strategies to improve its performance. One of those is to have a cache based on memcached[3] system. Several tasks, such as indexing the fingerprint in Solr, take too much time, making users wait for the operation to complete. All these heavy tasks are added to a queue message (Rabbit MQ[4]) and the Celery backend executes them in background.

5.4.3 Software design

Dynamic Fingerprint Schemas

To integrate *m* types of databases from different healthcare units, we created a system that supports *m* types of fingerprint schemas. Taking into account that the broader concept of fingerprint schema is a set of aggregated data about a database, it could be translated to questionnaires or aggregated data that could be imported, for instance, in TSV. The key idea behind the fingerprint schemas is that we do not know what kind of questions or data will be introduced. Thus, our implementation needs to be flexible and not bind to a set of questions.

Therefore, we developed a dynamic questionnaire with several groups of questions, some with dependencies between them (Figure 5.11). The *Questionnaire* is actually the type of database that will be fingerprinted. Each questionnaire should have groups of questions, which we named *QuestionSet.* Then, each *QuestionSet* will have a group of Question that

[1] https://www.djangoproject.com
[2] http://getbootstrap.com
[3] www.memcached.org
[4] www.rabbitmq.org

can be of any type. The implementation of the answer components is very flexible and can be extended.

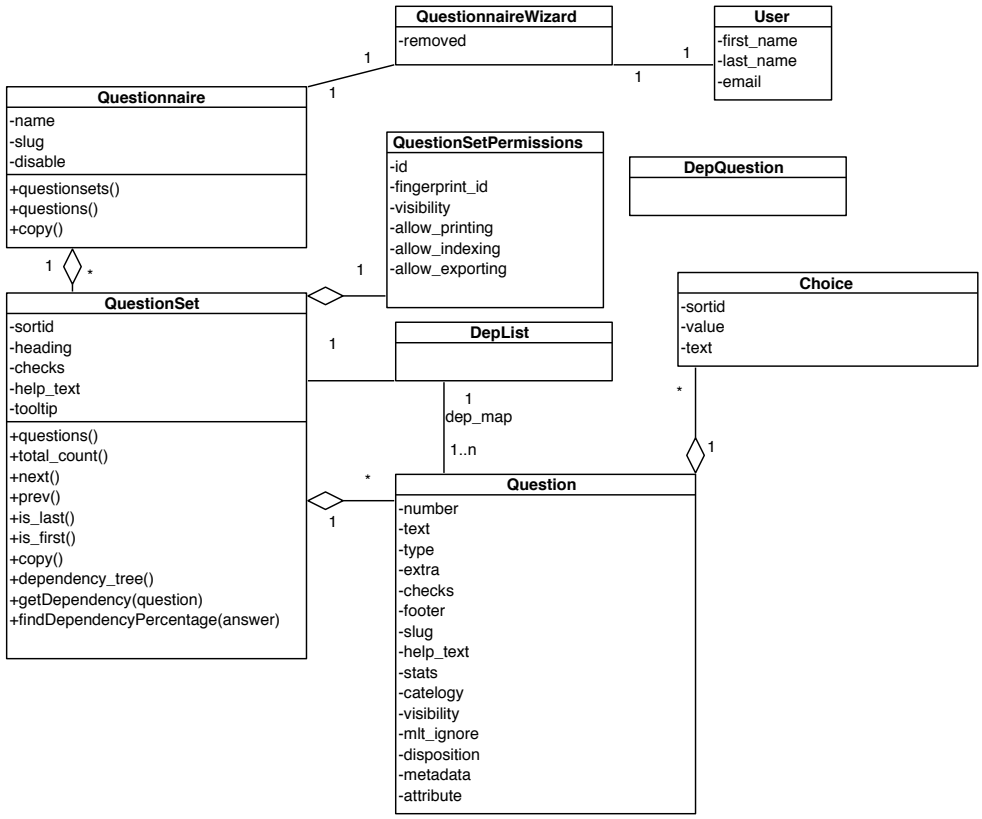

Figure 5.11: Class diagram questionnaire

Each fingerprint schema contains several groups of questions. Within these, each question can be individually defined, including the answer type (for instance, multiple-choice, plain-text, yes or no, and many others). The administrators can change each catalogue entry in the administration panel and the application will be automatically updated. The fingerprint schema can be developed using Excel spreadsheets to easily reach the end-users. We developed an import service to automatically load their questionnaires. Table 5.1 lists the supported set of components.

Fingerprint

The *Fingerprint* is a particular instance of a fingerprint schema and it represents a set of aggregated data that characterizes, generically, a biomedical database (Figure 5.12). In our particular model, to support the answers of the *Questionnaire,* we rely on the *Answer* entity. Moreover, since the information of the database is very sensitive, it was our decision to keep tracking the changes that are made and safeguard all the answers' history in the *AnswerChange* entity.

Table 5.1: Fingerprint data types.

Component	Visual render
open	Open Answer, single line [input]
open-button	Open Answer, single line [input] with a button to validate
open-textfield	Open Answer, multi-line [textarea]
choice-yesno	Yes/No Choice [radio]
choice-yesnocomment	Yes/No Choice with optional comment [radio, input]
choice-yesnodontknow	Yes/No/Don't know Choice [radio]
comment	Comment Only
choice	Choice [radio]
choice-freeform	Choice with a freeform option [radio]
choice-multiple	Multiple-Choice, Multiple-Answers [checkbox]
choice-multiple-freeform	Multiple-Choice, Multiple-Answers, plus freeform [checkbox, input]
publication	Publication
datepicker	Date choice

The fingerprint can also integrate other information, such as population characteristics, literature review, i.e. articles published about the databases, documents, or even other studies exported by third-party applications.

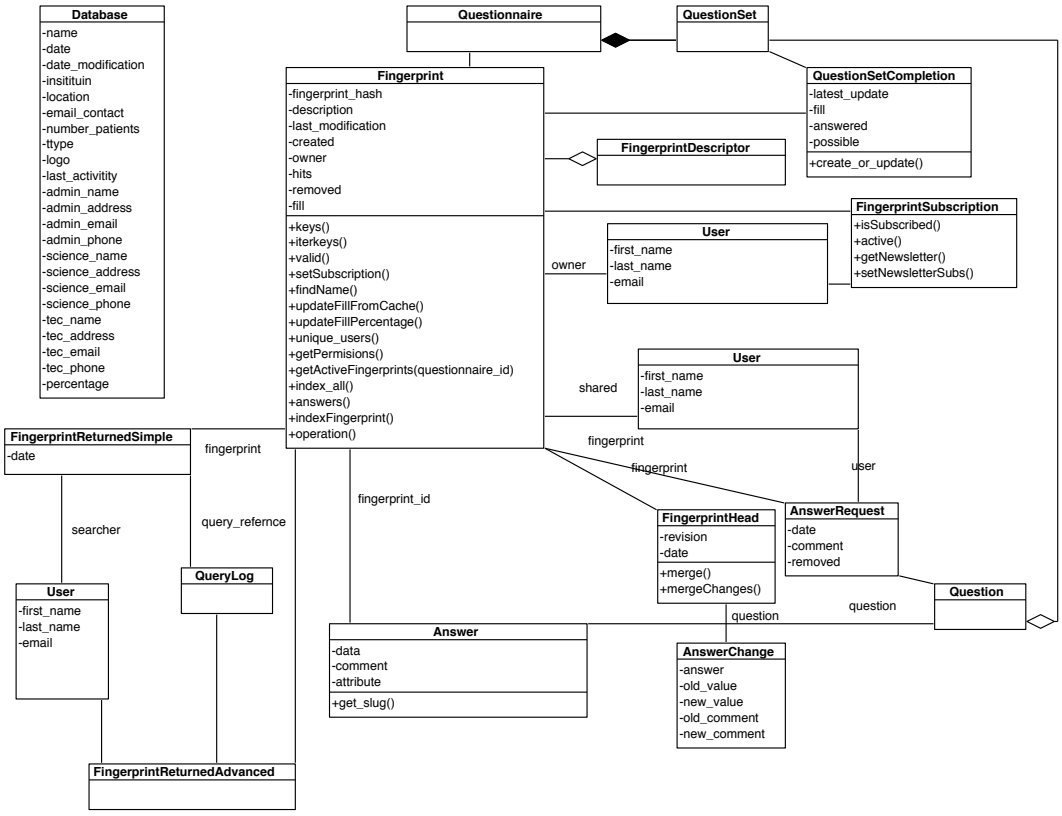

Figure 5.12: Fingerprint Class diagram

In the context of fingerprinting databases, we also developed a *publication* type of question, as mentioned in Table 5.1. The correspondent web widget allows fetching the

publication's information, using a Pubmed identifier, which is the identifier of one of the most popular databases in the biomedical domain. It does not only fetch the title and metadata about the publication, but also the abstract. Thereafter, the abstract is annotated using Becas [199], a web application service that provides biomedical concept identification. With this annotation, the user will have a better notion of the concepts identified in the database publication and will be linked to other relevant knowledge resources.

Search capabilities

One of the catalogue's main features is to allow searching through the database fingerprints. On the one hand, the database owner can see his personal databases. On the other hand, all users can have access to their databases of interest (according to their profile). Moreover, the users, for instance researchers, are able, not only to see all databases, but also to search for specific terms, e.g. "cardiac diseases" or "diabetes".

The free text capability does not fulfill all user requirements. Thus, an advanced search was also developed. This feature allows filling, in the fingerprint schema, the fields that one wants to search for. Then, several field restrictions can be combined using Boolean logic and nested Boolean queries.

To support such backend, all the answers are indexed using Solr. Nevertheless, a retrieve model has to be setup to score, sort and improve the quality of results. We followed the standards information retrieval rules, but also adapted the index engine for our particular model. For instance, in situations with a typical multi-choice answer such as "Do you collect CSF?", if the fingerprint is "yes", then, if a query by "CSF" is made, this fingerprint should be retrieved. This means that it is not only the answer that should be analyzed, but also the question, according to a particular answer.

The developed backend was also used to give suggestions in the free text query. The autocomplete suggestions are supported by another core (index schema) of Solr. For the implementation, we rely on a white space tokenizer with an edge filter n-gram between 1 and 25. This is particular useful to accelerate the process of giving accurate suggestions while the user is typing the text.

Fingerprint comparison

During the gathering of end-users requirements, and the development of the Catalogue, we realized that it would be important to have a comparison feature. Thus, we implemented the comparison between database fingerprints using a similarity metric based on Levenshtein distance (to compare pairs of databases).

Population Characteristics

One of the important details about each database is a global statistical characterization, which is named as population characteristics. These features include, for instance, the number of patients per age, the average time of follow-up and many other numerical data.

To support this requirement, we defined a default schema that allows extracting population data from databases. The correspondent web application provides a generic and dynamic charts representation, with tabular information, where the user can export data as CSV/TSV. Moreover, the framework is flexible and it is possible to define how the information is visualized in each database. These dynamic settings are stored in the server and then supplied to the client side (browser) - see Figure 5.13. Thus, the settings file will describe how to represent and render the information.

Figure 5.13: Dynamic TSV loader charts

In the Catalogue, we developed a strategy to have *Documents* associated with the *Fingerprint* (Figure 5.14). Thus, we created services to load the configuration settings stored in files or database, through the *ConfCharts*. It followed the structure of *SetCharts,* containing several *Charts*. Each chart contains the axis that is described in *Axis* (for x and y) with several filters and operations that can be performed. One of the important features is the possibility to filter taking into account that databases are usually stratified by age groups, years, sex, and several other parameters. Moreover, it is also extremely important to define the scale, for instance, the number of bins that should be represented, (e.g. 1 year, 5 years and 10 years).

The population characteristics are uploaded into MongoDB, which allows the application to scale for large aggregated results. This database is highly flexible and allows storing dynamic documents of which we do not know the exact number of columns. Moreover, they supply already filter and aggregation functions that could not be possible in any other relational system.

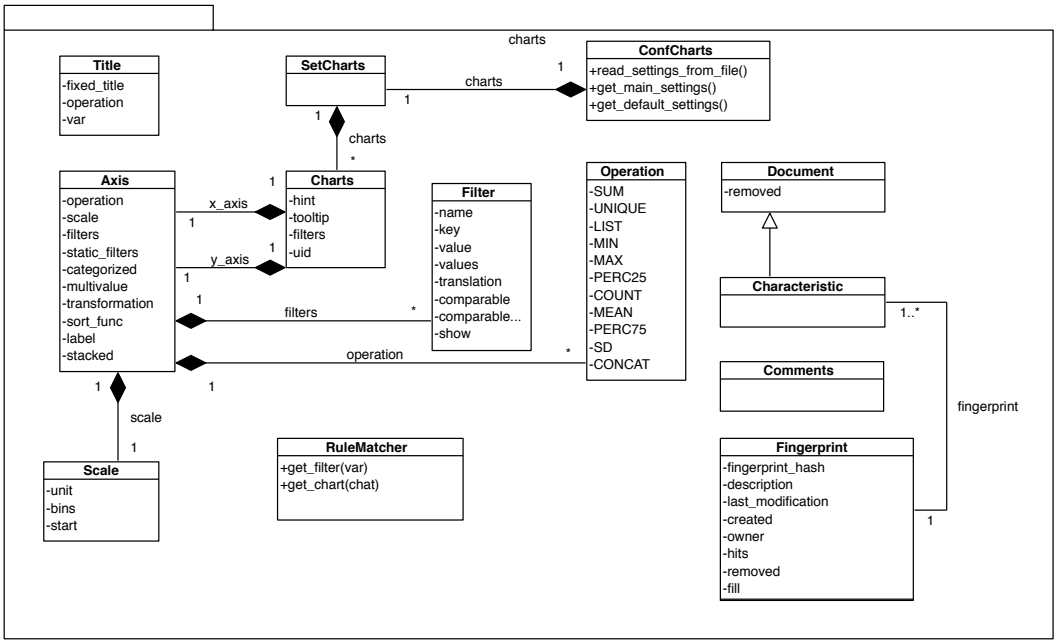

Figure 5.14: Population characteristics: class diagram

Achilles [1] is another module able to extract the population characteristics, as previously mentioned. Although this tool already allowed visualizing the data, it was mainly focused on the information that is stored in a folder. Hence, we developed several contributions to the AchillesWeb [2] open source project to permit data representation from any web endpoint. This module was also integrated in the Catalogue, as a plugin (following the architecture described in section 5.5).

Data aggregator

As already referred, the fingerprint can have a set of answers related with the Fingerprint Schema. Moreover, we designed a tool to automatically upload aggregated information generated in external applications (e.g. Jerboa and Achilles). But these two distinct views of databases also led us to create a new feature, combining data from the two sources. For instance, if it is required the number of patients, over the years, and per country, the necessary information has to be gathered from two different sources: fingerprint answers and population characteristic module *(*Figure 5.16*)*.

Similar to the population characteristics, this data aggregator also supports different fields, which may be defined in the configurations files. To support this dynamic model, fields are mapped in *Aggregation* and *AggregationField (*Figure 5.16*)*. For instance, most part of the database has information about location and the number of active population across the years, stratified by Male and Female. The information coming from two distinct sources could be aggregated and generate new information. For instance, if we

[1] https://github.com/OHDSI/Achilles
[2] https://github.com/OHDSI/AchillesWeb

consider a region level and we have fingerprints of the databases of all regions from one country, we will be able to know exactly how many patients are per region, per country, and across the years. Furthermore, it is also possible to stratify this population by particular disorders, which would not be easily achievable without this aggregation module. The aggregation process is an asynchronous task executed in background through Celery, to avoid real time problems (Figure 5.15).

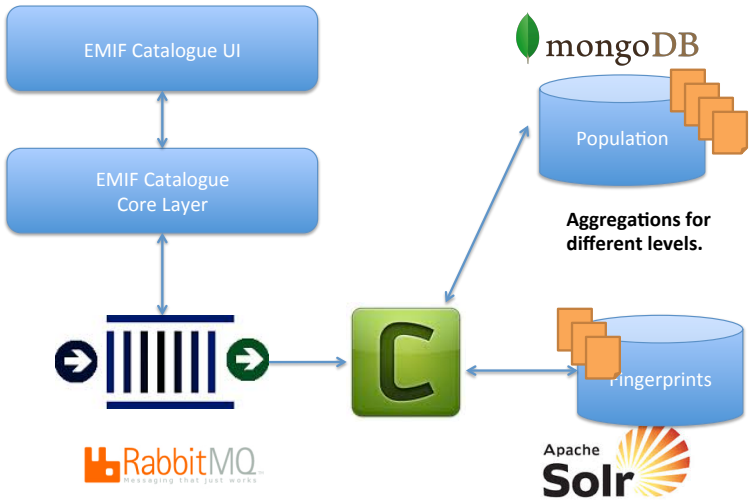

Figure 5.15: Data aggregation components using Celery at the core.

The data aggregation module needs to gather information from, at least, MongoDB and relational databases (answers from questionnaire). Then, it applies operations such as sum, counts and average, depending on the pre-defined aggregation settings.

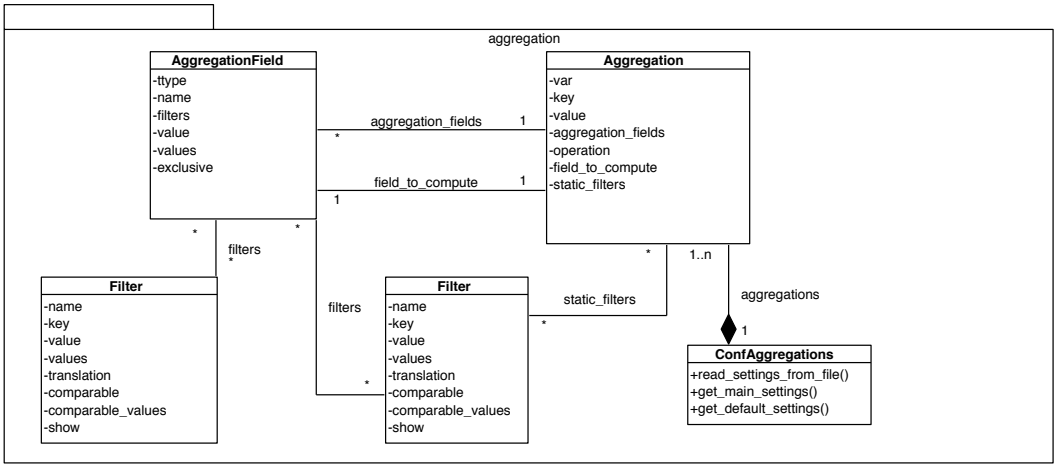

Figure 5.16: Data aggregation class diagram

5.4.4 Web Service API

The Catalogue is an application that only collects aggregated data and high-level information. Other tools such as Achilles, Jerboa or tranSMART allow collecting more

information, but they will only be available in the Data Custodians area. Nevertheless, some outputs generated by these, or other tools, might be associated with each fingerprint. We created a Web Service API that will be able to receive this information and include it as extra information in the fingerprint area (Figure 5.17).

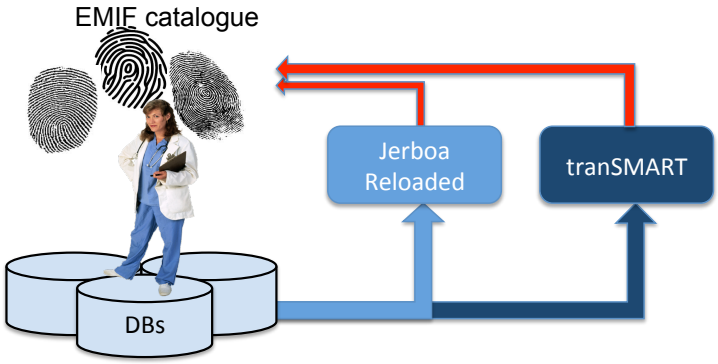

Figure 5.17: Example of interaction of Catalogue Web Service with third party applications.

The Web Service API is developed over a RESTful API that contains the following list of services:

- Insert and Edit values.
- Get the fingerprint data.

A double key schema allows controlling the access to this API. To use a web service, third-party applications need to know two distinct tokens: 1) the user token and 2) the database key. The access to the information is granted through the combination of both keys, being then possible to send extra data to the specific fingerprint IDs.

5.5 A microkernel architecture for software development

As we realized in the state of the art, there are already many tools that are important to integrate and extract relevant information from the databases, and that can fulfill some of the layers presented in our proposal (section 5.3). However, they are often isolated applications and difficult to integrate in third-party applications. The Catalogue also needs to support third-party components and, as an open source solution, allow others to extend functionalities, without knowing all the code base of the platform. To accomplish this ambition, we developed a microkernel architecture to allow the application to be easily extended in different ways, mainly in the client-side perspective. We granted the capability to support three plugin types, available to third-party developers:

- Global plugins: they may provide general views over all the databases, for a given user or for all users. Once added to the Catalogue, these plugins will be available on the user dashboard. Examples of global plugins are a tag cloud with the summary content of all databases, or worldwide population summaries. It

should provide information that takes into consideration all the databases, and aggregate data, displaying them to the user.

- Database-related plugins: this type of plugin provides only database-specific views, and will be available at the database view. Examples of database-related plugins are the literature tab, or the extra information tab coming from the web services API.
- Third-party applications: They are full web applications that are linked to the system, through the navigation menu (eventually linked with a SSO). They usually provide a complete different functionality, although they may share some features with the platform, like the authentication. The main goal of these plugins is to integrate their application features in the platform environment. Good examples of these applications are the workflow-management system and the EHR4CR platform.
- Full-fledged plugins: they can have their own page in the navigation menu, similar to third-party applications, but they are really Catalogue extensions and may also provide data integration with the system. The idea behind this type of plugin is allowing the development of completely new applications related to the Platform.

5.5.1 Plugin Lifecycle: Client-side

The plugin system is based on a re-rendering system in browser. After the initial re-rendering, each time an event occurs (that should be caught through the usual JavaScript event listeners), the plugin should update the content using *self.html()* or *self.append()*, and when it deems ready, execute *self.refresh()* to update the visualization. A diagram representing this interaction is shown in Figure 5.18.

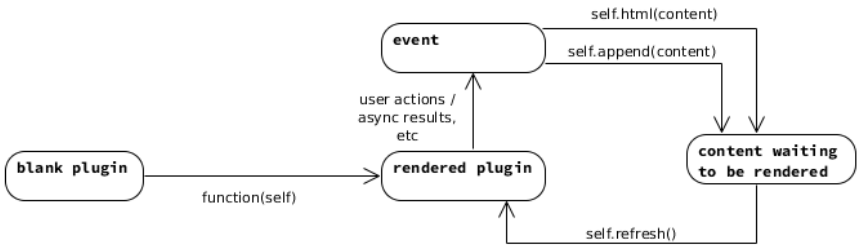

Figure 5.18: Plugin Lifecycle

5.5.2 Development Lifecycle

To be sustainable, a plugin development must follow a proper lifecycle. The first step is to obtain the administrator's approval to develop plugins. After that, the user can create

any number of plugins. Each plugin can have a series of versions, although users can develop and live-preview their plugin versions during development. Whenever a plugin version is deemed ready to production, it must be submitted for administrator's approval to become available. Any further changes to an already approved version will remove the approval status, and the plugin will have to be submitted again.

The plugin version available for the other users is always the biggest-numbered approved version (considered the latest stable version). A diagram of this development cycle is shown in Figure 5.19.

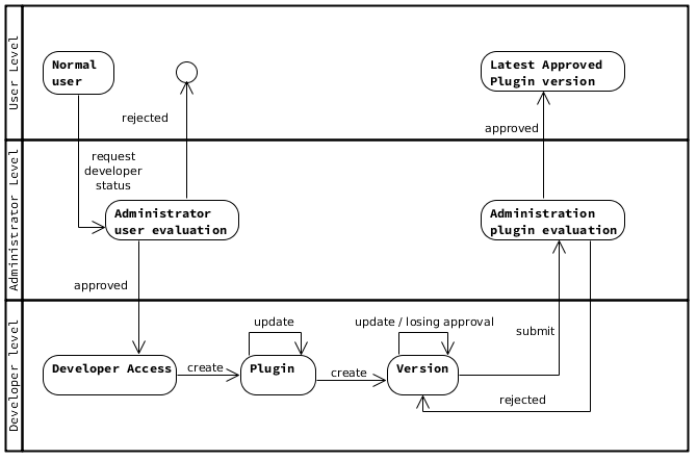

Figure 5.19: Development cycle

5.6 Results and discussion

In this section, we will present the developed application, the user experience and the implementation in the EMIF Platform [21]. The presented implementation proved to have strong impact in the biomedical researchers both in academia and pharmaceutical companies. It was already possible to summarize several databases types, with several medical institutions around the world. The developed software is free and open source available in https://github.com/bioinformatics-ua/catalogue.

5.6.1 Features and user experience

Catalogue only allows registered users to access the fingerprints information. The first impact of the user with the Catalogue is shown in Figure 5.20.

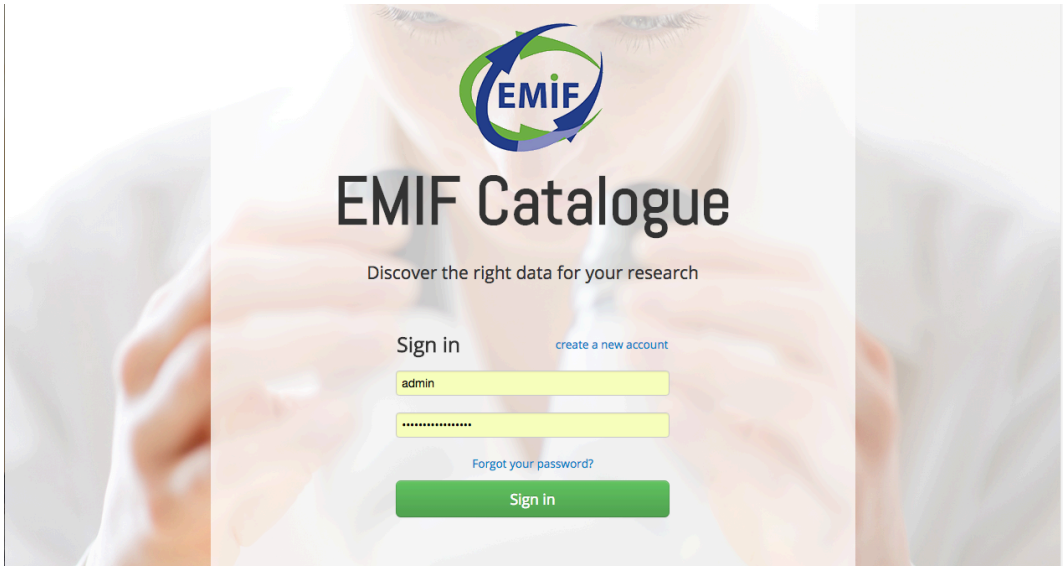

Figure 5.20: Landing page - registration and login are possible at this step.

After the administration's approval, a new email is sent to the user allowing him to login with his credentials. Then, the user will be able to access his profiles and interests in the right hand top corner (Figure 5.21).

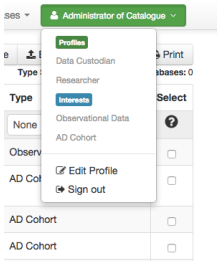

Figure 5.21: User roles and interests

For the development of the Catalogue, we had an extensive list of user requirements to take into account and distinct user groups performed several evaluation sessions along the different releases. One major request was to have a central place with global statistics and an overview of the last actions performed in the overall Catalogue. Therefore, we developed a Dashboard to answer this request, consisting on a centralized place where users can consult the last updates in their subscribed fingerprints (Figure 5.22). The dashboard is dynamic and each user can customize it according to his personal preferences.

Figure 5.23 shows the general questionnaire interface (in this case, Alzheimer Cohort). For each question, the user can add comments, only visible if selected. Due to the large amount of questions in the fingerprint template, it is also possible to collapse and expand all contents. Moreover, a green icon that indicates which questions are already filled was added, increasing the Catalogue usability. To help Database Custodians to update the fingerprint, it is also possible to show only the non-filled questions.

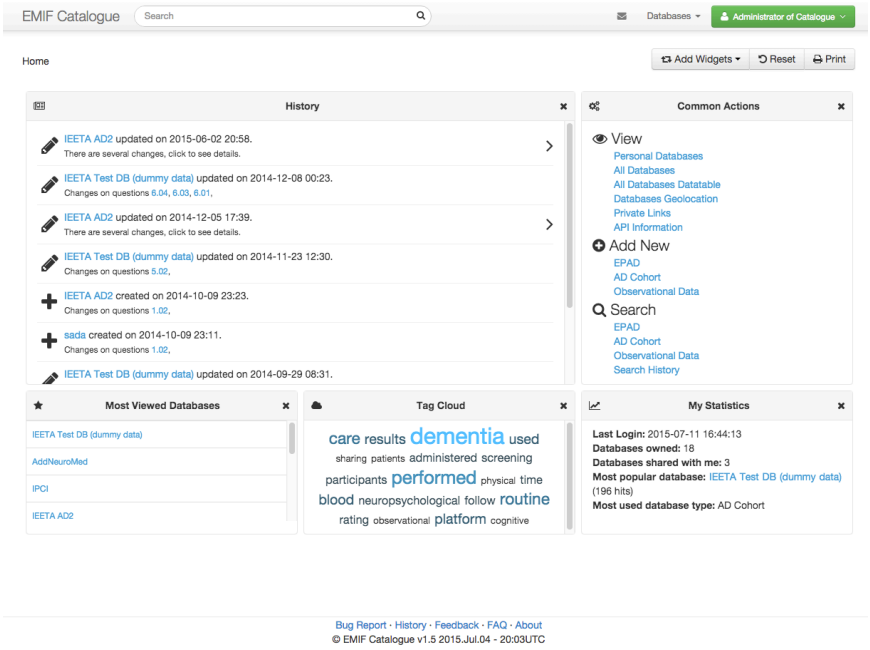

Figure 5.22: Catalogue dashboard: a summary view

As already mentioned, the fingerprint templates' length can increase, creating opportunities to enrich the Catalogue with new dynamic widgets, apart from the typical ones, such as multiple choices, yes/no questions or open answers. For instance, Figure 5.23 shows a multiple-choice question, where the data holder can also add information regarding each option. Moreover, questions can have dependencies between them. For instance, a set of question will appear depending on previous answers to the question to which it is dependent. On the right side of the question, there is an icon to allow adding comments to each answer. This can be useful in case of closed answers and when the user wants to add other relevant information. On the left side, all the subgroups of questions (previous referred as *QuestionSets)* are presented. It is possible to easily move from one subset of question to another, having access to the filling percentage of each set. Moreover, in the top right side it is indicated the "Permissions" to which the database owner has access. In this panel, it is possible to specify if the questions are public or private and if the user will be allowed to either export, or print them.

While fingerprinting the database, the researcher might as well be interested in scientific publications based on that database. Thus, a specific component automatically collects publication information based only on the Pubmed ID (Figure 5.28). This feature is particularly important because it guarantees publications information accuracy, and a standardized format for visualization.

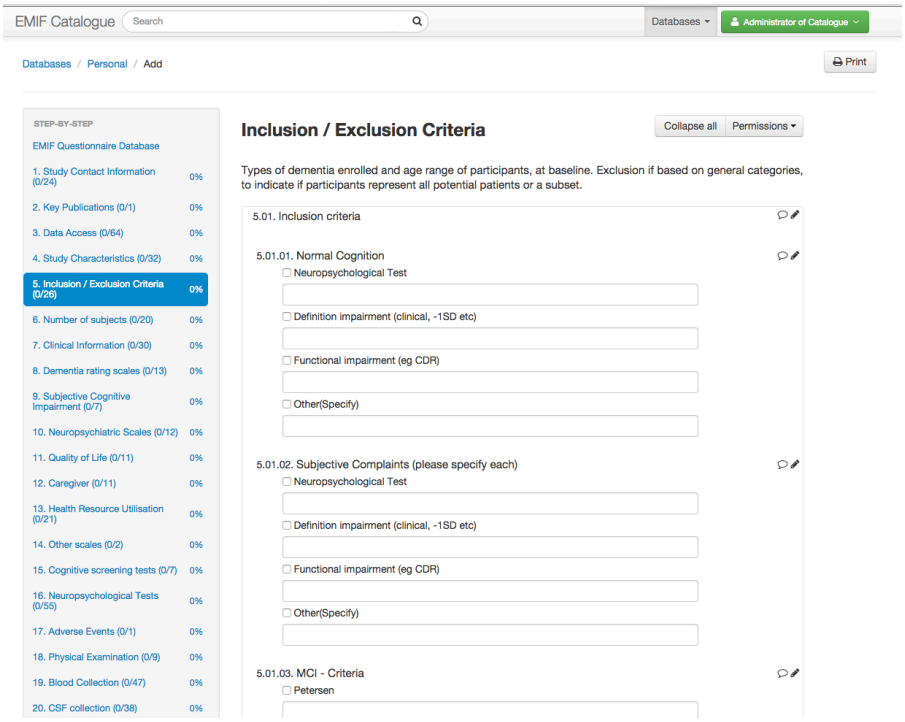

Figure 5.23: Example of a fingerprint schema design

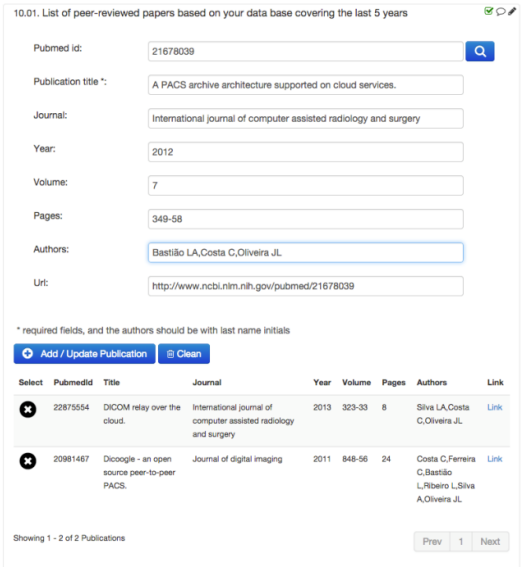

Figure 5.24: Publication widget for the questionnaire

Some fingerprint schemas may have hundreds of questions and answering all of them can sometimes be an excessive time consuming process. This can constitute an important issue, not only due to the required amount of time to fill in all the questions, but also because the user may not have all the necessary answers. Furthermore, a researcher can search for a fingerprint with potential interest and realize that more information is needed. Thus, it is possible to send an "Answer request" to the database owner, who will receive a notification about which questions the researcher would like to see filled in (Figure 5.25).

Therefore, the interaction between database custodians and clinical researchers is much larger and simplified.

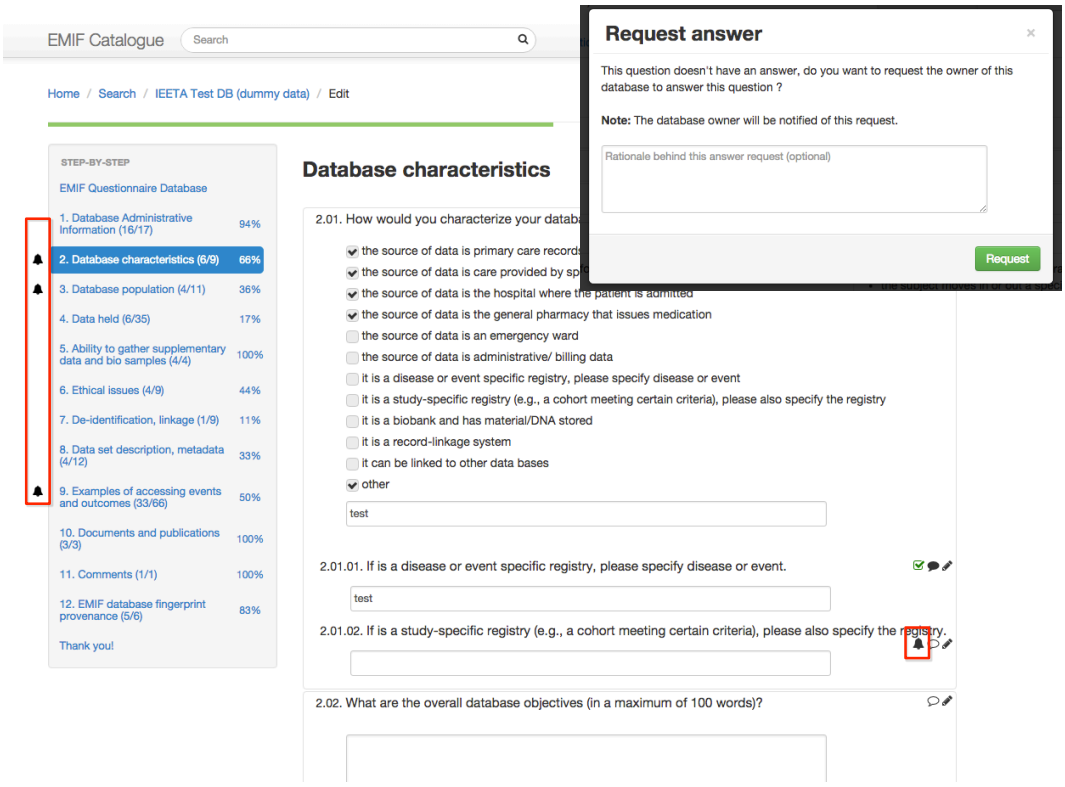

Figure 5.25: "Answer request" allows a user asking for particular questions.

The ownership of a database is highly important and, considering the fingerprint approach, there are several options available for the user. It is possible to share the ownership of the fingerprint, for instance, invite other users to edit the same fingerprint. This will allow a collaborative fingerprint edition (Figure 5.26). Moreover, for the non-registered users or people that only have access to a particular set of databases, the database owner has the capability to create a link that grants the access to a fingerprint to anyone who has it (Figure 5.26).

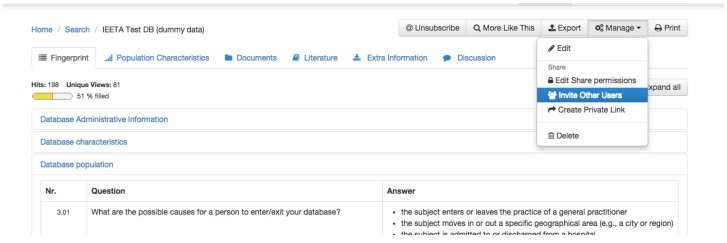

Figure 5.26: Database owner menu to manage the fingerprint. It is possible to edit, share, and invite other users to be database owners, as also to create a private link to be share with other non-registered users.

The private links have two control specifications: 1) expiration date and 2) number of views allowed. For instance, it is possible to share the same database with different persons, using different links, with distinct permissions (Figure 5.27). This possibility is

very useful for situations when, for instance, the database owner wants to share his database with users outside Catalogue.

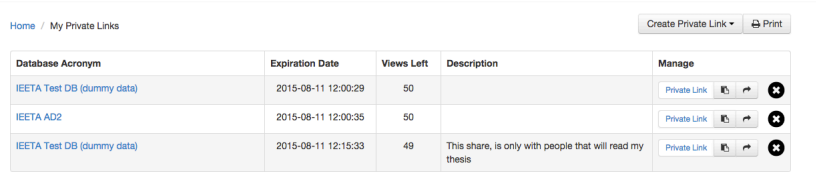

Figure 5.27: Management of private links: share with non-registered users

As already described in the previous section, one possible way to present the information is based on the Population Characteristics (Figure 5.28 and Figure 5.29). The configuration chart module allows defining which charts will appear and how they appear, e.g. as line, multi-bar, or stacked bar. Examples of chart types are, for instance, active patients, birthdates and many others. They will be listed on left hand side, as shown in Figure 5.28. Data filters are available at right hand side.

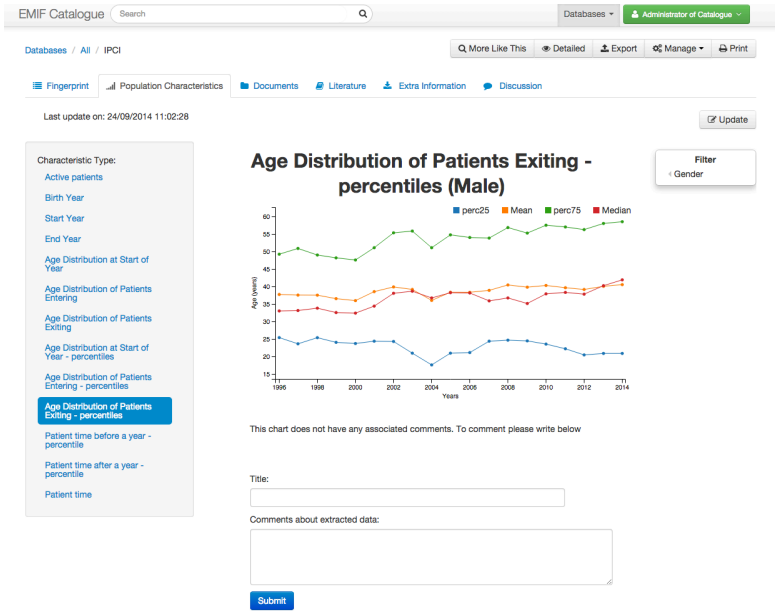

Figure 5.28: Population characteristics - Percentiles

5.6.2 A case study in the EMIF Platform

Users and Fingerprints

In the context of EMIF Platform, there is a need to integrate medical records, clinical and other omics information from different sources. Aiming to tackle this problem, we used the Catalogue to provide the first step towards answering the questions from the clinical researchers, in terms of knowing and comparing data sources capabilities, was created. The current version of the EMIF Catalogue is accessible at http://bioinformatics.ua.pt/emif, only for registered users until a full governance model is

set up in the project, which is expected to be completed by the end of 2017. Two distinct communities are already using the EMIF Catalogue:

- Observational Data sources: the fingerprint schema has 12 groups of questions, with 212 questions.
- Alzheimer Cohorts: the fingerprint schema has 27 groups of questions, with 558 questions.

The EMIF Catalogue has already collected 59 fingerprints of different Alzheimer's Cohort databases and 15 from Electronic Healthcare records (EHR) data sources, corresponding to a total around of 30 million patients. Overall, the Catalogue has more than one hundred registered users, including both database owners and researchers.

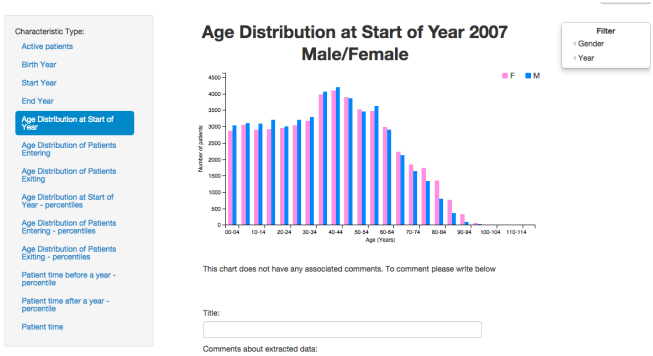

Figure 5.29: Population Characteristics - example of Age distribution

Usage statistics

Given the relevance of the user feedback, it is important to monitor the use of the system. To allow this, users' actions are being gathered, helping us to monitor catalogue usage and tracking actions for abnormal issues. Since its first release in April 2013, the number of users of the EMIF catalogue has been increasing (Figure 5.30).

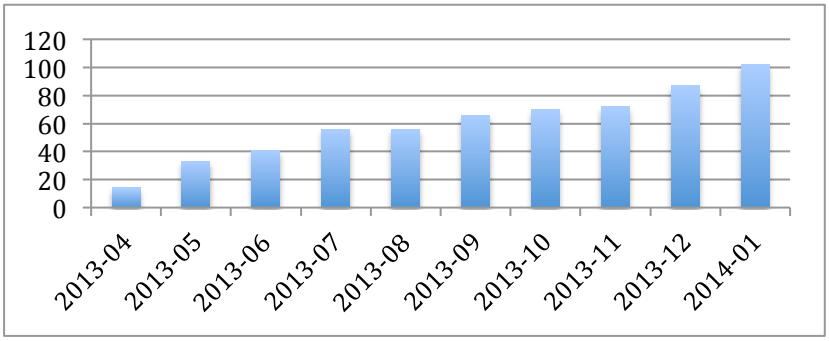

Figure 5.30: Number of registered users

As expected, the number of databases has been also increasing along the same period (Figure 5.31).

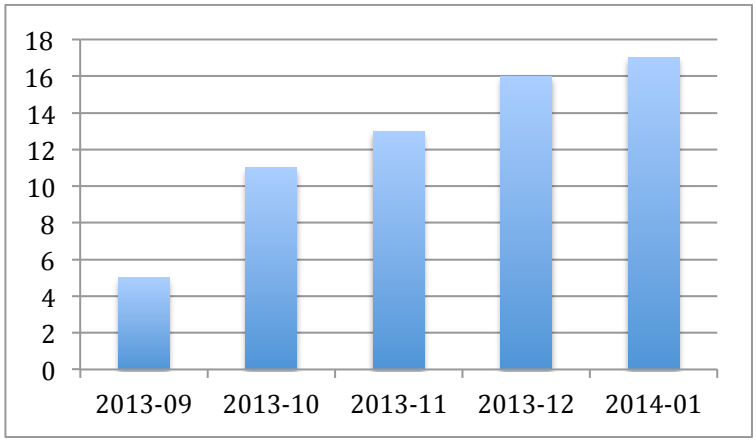

Figure 5.31: Number of databases over time

Figure 5.32 presents the queries performed by the EMIF community. Users have not only performed searches by free text, but also through advanced search. The history also helps the users to improve the navigation and consult past search history, which may be helpful to redefine and perform the same queries over the time, to verify possible updates.

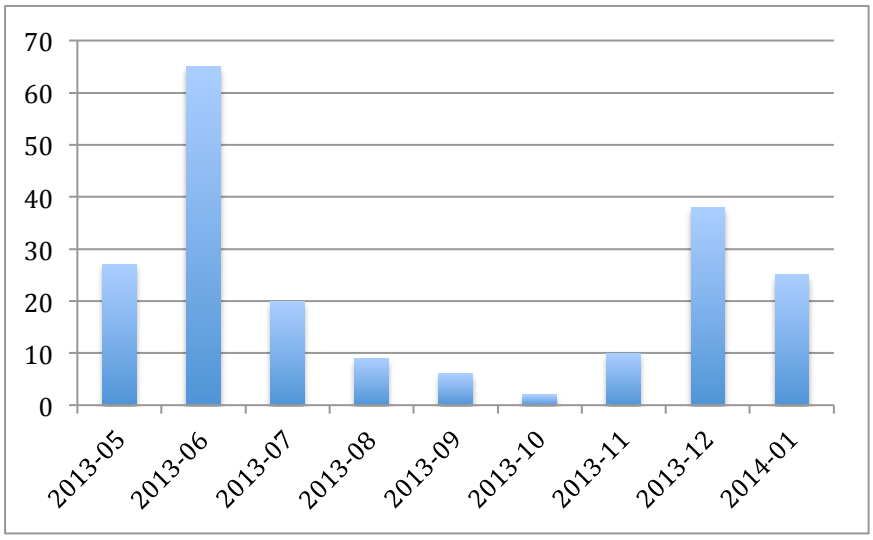

Figure 5.32: Number of the queries over time

Finally, Figure 5.33 shows the increasing number of visits to the EMIF Catalogue along time. It is possible to conclude that the Catalogue use is constant, without many blackout periods, except for August, which corresponds to the usual holiday period. Nevertheless, it is also important to highlight some peaks of major affluence, typically corresponding to the times when researchers need to extract results for their own research and/or reports.

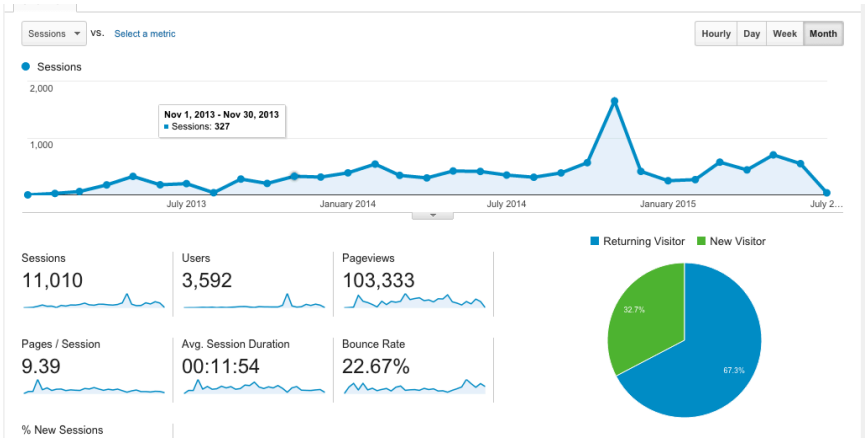

Figure 5.33: Users of Catalogue in the EMIF Platform

5.7 Final considerations

Biomedical researchers have been facing several difficulties to find databases or information that fulfill their requirements. In this chapter, we have presented a microkernel software architecture that is able to fingerprint any kind of healthcare and research databases, aiming at facilitating the selection of appropriate data sources that may help answering a particular research question. This is an open source project that can be followed in https://github.com/bioinformatics-ua/catalogue.

The Catalogue solution was developed as an approach to widespread the information about clinical and patient-level database, without exposing private data. The goal was achieved and the platform can also be enriched with more information, not only a set of the questions but also information about the population that can be extracted using open source tools or stratified manually using an Excel spreadsheet and imported in the Catalogue.

The system is already being used to collect fingerprints from a wide selection of cohort and EHR databases.

6 Conclusion and future directions

"Focus on where you want to go, instead of where you have been." - John M. Templeton, Worldwide Laws of Life

6.1 Conclusion

Biomedical data integration is a blooming research area that creates new challenges on several distinct topics of computer science and of biomedical informatics. The data being generated in medical institutions and in pharmaceuticals companies are increasing at an unprecedented pace, and its fully exploration is crucial for a better healthcare provisioning. Many repositories currently kept by medical institution have a huge potential for research. However, most of the times, they are generated for its primary use, then archived and rarely accessed again. The multiple biomedical databases that are being produced contain patients' history, treatment trajectories, drug-event associations, drug-drug interactions, rare cases, and many other evidences that can be highlighted when a broad analysis of these data is performed. Moreover, the statistical analysis may also improve institutions efficacy and productivity.

Data integration in the biomedical domain offers a new way of thinking about medicine research. Nevertheless, in order to successfully accomplish that goal, several issues have yet to be better investigated. In the course of this document those problems have been addressed, and it was possible to understand that the standards may help the process but do not solve everything. There are still missing regulations to easily create the channels to integrate and share information. Hence, researchers adopted new ways to live in this

chaos, where they assume that the databases will not be normalized and manual or semi-automatic processes will have to be applied.

In this doctorate, we proposed several software architectures to integrate biomedical data in several distinct scenarios. We have also created a platform that facilitates the exploration of this information across databases.

6.2 Outcomes

In the process of creating strategies to integrate and share databases across multiple hospital units we achieved several results.

The first hypothesis was about if a Cloud-based software architecture could solve the integration between different institutions. To cope with this, we proposed a solution to integrate medical imaging from *n* healthcare units. Thereafter, many problems have been raised, and the first one was which Cloud provider we should use. For this, we developed a common API for delivering services over multi-vendor Cloud resources framework to grant interoperability between different Cloud providers. A DICOM relay over the Cloud was the following proposed solution, based on this Cloud framework. Next, a system to federate multiple medical imaging repositories was implemented. Moreover, a centralized platform for geo-distributed PACS management was developed to ensure that the services are working continuously.

The second research hypothesis addressed data integration over *n* healthcare units. We proposed a federated strategy to inquire databases from a single and centralized platform in a scalable manner. This sensor-based architecture is a contribution to integrate multiple sources of information related to medical imaging departments. The architecture allows monitoring images workflow and it is fully extensible to add new data sources. Focusing in facilitating clinical research, we have also performed dose and productivity analysis from information collected in several hospitals. Furthermore, a normalization schema was developed over heterogeneous medical imaging measurements, to allow studying the impact of radiation dose.

Finally, the last research hypothesis dealt with the integration of multiple levels of data detail from multiple healthcare units. We proposed an architecture, supported by open source software, to summarize patient-level data across borders. The main idea of this contribution was to create a framework that is able to aggregate summarized data from disperse databases. Furthermore, this is currently the main integration component in several European projects.

While exploring new software architectures to improve the integration of biomedical databases, several new problems were raised and solved. All the proposed solutions have

been validated with thousands of patients and users, with a huge impact in real-life environments.

6.3 Future work

Along the methods, designed solutions and results that were addressed in this thesis, there are several research lines that can be explored in future work, such as:

- Semantic text annotators in all data sources: while during the doctorate we developed some strategies to gather information provided from many data sources (normalized and not normalized), there is still lot of work to do in terms of text annotators to automatic classify concepts and its ontologies in biomedical documents.
- Semantic image annotators: this is yet a less investigated area in which a potential research goal is the creation of classification and annotation methods that automatically match parts of body and disorders into a biomedical vocabulary (e.g. UMLS).
- Widen data sources integration: the extraction of clinical evidences from existing data will be as good as higher the amount of data that mining and knowledge extraction algorithms can rely upon. Many other data sources can be relevant to integrate, such as omics information, prescriptions, and images, which can be described by content-based image retrieval (CBIR) tools.
- Drill-down in a fully integrated way: as the developed work affect different levels of information, the process of accessing it should be possible in a more integrated way. For instance, researchers should be able to search for databases that fulfill their requirements, and issue a set of queries to be executed in diverse patient repositories.

7 References

[1] B. Peters and A. Sette, "Integrating epitope data into the emerging web of biomedical knowledge resources," *Nature Reviews Immunology,* vol. 7, pp. 485-490, 2007.

[2] U. Hahn, M. Romacker, and S. Schulz, "Creating knowledge repositories from biomedical reports: the MEDSYNDIKATE text mining system," 2002.

[3] A. M. Cohen and W. R. Hersh, "A survey of current work in biomedical text mining," *Briefings in bioinformatics,* vol. 6, pp. 57-71, 2005.

[4] M. Hu, W. Pavlicek, P. T. Liu, M. Zhang, S. G. Langer, S. Wang*, et al.*, "Informatics in Radiology: Efficiency Metrics for Imaging Device Productivity," *Radiographics,* vol. 31, pp. 603-616, 2011.

[5] M. Santos, L. Bastião, C. Costa, A. Silva, and N. Rocha, "DICOM and Clinical Data Mining in a Small Hospital PACS: A Pilot Study," *ENTERprise Information Systems,* pp. 254-263, 2011.

[6] L. Zon, "Translational Research: The Path for Bringing Discovery to Patients," *Cell stem cell,* vol. 14, pp. 146-148, 2014.

[7] P. M. Coloma, M. J. Schuemie, G. Trifirò, R. Gini, R. Herings, J. Hippisley‐Cox*, et al.*, "Combining electronic healthcare databases in Europe to allow for large‐scale drug safety monitoring: the EU‐ADR Project," *Pharmacoepidemiology and drug safety,* vol. 20, pp. 1-11, 2011.

[8] C. Daniel, E. Albuisson, T. Dart, P. Avillach, M. Cuggia, and Y. Guo, "Translational Bioinformatics and Clinical Research Informatics," in *Medical Informatics, e-Health*, A. Venot, A. Burgun, and C. Quantin, Eds., ed: Springer Paris, 2014, pp. 429-461.

[9] M. Gottwald, "How Can the Innovative Medicines Initiative Help to Make Medicines Development More Efficient?," in *Re-Engineering Clinical Trials: Best Practices for Streamlining the Development Process*, ed, 2014, p. 55.

[10] S. Marceglia, P. Fontelo, and M. J. Ackerman, "Transforming consumer health informatics: connecting CHI applications to the health-IT ecosystem," *Journal of the American Medical Informatics Association,* p. ocu030, 2015.

[11] D. B. Fridsma, J. Evans, S. Hastak, and C. N. Mead, "The BRIDG project: a technical report," *Journal of the American Medical Informatics Association,* vol. 15, pp. 130-137, 2008.

[12] D. P. Hansen, C. Pang, and A. Maeder, "HDI: integrating health data and tools," *Soft Computing-A Fusion of Foundations, Methodologies and Applications,* vol. 11, pp. 361-367, 2007.

[13] S. Szalma, V. Koka, T. Khasanova, and E. D. Perakslis, "Effective knowledge management in translational medicine," *Journal of translational medicine,* vol. 8, p. 68, 2010.

[14] A. E. Cuellar and P. J. Gertler, "Strategic integration of hospitals and physicians," *Journal of Health Economics,* vol. 25, pp. 1-28, 2006.

[15] L. A. Bastião Silva, C. Costa, and J. L. Oliveira, "A common API for delivering services over multi-vendor cloud resources," *Journal of Systems and Software,* vol. 86, pp. 2309-2317, 2013.

[16] L. A. B. Silva, C. Costa, and J. L. Oliveira, "DICOM relay over the cloud," *International Journal of Computer Assisted Radiology and Surgery,* pp. 1-11, 2012.

[17] L. A. B. Silva, R. Pinho, L. S. Ribeiro, C. Costa, and J. L. Oliveira, "A Centralized Platform for Geo-Distributed PACS Management," *Journal of digital imaging,* vol. 27, pp. 165-173, 2014.

[18] L. A. B. Silva, S. Campos, C. Costa, and J. L. Oliveira, "Sensor-Based Architecture for Medical Imaging Workflow Analysis," *Journal of Medical Systems,* vol. 38, pp. 1-11, 2014.

[19] L. A. B. Silva, L. S. Ribeiro, M. Santos, N. Neves, D. Francisco, C. Costa*, et al.*, "Normalizing Heterogeneous Medical Imaging Data to Measure the Impact of Radiation Dose," *Journal of digital imaging,* pp. 1-13, 2015.

[20] L. A. B. Silva, C. Días, J. v. d. Lei, and J. L. Oliveira, "Architecture to summarize patient-level data across borders and countries," in *MedInfo*, São Paulo, Brazil, 2015.

[21] EMIF. *2015*. Available: http://www.emif.eu/

[22] L. A. B. Silva, C. Costa, and J. L. Oliveira, "Strengths and weaknesses regarding support of medical images repositories over the cloud," *Imaging Management,* vol. 11 (5), pp. 14-15., 2012.

[23] L. A. B. Silva, C. Costa, and J. L. Oliveira, "Data Ownership & Protection Issues - Strengths and Weaknesses of Using Cloud Computing," *Healthcare IT Management,* vol. 7, pp. 20-22, 2012.

[24] L. A. B. Silva, C. Costa, and J. L. Oliveira, "Migrating PACS to the cloud – advantages and drawbacks," *International Hospital and Equipment,* vol. 38, pp. 16-18, 2013.

[25] L. A. B. Silva, C. Costa, and J. L. Oliveira, "Integrity And The Personal Health Record - How to store, access and explore integrated health records and ensure their privacy," *Health Managemetn,* vol. 15, pp. 74-75, 2014.

[26] C. Costa, C. Ferreira, L. Bastião, L. Ribeiro, A. Silva, and J. Oliveira, "Dicoogle - an Open Source Peer-to-Peer PACS," *Journal of Digital Imaging,* pp. 1-9, 2010.

[27] L. A. B. Silva, C. Costa, and J. L. Oliveira, "Semantic Search over DICOM Repositories," in *Healthcare Informatics (ICHI), 2014 IEEE International Conference on*, 2014, pp. 238-246.

[28] I. Iakovidis, "Towards personal health record: current situation, obstacles and trends in implementation of electronic healthcare record in Europe1," *International Journal of Medical Informatics,* vol. 52, pp. 105-115, 1998.

[29] P. C. Tang, J. S. Ash, D. W. Bates, J. M. Overhage, and D. Z. Sands, "Personal health records: definitions, benefits, and strategies for overcoming barriers to adoption," *Journal of the American Medical Informatics Association,* vol. 13, pp. 121-126, 2006.

[30] e-HPHRW Group, "Practice brief. The role of the personal health record in the EHR," *Journal of AHIMA/American Health Information Management Association,* vol. 76, p. 64A, 2005.

[31] A. Bakker and J. Mol, "Hospital information systems," *Effective health care,* vol. 1, pp. 215-223, 1983.

[32] B. I. Blum, "Hospital Information Systems," in *Clinical information systems*, ed: Springer, 1986, pp. 217-251.

[33] B. P. Bloomfield, R. Coombs, D. J. Cooper, and D. Rea, "Machines and manoeuvres: responsibility accounting and the construction of hospital information systems," *Accounting, Management and Information Technologies,* vol. 2, pp. 197-219, 1992.

[34] G. SMITH, "Introduction To Ris And Pacs," *PACS: a guide to the digital revolution,* p. 9, 2006.

[35] H. K. Huang, *PACS and imaging informatics: Basic Principles and Applications*, 2nd ed. New Jersey: Wiley & Blackwell, Hoboken, 2010.

[36] C. Costa, J. L. Oliveira, A. Silva, V. G. Ribeiro, and J. Ribeiro, "Design, development, exploitation and assessment of a Cardiology Web PACS," *Comput Methods Programs Biomed,* vol. 93, pp. 273-282, 2009.

[37] "Prepare for Disasters & Tackle Terabytes When Evaluating Medical Image Archiving," A Frost & Sullivan Healthcare Article. http://www.frost.com. (Available September 2015)

[38] M. Kanehisa and S. Goto, "KEGG: Kyoto Encyclopedia of Genes and Genomes," *Nucl. Acids Res.,* vol. 28, pp. 27-30, January 1, 2000 2000.

[39] M. Kanehisa, S. Goto, M. Hattori, K. F. Aoki-Kinoshita, M. Itoh, S. Kawashima*, et al.*, "From genomics to chemical genomics: new developments in KEGG," *Nucl. Acids Res.,* vol. 34, pp. D354-357, January 1, 2006 2006.

[40] G. Joshi-Tope, M. Gillespie, I. Vastrik, P. D'Eustachio, E. Schmidt, B. de Bono*, et al.*, "Reactome: a knowledgebase of biological pathways," *Nucleic acids research,* vol. 33, pp. D428-D432, 2005.

[41] C. F. Schaefer, K. Anthony, S. Krupa, J. Buchoff, M. Day, T. Hannay*, et al.*, "PID: the pathway interaction database," *Nucleic acids research,* vol. 37, pp. D674-D679, 2009.

[42] A. Bairoch, R. Apweiler, C. H. Wu, W. C. Barker, B. Boeckmann, S. Ferro*, et al.*, "The universal protein resource (UniProt)," *Nucleic acids research,* vol. 33, pp. D154-D159, 2005.

[43] N. J. Mulder, R. Apweiler, T. K. Attwood, A. Bairoch, A. Bateman, D. Binns*, et al.*, "New developments in the InterPro database," *Nucl. Acids Res.,* vol. 35, pp. D224-228, January 12, 2007 2007.

[44] S. T. Sherry, M.-H. Ward, M. Kholodov, J. Baker, L. Phan, E. M. Smigielski, *et al.*, "dbSNP: the NCBI database of genetic variation," *Nucl. Acids Res.,* vol. 29, pp. 308-311, January 1, 2001 2001.

[45] A. Hamosh, A. F. Scott, J. S. Amberger, C. A. Bocchini, and V. A. McKusick, "Online Mendelian Inheritance in Man (OMIM), a knowledgebase of human genes and genetic disorders," *Nucl. Acids Res.,* vol. 33, pp. D514-517, January 1, 2005 2005.

[46] C. E. Lipscomb, "Medical Subject Headings (MeSH)," *Bull Med Libr Assoc,* vol. 88, pp. 265-6, Jul 2000.

[47] W. K. Seng, M. H. Kim, R. Besar, and F. Salleh, "A Secure Model for Medical Data Sharing," *International Journal of Database Theory and Application,* vol. 1, pp. 45-52.

[48] S. Bennett, M. Bhuller, and R. Covington. (2009, August 2009). Architectural Strategies for Cloud Computing.

[49] A. Rosenthal, P. Mork, M. H. Li, J. Stanford, D. Koester, and P. Reynolds, "Cloud computing: a new business paradigm for biomedical information sharing," *J Biomed Inform,* vol. 43, pp. 342-53, Apr 2010.

[50] *Rackspace Hosting*. Available: http://www.rackspace.com/ (Available September 2015)

[51] E. F. Codd, "Relational database: a practical foundation for productivity," *Communications of the ACM,* vol. 25, pp. 109-117, 1982.

[52] M. T. Ozsu and P. Valduriez, *Principles of distributed database systems*: Springer-Verlag New York Inc, 2011.

[53] J. Pokorny, "NoSQL databases: a step to database scalability in web environment," 2011.

[54] R. Cattell, "Scalable sql and nosql data stores," *ACM SIGMOD Record,* vol. 39, pp. 12-27, 2011.

[55] R. P. Padhy, M. R. Patra, and S. C. Satapathy, "RDBMS to NoSQL: Reviewing Some Next-Generation Non-Relational Database'sǁ," *International Journal of Advanced Engineering Science and Technologies,* vol. 11, pp. 15-30, 2011.

[56] J. Lennon, "Introduction to couchdb," *Beginning CouchDB,* pp. 3-9, 2009.

[57] K. Banker, "MongoDB in Action," ed: Manning Pubs Co Series. Manning Publications, 2011.

[58] D. Featherston, "Cassandra: Principles and Application," *University of Illinois,* 2010.

[59] M. N. Vora, "Hadoop-HBase for large-scale data," 2011.

[60] G. Manyam, M. A. Payton, J. A. Roth, L. V. Abruzzo, and K. R. Coombes, "Relax with CouchDB-Into the non-relational DBMS era of Bioinformatics," *Genomics,* 2012.

[61] R. C. Taylor, "An overview of the Hadoop/MapReduce/HBase framework and its current applications in bioinformatics," *BMC bioinformatics,* vol. 11, p. S1, 2010.

[62] M. Corporation. *Windows Azure Platform*. Available: http://azure.microsoft.com/ (September 2015)

[63] C. Goble and R. Stevens, "State of the nation in data integration for bioinformatics," *Journal of biomedical informatics,* vol. 41, pp. 687-693, 2008.

[64] C. White, "The next generation of business intelligence: operational BI," *DM Review Magazine,* 2005.

[65] E. Kerkri, C. Quantin, F. Allaert, Y. Cottin, P. Charve, F. Jouanot*, et al.*, "An approach for integrating heterogeneous information sources in a medical data warehouse," *Journal of Medical Systems,* vol. 25, pp. 167-176, 2001.

[66] R. Lenz and K. Kuhn, "Intranet meets hospital information systems: the solution to the integration problem?," *Methods Inf Med,* vol. 40, pp. 99-105, 2001.

[67] B. Devlin and L. D. Cote, *Data warehouse: from architecture to implementation*: Addison-Wesley Longman Publishing Co., Inc., 1996.

[68] L. Niswonger, M. T. Roth, P. Schwarz, and E. Wimmers, "Transforming heterogeneous data with database middleware: Beyond integration," *Bulletin of the Technical Committee on,* p. 31, 1999.

[69] L. M. Haas, E. T. Lin, and M. A. Roth, "Data integration through database federation," *IBM Systems Journal,* vol. 41, pp. 578-596, 2002.

[70] R. Perrey and M. Lycett, "Service-oriented architecture," 2003.

[71] A. D. Giordano, *Data Integration Blueprint and Modeling: Techniques for a Scalable and Sustainable Architecture*: IBM Press, 2010.

[72] T. Erl, *Service-oriented architecture: concepts, technology, and design*: Prentice Hall PTR, 2005.

[73] W3C, "RDF - Semantic Web Standards," ed, 2014.

[74] W3C, "OWL 2 Web Ontology Language - Document Overview (Second Edition)," ed, 2014.

[75] E. Bonsma and J. Vrijnsen, "Homogenising access to heterogeneous biomedical data sources," *George Potamias Vassilis Moustakis (eds.),* p. 9, 2009.

[76] I. N. Sarkar, "Biomedical informatics and translational medicine," *Journal of translational medicine,* vol. 8, p. 22, 2010.

[77] N. Gehlenborg, S. I. O'Donoghue, N. S. Baliga, A. Goesmann, M. A. Hibbs, H. Kitano*, et al.*, "Visualization of omics data for systems biology," *Nature methods,* vol. 7, pp. S56-S68, 2010.

[78] C. Angulo, E. Reig, J. A. Maldonado, D. Moner, D. Boscá, D. Pérez*, et al.*, "Pangea-LE: a non-invasive lightweight biomedical data integration engine," 2008.

[79] M. Robles, J. Fernández-Breis, J. Maldonado, D. Moner, C. Martínez-Costa, D. Bosca*, et al.*, "ResearchEHR: use of semantic web technologies and archetypes for the description of EHRs," *Studies in health technology and informatics,* vol. 155, p. 129, 2010.

[80] 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," *Journal of Biomedical Informatics,* 2011.

[81] J. A. Maldonado, D. Moner, D. Boscá, J. T. Fernández-Breis, C. Angulo, and M. Robles, "LinkEHR-Ed: A multi-reference model archetype editor based on formal semantics," *International Journal of Medical Informatics,* vol. 78, pp. 559-570, 2009.

[82] T. Beale, S. Heard, D. Kalra, and D. Lloyd, "OpenEHR architecture overview," *The OpenEHR Foundation,* 2006.

[83] Y. H. Shih, C. Y. Lien, C. H. Chen, C. H. Hsiao, and W. C. Chu1Ж, "A Secure Knowledge Discovery Framework for Clinical Informatics."

[84] A. N. E. Fadly, B. Rance, N. Lucas, C. Mead, G. Chatellier, P. Y. Lastic*, et al.*, "Integrating clinical research with the healthcare enterprise: from the RE-USE project to the EHR4CR platform," *Journal of biomedical informatics,* 2011.

[85] C. Costa, F. Freitas, M. Pereira, A. Silva, and J. Oliveira, "Indexing and retrieving DICOM data in disperse and unstructured archives," *International Journal of Computer Assisted Radiology and Surgery,* vol. 4, pp. 71-77, 2009.

[86] M. Santos, L. Bastião, C. Costa, A. Silva, and N. Rocha, "DICOM and Clinical Data Mining in a Small Hospital PACS: A Pilot Study." vol. 221, M. M. Cruz-Cunha, J. Varajão, P. Powell, and R. Martinho, Eds., ed: Springer Berlin Heidelberg, 2011, pp. 254-263.

[87] E. Monteiro, L. Bastião, and C. Costa, "CloudMed: Promoting Telemedicine Processes Over the Cloud," presented at the 7th Iberian Conference on Information Systems and Technologies, Madrid, 2012.

[88] R. A. Heckemann, T. Hartkens, K. K. Leung, Y. Zheng, D. L. G. Hill, J. V. Hajnal*, et al.*, "Information extraction from medical images: Developing an e-Science application based on the Globus toolkit," 2003.

[89] J. L. Oliveira, G. M. S. Dias, I. F. C. Oliveira, P. D. N. S. d. Rocha, I. Hermosilla , J. Vicente*, et al.*, "DiseaseCard: A Web-based Tool for the Collaborative Integration of Genetic and Medical Information," in *5th International Symposium, ISBMDA 2004: Biological and Medical Data Analysis*, 2004, pp. 409-417.

[90] S. Madhavan, J. C. Zenklusen, Y. Kotliarov, H. Sahni, H. A. Fine, and K. Buetow, "Rembrandt: helping personalized medicine become a reality through integrative translational research," *Molecular Cancer Research,* vol. 7, p. 157, 2009.

[91] S. N. Murphy, M. E. Mendis, D. A. Berkowitz, I. Kohane, and H. C. Chueh, "Integration of clinical and genetic data in the i2b2 architecture," 2006.

[92] M. Reich, T. Liefeld, J. Gould, J. Lerner, P. Tamayo, and J. P. Mesirov, "GenePattern 2.0," *Nature genetics,* vol. 38, pp. 500-501, 2006.

[93] H. Kuehn, A. Liberzon, M. Reich, and J. P. Mesirov, "Using GenePattern for gene expression analysis," *Curr Protoc Bioinformatics,* 2008.

[94] L. S. Ribeiro, C. Costa, and J. L. Oliveira, "Clustering of distinct PACS archives using a cooperative peer-to-peer network," *Computer Methods and Programs in Biomedicine,* vol. 108, pp. 1002-1011, 2012.

[95] G. Weisser, S. Ruggiero, A. Runa, C. Düber, W. Neff, and M. Walz, "Online Availability Check of Teleradiology Components," *Journal of Digital Imaging,* vol. 20, pp. 393-401, 2007/12/01 2007.

[96] H. K. Huang, "PACS and imaging informatics: Basic Principles and Applications," 2004.

[97] B. Rimal and E. Choi, "A Conceptual Approach for Taxonomical Spectrum of Cloud Computing," in *Ubiquitous Information Technologies & Applications, 2009. ICUT '09. - Proceedings of the 4th International Conference* Fukuoka, 2010, pp. 1-6.

[98] N. E. M. A.-. NEMA, "Digital Imaging and Communications in Medicine (DICOM) - Part 3," in *Information Object Definitions*, ed.

[99] DICOM-P4, "Digital Imaging and Communications in Medicine (DICOM), Part 4: Service Class Specifications," National Electrical Manufacturers Association2015.

[100] DICOM-P7, "Digital Imaging and Communications in Medicine (DICOM), Part 7: Message Exchange," National Electrical Manufacturers Association2015.

[101] G. V. Koutelakis and D. K. Lymberopoulos, "WADA service: an extension of DICOM WADO service," *Trans. Info. Tech. Biomed.,* vol. 13, pp. 121-130, 2009.

[102] A. C. R. Nema, "DICOM Supplement 163: STore Over the Web by REpresentations State Transfer (REST) Services (STOW-RS)," ed, 2011.

[103] A. C. R. Nema, "DICOM Supplement 166: Query based on ID for DICOM Objects by Representational State Transfer (REST) Services (QIDO-RS)," ed.

[104] IHE, "Cross-Enterprise Document Sharing for Imaging (XDS-I)," ed.

[105] L. S. Ribeiro, L. Bastião, C. Costa, and J. L. Oliveira, "EMAIL-P2P GATEWAY to Distributed Medical Imaging Repositories," in *HealthInf 2010*, Spain, Valencia, 2010.

[106] G. Weisser, U. Engelmann, S. Ruggiero, A. Runa, A. Schrˆter, S. Baur*, et al.*, "Teleradiology applications with DICOM-e-mail," *European radiology,* vol. 17, pp. 1331-1340, 2007.

[107] J. A. Hernandez, C. J. Acuna, M. V. de Castro, E. Marcos, M. López, and N. Malpica, "Web-PACS for multicenter clinical trials," *Information Technology in Biomedicine, IEEE Transactions on,* vol. 11, pp. 87-93, 2007.

[108] DICOM-P18, "Digital Imaging and Communications in Medicine (DICOM), Part 18: Web Access to DICOM Persistent Objects (WADO)." National Electrical Manufacturers Association2015.

[109] C. Yang, C. Chen, and M. Yang, "Implementation of a medical image file accessing system in co-allocation data grids," *Future Generation Computer Systems,* 2010.

[110] A. Sharma, T. Pan, B. B. Cambazoglu, M. Gurcan, T. Kurc, and J. Saltz, "VirtualPACS--a federating gateway to access remote image data resources over the grid," *J Digit Imaging,* vol. 22, pp. 1-10, Mar 2009.

[111] B. J. Liu, M. Z. Zhou, and J. Documet, "Utilizing data grid architecture for the backup and recovery of clinical image data," *Comput Med Imaging Graph,* vol. 29, pp. 95-102, Mar-Apr 2005.

[112] H. Huang, A. Zhang, B. Liu, Z. Zhou, J. Documet, N. King*, et al.*, "Data grid for large-scale medical image archive and analysis," 2005, pp. 1005-1013.

[113] C. Chen and W. Wang, "Implementation of a Medical Image File Accessing System on Cloud Computing," in *Computational Science and Engineering (CSE) - 2010*, Hong Kong, China, 2010.

[114] C. Viana-Ferreira, C. Costa, and J. L. Oliveira, "Dicoogle relay-a cloud communications bridge for medical imaging," in *Computer-Based Medical Systems (CBMS), 2012 25th International Symposium on*, 2012, pp. 1-6.

[115] D. S. Marcus, T. R. Olsen, M. Ramaratnam, and R. L. Buckner, "The extensible neuroimaging archive toolkit," *Neuroinformatics,* vol. 5, pp. 11-33, 2007.

[116] (2015). *BioMedBridges*. Available: http://www.biomedbridges.eu/ (Available September 2015)

[117] S. Klein, E. Vast, J. van Soest, A. Dekker, M. Koek, and W. Niessen, "XNAT imaging platform for BioMedBridges and CTMM TraIT," *Journal of Clinical Bioinformatics,* vol. 5, p. S18, 2015.

[118] L. S. Ribeiro, C. Costa, and J. L. Oliveira, "Current Trends in Archiving and Transmission of Medical Images," vol. 5. pp. 91-106 ed, 2011.

[119] J. Zhang, K. Zhang, Y. Yang, J. Sun, T. Ling, G. Wang*, et al.*, "Grid-based implementation of XDS-I as part of image-enabled EHR for regional healthcare in Shanghai," *International Journal of Computer Assisted Radiology and Surgery,* vol 6. pp. 1-12, 2011.

[120] L. S. Ribeiro, C. Viana-Ferreira, J. L. Oliveira, and C. Costa, "XDS-I outsourcing proxy: ensuring confidentiality while preserving interoperability," *Biomedical and Health Informatics, IEEE Journal of,* vol. 18, pp. 1404-1412, 2014.

[121] T. Marques Godinho, C. Viana-Ferreira, L. Bastiao Silva, and C. Costa, "A Routing Mechanism for Cloud Outsourcing of Medical Imaging Repositories," 2014.

[122] T. M. Godinho, L. Silva, C. Viana-Ferreira, C. Costa, and J. L. Oliveira, "Enhanced regional network for medical imaging repositories," in *Information Systems and Technologies (CISTI), 2013 8th Iberian Conference on*, 2013, pp. 1-6.

[123] T. M. Godinho, L. A. B. Silva, C. Costa, and J. L. Oliveira, "Multi-provider Architecture for Cloud Outsourcing of Medical Imaging Repositories," *EHealth-For Continuity of Care: Proceedings of MIE2014,* vol. 205, p. 146, 2014.

[124] O. S. Pianykh, *Digital Imaging and Communications in Medicine (DICOM): A Practical Introduction and Survival Guide*: Springer, 2008.

[125] L. A. B. Silva, C. Costa, and J. L. Oliveira, "A common API for delivering services over multi-vendor cloud resources," *Journal of Systems and Software,* vol. 86, pp. 2309-2317, 2013.

[126] B. P. Rimal, A. Jukan, D. Katsaros, and Y. Goeleven, "Architectural requirements for cloud computing systems: an enterprise cloud approach," *Journal of Grid Computing,* vol. 9, pp. 3-26, 2011.

[127] L. Silva, C. Costa, and J. L. Oliveira, "An agile framework to support distributed medical imaging scenarios," in *Healthcare Informatics (ICHI), 2013 IEEE International Conference on*, 2013, pp. 345-350.

[128] E. Pinho, L. Bastiao Silva, and C. Costa, "A cloud service integration platform for web applications," in *High Performance Computing & Simulation (HPCS), 2014 International Conference on*, 2014, pp. 366-373.

[129] OsiriX. *Osirix DICOM Viewer*. Available: http://www.osirix-viewer.com/ (Available September 2015)

[130] O. Ratib and A. Rosset, "Open-source software in medical imaging: development of OsiriX," *International Journal of Computer Assisted Radiology and Surgery,* vol. 1, pp. 187-196, 2006.

[131] Offis.. *dcmtk*. Available: http://dicom.offis.de/ (Available September 2015)

[132] N. C. Institute.. *Conquest DICOM Software*. Available: http://ingenium.home.xs4all.nl/dicom.html (Available September 2015)

[133] PubNub. *PubNub*. Available: http://www.pubnub.com/ (Available September 2015)

[134] C. Costa, C. Ferreira, L. Bastião, L. Ribeiro, A. Silva, and J. L. Oliveira, "Dicoogle - an Open Source Peer-to-Peer PACS," *Journal of Digital Imaging,* vol. 1, pp. 1-9, 2010.

[135] D. Ludwick and J. Doucette, "Adopting electronic medical records in primary care: lessons learned from health information systems implementation experience in seven countries," *International Journal of Medical Informatics,* vol. 78, p. 22, 2009.

[136] S. Wang, W. Pavlicek, C. C. Roberts, S. G. Langer, M. Zhang, M. Hu*, et al.*, "An automated DICOM database capable of arbitrary data mining (including radiation dose indicators) for quality monitoring," *Journal of Digital Imaging,* vol. 24, pp. 223-33, Apr 2011.

[137] Y.-F. Wang, M.-Y. Chang, R.-D. Chiang, L.-J. Hwang, C.-M. Lee, and Y.-H. Wang, "Mining Medical Data: A Case Study of Endometriosis," *Journal of medical systems,* vol. 37, pp. 1-7, 2013.

[138] D. A. Share, D. A. Campbell, N. Birkmeyer, R. L. Prager, H. S. Gurm, M. Moscucci*, et al.*, "How a regional collaborative of hospitals and physicians in Michigan cut costs and improved the quality of care," *Health Affairs,* vol. 30, pp. 636-645, 2011.

[139] A. G. de Belvis, F. Francesca, S. M. Lucia, V. Luca, F. Giovanni, and R. Walter, "The financial crisis in Italy: Implications for the healthcare sector," *Health Policy,* 2012.

[140] S. Nuti, M. Vainieri, and M. Frey, "Healthcare resources and expenditure in financial crisis: scenarios and managerial strategies," *Journal of Maternal-Fetal and Neonatal Medicine,* vol. 25, pp. 40-43, 2012.

[141] I. El Azami, M. O. C. Malki, and C. Tahon, "Integrating Hospital Information Systems in Healthcare Institutions: A Mediation Architecture," *Journal of medical systems,* vol. 36, pp. 3123-3134, 2012.

[142] O. S. Pianykh, *Digital imaging and communications in medicine (DICOM)*: Springer, 2008.

[143] R. H. Dolin, L. Alschuler, S. Boyer, C. Beebe, F. M. Behlen, P. V. Biron*, et al.*, "HL7 clinical document architecture, release 2," *Journal of the American Medical Informatics Association,* vol. 13, pp. 30-39, 2006.

[144] H. L. Seven, "HL7 Version 3 Standard: Section 3: Clinical and Administrative Domains," ed, 2014.

[145] A. J. McMurry, S. N. Murphy, D. MacFadden, G. Weber, W. W. Simons, J. Orechia*, et al.*, "SHRINE: Enabling Nationally Scalable Multi-Site Disease Studies," *PLOS ONE,* vol. 8, p. e55811, 2013.

[146] S. Madhavan, J.-C. Zenklusen, Y. Kotliarov, H. Sahni, H. A. Fine, and K. Buetow, "Rembrandt: helping personalized medicine become a reality through integrative translational research," *Molecular Cancer Research,* vol. 7, pp. 157-167, 2009.

[147] S. Gutman and L. G. Kessler, "The US Food and Drug Administration perspective on cancer biomarker development," *Nature Reviews Cancer,* vol. 6, pp. 565-571, 2006.

[148] J. Pathak, J. Wang, S. Kashyap, M. Basford, R. Li, D. R. Masys*, et al.*, "Mapping clinical phenotype data elements to standardized metadata repositories and controlled terminologies: the eMERGE Network experience," *Journal of the American Medical Informatics Association,* vol. 18, pp. 376-386, 2011.

[149] D. L. Rubin and T. S. Desser, "A data warehouse for integrating radiologic and pathologic data," *J Am Coll Radiol,* vol. 5, pp. 210-7, Mar 2008.

[150] T. Botsis, G. Hartvigsen, F. Chen, and C. Weng, "Secondary use of EHR: data quality issues and informatics opportunities," *AMIA summits on translational science proceedings,* vol. 2010, p. 1, 2010.

[151] H. E. Källman, E. Halsius, M. Olsson, and M. Stenström, "DICOM Metadata repository for technical information in digital medical images," *Acta Oncol,* vol. 48, pp. 285-8, 2009.

[152] X. Zhou, S. Chen, B. Liu, R. Zhang, Y. Wang, P. Li*, et al.*, "Development of traditional Chinese medicine clinical data warehouse for medical knowledge discovery and decision support," *Artificial Intelligence in Medicine,* vol. 48, pp. 139-152, 2010.

[153] T. Rajala, S. Savio, J. Penttinen, P. Dastidar, M. Kähönen, H. Eskola*, et al.*, "Development of a research dedicated archival system (TARAS) in a university hospital," *Journal of digital imaging,* vol. 24, pp. 864-873, 2011.

[154] A. Jahnen, S. Kohler, J. Hermen, D. Tack, and C. Back, "Automatic computed tomography patient dose calculation using DICOM header metadata," *Radiat Prot Dosimetry,* vol. 147, pp. 317-20, Sep 2011.

[155] L. M. Prevedello, K. P. Andriole, R. Hanson, P. Kelly, and R. Khorasani, "Business intelligence tools for radiology: creating a prototype model using open-source tools," *J Digit Imaging,* vol. 23, pp. 133-41, Apr 2010.

[156] F. Valente, L. A. B. Silva, T. Godinho, and C. Costa, "Anatomy of an Extensible Open Source PACS," *Journal of Digital Imaging,* 2015.

[157] M. L. Braunstein, "Health Big Data and Analytics," in *Practitioner's Guide to Health Informatics*, ed: Springer, 2015, pp. 133-149.

[158] IHE, "Quality, Research, and Public Health Profiles," ed.

[159] T. Skripcak, C. Belka, W. Bosch, C. Brink, T. Brunner, V. Budach*, et al.*, "Creating a data exchange strategy for radiotherapy research: Towards federated databases and anonymised public datasets," *Radiotherapy and Oncology,* vol. 113, pp. 303-309, 2014.

[160] I. Neamatullah, M. Douglass, L.-w. Lehman, A. Reisner, M. Villarroel, W. Long*, et al.*, "Automated de-identification of free-text medical records," *BMC medical informatics and decision making,* vol. 8, p. 32, 2008.

[161] Ö. Uzuner, Y. Luo, and P. Szolovits, "Evaluating the state-of-the-art in automatic de-identification," *Journal of the American Medical Informatics Association,* vol. 14, pp. 550-563, 2007.

[162] L. A. B. Bastião, C. Costa, and J. L. Oliveira, "A PACS archive architecture supported on cloud services," *International Journal of Computer Assisted Radiology and Surgery,* vol. 7, pp. 349-358, 2012.

[163] Internacional Standarization Organization (ISO), "ISO 3166," ed.

[164] L. Bastiao Silva, L. Beroud, C. Costa, and J. L. Oliveira, "Medical imaging archiving: A comparison between several NoSQL solutions," in *Biomedical and Health Informatics (BHI), 2014 IEEE-EMBS International Conference on*, 2014, pp. 65-68.

[165] S. Campos, C. Costa, and L. Bastião, "A Network Sensor for Medical Imaging Workflows," presented at the 7th Iberian Conference on Information Systems and Technologies, Madrid, 2012.

[166] M. Santos, L. Bastião, C. Costa, A. Silva, and N. Rocha, "DICOM and clinical data mining in a small hospital pacs: A pilot study," in *ENTERprise Information Systems*, ed: Springer, 2011, pp. 254-263.

[167] R. Feneck, J. Kneeshaw, and M. Ranucci, *Core Topics in Transesophageal Echocardiography*: Cambridge University Press, 2010.

[168] A. S. S. Brás, J. Ribeiro, J.L. Oliveira, "New insights in echocardiography based left-ventricle dynamics assessment," in *International Work-Conference on Bioinformatics and Biomedical Engineering*, 2015.

[169] E. P. Tamm, X. J. Rong, D. D. Cody, R. D. Ernst, N. E. Fitzgerald, and V. Kundra, "Quality initiatives: CT radiation dose reduction: how to implement change without sacrificing diagnostic quality," *Radiographics,* vol. 31, pp. 1823-1832, 2011.

[170] T. S. Cook, S. L. Zimmerman, S. R. Steingall, A. D. Maidment, W. Kim, and W. W. Boonn, "Informatics in radiology: RADIANCE: an automated, enterprise-wide solution for archiving and reporting CT radiation dose estimates," *Radiographics,* vol. 31, pp. 1833-1846, 2011.

[171] M. S. Luis A. Bastião Silva, Carlos Costa, José Luis Oliveira, "Dicoogle Statistics: analyzing efficiency and service quality of digital imaging laboratories," in *27th Computer Assisted Radiology and Surgery* Heidelberg, Germany, 2013.

[172] C. H. McCollough, G. H. Chen, W. Kalender, S. Leng, E. Samei, K. Taguchi*, et al.*, "Achieving routine submillisievert CT scanning: report from the summit on management of radiation dose in CT," *Radiology,* vol. 264, pp. 567-580, 2012.

[173] B. Ganeshan, E. Panayiotou, K. Burnand, S. Dizdarevic, and K. Miles, "Tumour heterogeneity in non-small cell lung carcinoma assessed by CT texture analysis: a potential marker of survival," *European radiology,* vol. 22, pp. 796-802, 2012.

[174] Milton Santos, Luis A. Bastião Silva, Milton Santos, Carlos Costa, José Luis Oliveira, C. Costa*, et al.*, "Clinical Data Mining in Small Hospital PACS: Contributions for Radiology," *Information Systems and Technologies for Enhancing Health and Social Care,* p. 236, 2013.

[175] J. A. Christner, J. M. Kofler, and C. H. McCollough, "Estimating effective dose for CT using dose–length product compared with using organ doses: consequences of adopting International Commission on Radiological Protection Publication 103 or dual-energy scanning," *American Journal of Roentgenology,* vol. 194, pp. 881-889, 2010.

[176] P. Lopes, L. A. B. Silva, J. L. Oliveira, J. L. Oliveira, and C. U. de Santiago, "Challenges and Opportunities for Exploring Patient-level Data."

[177] A. El Fadly, B. Rance, N. Lucas, C. Mead, G. Chatellier, P.-Y. Lastic*, et al.*, "Integrating clinical research with the Healthcare Enterprise: From the RE-USE project to the EHR4CR platform," *Journal of Biomedical Informatics,* vol. 44, Supplement 1, pp. S94-S102, 12// 2011.

[178] J. L. Oliveira, P. Lopes, T. Nunes, D. Campos, S. Boyer, E. Ahlberg*, et al.*, "The EU‐ADR Web Platform: delivering advanced pharmacovigilance tools," *Pharmacoepidemiology and Drug Safety,* vol. 22, pp. 459-467, 2013.

[179] M. Schuemie, R. Gini, P. Coloma, H. Straatman, R. C. Herings, L. Pedersen*, et al.*, "Replication of the OMOP Experiment in Europe: Evaluating Methods for Risk Identification in Electronic Health Record Databases," *Drug Safety,* vol. 36, pp. 159-169, 2013/10/01 2013.

[180] J. Ison, M. Kalaš, I. Jonassen, D. Bolser, M. Uludag, H. McWilliam*, et al.*, "EDAM: an ontology of bioinformatics operations, types of data and identifiers, topics and formats," *Bioinformatics,* vol. 29, pp. 1325-1332, 2013.

[181] P. A. Harris, R. Taylor, R. Thielke, J. Payne, N. Gonzalez, and J. G. Conde, "Research electronic data capture (REDCap)-A metadata-driven methodology and workflow process for providing translational research informatics support," *Journal of Biomedical Informatics,* vol. 42, p. 377, 2009.

[182] R. Platt and R. Carnahan, "The US Food and Drug Administration's Mini‐Sentinel Program," *Pharmacoepidemiology and Drug Safety,* vol. 21, pp. 1-303, 2012.

[183] M. A. Robb, J. A. Racoosin, R. E. Sherman, T. P. Gross, R. Ball, M. E. Reichman*, et al.*, "The US Food and Drug Administration's Sentinel Initiative: expanding the horizons of medical product safety," *Pharmacoepidemiology and Drug Safety,* vol. 21, pp. 9-11, 2012.

[184] *BridgeToData*. Available: http://www.bridgetodata.org/ (Available September 2015)

[185] C. Viertler and K. Zatloukal, "[Biobanking and Biomolecular Resources Research Infrastructure (BBMRI). Implications for pathology]," *Der Pathologe,* vol. 29, pp. 210-213, 2008.

[186] G.-J. B. van Ommen, O. Törnwall, C. Bréchot, G. Dagher, J. Galli, K. Hveem*, et al.*, "BBMRI-ERIC as a resource for pharmaceutical and life science industries: the development of biobank-based Expert Centres," *European Journal of Human Genetics,* 2014.

[187] P. Coorevits, M. Sundgren, G. O. Klein, A. Bahr, B. Claerhout, C. Daniel*, et al.*, "Electronic health records: new opportunities for clinical research," *Journal of Internal Medicine,* vol. 274, pp. 547-560, 2013.

[188] C. T. Smith, P. R. Williamson, and A. G. Marson, "Investigating heterogeneity in an individual patient data meta-analysis of time to event outcomes," *Statistics in Medicine,* vol. 24, pp. 1307-1319, 2005.

[189] A. P. Abernethy, A. Ahmad, S. Y. Zafar, J. L. Wheeler, J. B. Reese, and H. K. Lyerly, "Electronic Patient-Reported Data Capture as a Foundation of Rapid Learning Cancer Care," *Medical Care,* vol. 48, pp. S32-S38 10.1097/MLR.0b013e3181db53a4, 2010.

[190] P. Nisen and F. Rockhold, "Access to Patient-Level Data from GlaxoSmithKline Clinical Trials," *New England Journal of Medicine,* vol. 369, pp. 475-478, 2013.

[191] B. Wieseler, N. Wolfram, N. McGauran, M. F. Kerekes, V. Vervölgyi, P. Kohlepp*, et al.*, "Completeness of reporting of patient-relevant clinical trial outcomes: comparison of unpublished clinical study reports with publicly available data," *PLoS Medicine,* vol. 10, p. e1001526, 2013.

[192] P. Liu, J. Carpentier, P. Cheynet, L.-A. Denis, B. Guillon, G. Kremenek*, et al.*, "eTRIKS Cloud Platform: a Platform for Knowledge Management," in *International Symposium on Grids and Clouds (ISGC)*, 2015.

[193] H. Zetterberg, "Unresolved questions in Alzheimer's research: will biomarkers help?," *Biomarkers in medicine,* vol. 8, pp. 61-63, 2014.

[194] P. J. Visser, J. R. Streffer, and S. Lovestone, "A European Medical Information Framework for Alzheimer's disease (EMIF-AD)," *Alzheimer's & Dementia: The Journal of the Alzheimer's Association,* vol. 10, p. P799, 2014.

[195] G. Hripcsak, J. Duke, N. Shah, C. Reich, V. Huser, M. Schuemie*, et al.*, "Observational Health Data Sciences and Informatics (OHDSI): Opportunities for Observational Researchers," *Studies in health technology and informatics,* vol. 216, pp. 574-578, 2014.

[196] OHDSI, "OMOP Common Data Model Specifications Version 5.0," ed, 2014.

[197] P. E. Stang, P. B. Ryan, J. A. Racoosin, J. M. Overhage, A. G. Hartzema, C. Reich*, et al.*, "Advancing the science for active surveillance: rationale and design for the Observational Medical Outcomes Partnership," *Annals of internal medicine,* vol. 153, pp. 600-606, 2010.

[198] R. Gini, P. B. Ryan, J. S. Brown, E. Vacchi, M. Coppola, W. Cazzola*, et al.*, "Comparison Among EU-ADR, OMOP, Mini-Sentinel And MATRICE Strategies For

Data Extraction And Management," in *PHARMACOEPIDEMIOLOGY AND DRUG SAFETY*, 2013, pp. 189-190.

[199] T. Nunes, D. Campos, S. Matos, and J. L. Oliveira, "BeCAS: biomedical concept recognition services and visualization," *Bioinformatics,* vol. 29, pp. 1915-1916, 2013.

Annex A. Service Delivery Cloud Platform (SDCP)

In order to define an architecture to abstract the cloud layers we did a deep study of how the cloud was organized and in this annex we detailed how we design our approach. The Service Delivery Cloud Platform (SDCP) has its own entities that model the system architecture and describe how it is structured. The fundamental entities and associations of the infrastructure are described in Figure 6.1:

Agent – each gateway has to login using an agent account. Basically, agents are the entities situated inside the enterprise that relay the information to the cloud.

Domain – is a group of agents belonging to the same enterprise or the same trustable group/enterprise group. Thus, only agents of the same domain can communicate or access the data belonging to its domain.

Provider – defines a cloud provider and credentials to access them. It can be a storage, database or communication provider. These providers also belong to a domain.

Private Service – external services that can take advantage of Cloud Controller agents and cloud providers. This service will extend the functionality of the Cloud Controller.

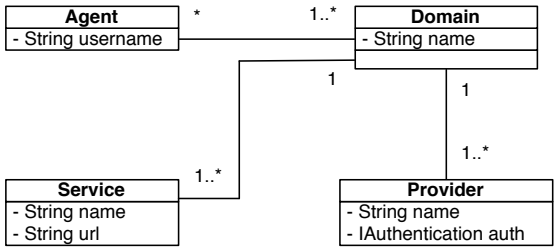

Figure 6.1 - Entities of Service Delivery Cloud Platform

These entities are actors and concepts of the SDCP. The domain is a very important concept because it characterizes the trustable model, i.e. models the relationships and the manages the control of the resources.

1. Cloud services

This section describes the implemented Cloud services. We will describe the three implemented cloud services and how the abstraction for these services was applied.

Cloud Streams

As expressed, the goal of our platform was to use any resources of the cloud without being locked to a specific provider. To implement this feature for storage services, we used an abstraction to write a set of bytes (i.e. blob) into the Cloud storage using typical Input/Output (I/O) streams. The designed abstraction assures provider independency but also makes it easier to extend to other cloud solutions. Two new I/O entities were implemented: *CloudInputStream* and *CloudOutputStream*, Figure 6.2. These entities are used to read/write in the storage services as a common Java stream mechanism. An important aspect regarding the writing of a blob is the access policy. By default, we assume that the blob is private, although the user can specify an ACL (Access Control List) to give permission to the blob.

A blobstore API has different features implemented in different cloud providers. Although several features are presented in the blobstore API, others are not often presented. Our abstraction will not consider these features, and in that case an extension to the platform will be necessary. Nonetheless, we take into account that several features are just used occasionally, and a trade-off was necessary.

At present, most cloud storage solutions do not offer an option to encrypt data when it is uploaded to the cloud. Our platform has an encryption/decryption layer on the client side, i.e. the cipher and decipher operations are executed on-the-fly on the enterprise side, through our abstraction. In that case, it is ciphered with AES (Advance Encryption Standard) algorithm and the key is stored in the Cloud Controller. On the other hand, multiple cloud providers can be supplied with a list of *CloudSocket* being blobs written in both and read from the first one that is available.

The developed Cloud Streams extend the IO Java streams. The Cloud socket contains the identifier of the implementation that will be used to call the most appropriate one for a specific service. JClouds is an open source framework for cloud development that already provides several cloud players, and as such we decided to build the Cloud Streams as an instance of JClouds blobstores. In addition, we implemented our local storage, following the proposed abstraction. Furthermore, new APIs of different blobstore cloud vendors can be easily implemented using the proposed abstraction.

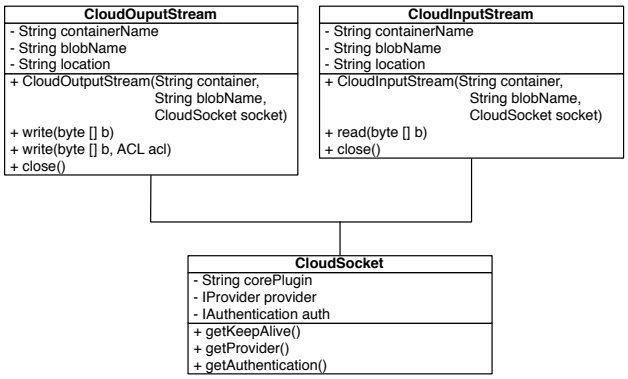

Figure 6.2 - Cloud Input/Output

Columnar data abstraction

As in the previous storage service, we have also developed the same generic API upon cloud databases. This aims to create an abstraction to columnar data, for instance, SimpleDB, Azure Table and other cloud databases publicly available. Nowadays there is a new trend to store information in columnar data instead of the traditional relational system. These tables are very dynamic and the developer does not need to pre-define a model, because the structure auto-fits the data.

There were several problems regarding scalability, which have to be solved in this abstraction. For instance, Amazon Simple DB uses a mainly horizontal scalability, in opposition to Azure Table, which allows control of the vertical partition. Each partition key represents a different node to have the information. This issue was solved through the Table ID, which identifies the Table name, together with the node label or the location label. The idea was to contain generic features that can be applied in many database services. We implemented two of the available APIs, but it will work for other databases. The Java SDK already uses a high-level abstraction for databases, named JPA (Java Persistence API). Although it is widely used with Object-Oriented databases, we decided to follow this standard for two main reasons: JPA is often used by Java developers to abstract the access to databases, and it is easy to keep compatibility with these applications and the chosen an API that fits the JPA abstraction. Thus, it was decided to use the same JPA methods and also add other methods that are specific to the Cloud databases, such as create/remove tables (Figure 6.3). Also, for representation of the results, we use a library named Guava (Google Collections), which provides very generic Java collections, e.g., the Table collection. Table is a triple values class <R, C, V> data structure, i.e. Row, Column and Value. This representation is perfect to retrieve the results of queries and also to insert new data into the columnar tables. Figure 6.3 shows the architecture of the columnar data abstraction. A Select action is executed quite similarly to the JPA method. It uses a small set of the SQL and just conditions are

considered. Complex queries with joins will not be considered in this abstraction, since the columnar tables do not support such a feature.

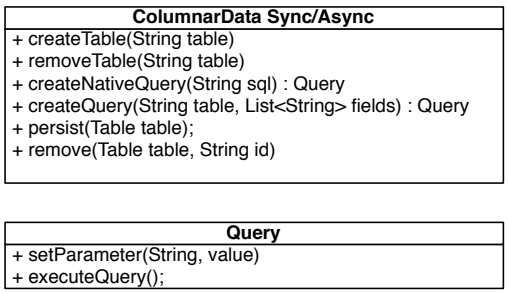

Figure 6.3 - Abstraction columnar data

The data columnar abstraction was deployed on Amazon SimpleDB. The API has a different representation from the JPA abstraction. For instance, each row is called Item, and each item has several attributes that are not structured and can change dynamically for each item. That is why it is called columnar data, because it can be different for each row, i.e. each record can contain different fields. Thus, when creating the table, we do not define a structure as in a common database.

The SimpleDB uses a RESTful web service to supply the programmatic interaction with the developer and the results are retrieved in XML files with the responses. So the first step was to create the XML parsers and client communication with the REST AWS interface, as described in their specification. The abstraction for this service is quite similar to the Cloud Streams, and any specific implementation has to be compliant with interfaces described in Figure 6.3. In the SimpleDB case, we implemented all functions documented in the abstraction. The model copes with the common API with minor conversions.

Notification abstraction

The notification abstraction aims to dynamically create a message-based communication, based on the Publish/Subscribe mechanism. It is asynchronous, allowing application delivery using this platform to tackle the polling issue often implemented in many applications to simulate an asynchronous system. However, not many Publish/Subscribe public services use only HTTP/HTTPS. We will take the example of PubNub, although a new instance can be implemented, for example using other public services such as Channel API of Google AppEngine, or other protocols like XMPP which support the Publish/Subscribe mechanism. Moreover, the polling approach can also fit the abstraction, and in that case the subscriber has to poll the server until it has a signal message, and then it will call the Receiver callback. Also, for instance, Ajax Push Engine (APE) can be installed in a public cloud provider like Amazon EC2, and the service can be used with quite similar behavior to PubNub. Nevertheless, there are also very similar

services based on the Publish/Subscribe model, for instance, Amazon SQS and Azure Queue.

In this service abstraction, we used an Observer Pattern, and in the current implementation we created two entities: Publish and Subscribe (see **Figure 5**). The channel represents the domain of each agent, and it assures that the communication can only be established between agents of the same domain.

It is important to mention that PubNub specific implementation is quite analogous to the one proposed. So the abstraction classes will call the implementation of PubNub directly using the adapter pattern, similarly to the other previously presented abstractions.

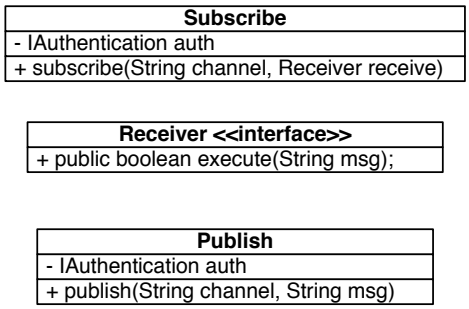

Figure 6.4: Publish/Subscribe abstraction

2. Cloud Controller

The Cloud Controller is a major component of our architecture responsible for functionalities such as: aggregating providers' credentials; controlling access to cloud resources; managing authentication processes with Cloud Gateways; and addition of new services.

This controller provides an API that can be used by third party applications to access their services. The Controller communicates through HTTP/HTTPS, using RESTful specification; thereby it will be much easier for other entities to access services. The Cloud Controller allows us to store credentials of cloud providers for different services, such as blobstore, database and communication (Figure 6.5). Also, the ciphered keys used to cipher and decipher the blobs are stored in the Cloud controller, unless the developer explicitly denies the action. Moreover, it also supports addition of external services used by third party applications, extending in this way the Cloud Controller functionality. This platform was instanced with several end-user services associated with Medical Imaging (Repository Data Privacy), particularly the safe storage of medical data in multiple cloud players.

There is critical information in diverse scenarios. In such cases, the developer can create a new service in a private cloud to keep the more restricted access data. Our platform will

be compatible with public or private clouds. Moreover, the Cloud Gateway can cipher the data before sending it to the cloud, and store the keys in these private services.

RESTful API

The interface to external applications is issued as a RESTful web service that provides several interfaces, starting with an authentication mechanism. User validation is based on username and password and if the login is valid, the web service returns a token that will be used to validate subsequent operations. We created functions to get cloud provider and services information.

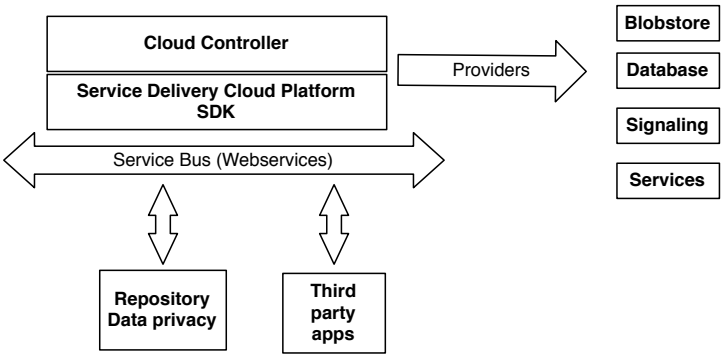

Figure 6.5 - Cloud Controller - Architecture

Dashboard panel

In addition to the web services API, the Cloud Controller also provides a web portal interface (Figure 6.6), whereby administrators can add or remove new cloud providers (storage, database, services, etc) and also check the operation's logs. This portal was implemented through GWT (Google Web Toolkit) technologies. Also, they can create new domains, add/remove/ban agents and add new services. This dashboard also allows the user to setup a threshold of cloud provider requests because the actions of gateways cloud interactions are sent to the Cloud Controller.

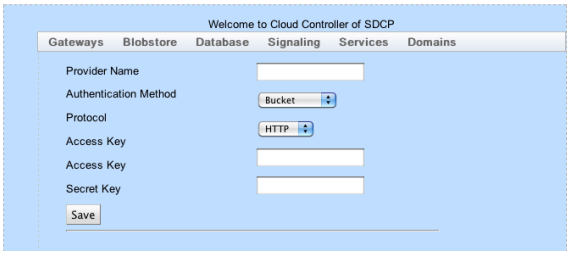

Figure 6.6 - Cloud Controller Dashboard

Cloud Gateway

The Cloud Gateway is a very important component of the architecture. Basically it is an application that loads new services dynamically. It grants authentication from the Cloud Controller and automatically loads the services that are uploaded by the user. Cloud Gateway can run as a daemon. Also, it has an optional external GUI that allows the user to load new plugins/applications or see operation logs. For instance, new adapters for new cloud providers are loaded in the Cloud Gateway.

The architecture of Cloud Gateway (Figure 6.7) also uses the SDCP-SDK. Namely, it has access to the plugin core mechanism to load new plugins. Moreover, the interfaces used in API plugins will be instantiated automatically using the Inversion of Control pattern. The plugins to the Cloud Gateway can be services programmed in Java, directly using the SDCP-SDK, but we offer the possibility of external applications, sending information to the cloud through a web service interface. This raises a question: what is the advantage of using the web service API? Third-party application will be allowed to store, access, and use resources from multiple public clouds, using a normalized interface. Thus, third-party applications do not need to be coupled as a Cloud Gateway Java plugin.

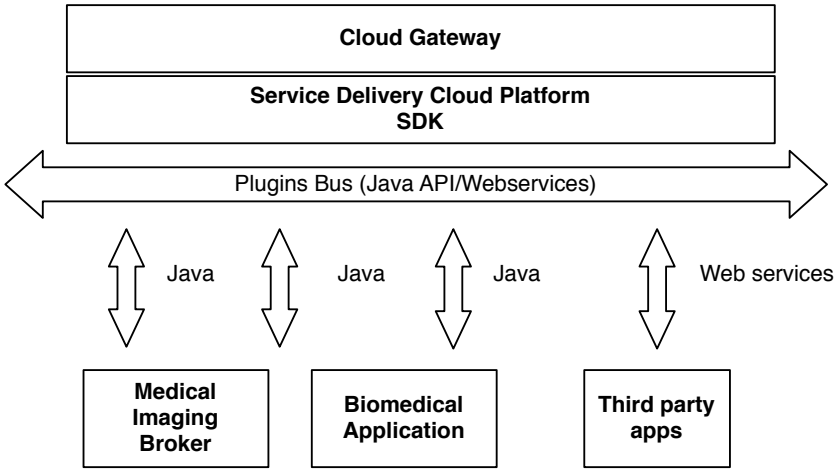

Figure 6.7 - Cloud Gateway architecture

Nearly every web application requires an authentication system. The Cloud Gateway is the middleware layer that allows access to cloud resources, and thereby it requires a user validation system. The Cloud Gateway authentication is used through the RESTful web services that access the Cloud Controller web services, previously described. When the gateway application starts, it requires a username and password for the end-user. Next, Cloud Gateway executes authentication and saves the token.

3. SDCP-SDK

The end-users of the SDCP are allowed to develop applications that use the cloud resources, as well as new plugins to new cloud providers. Thus, the applications can also take advantage of other cloud providers that the developer wants to support. To create these new applications, the developer will use the SDCP-SDK.

The SDCP-SDK defines contracts and specification of the platform, including the communication between the Cloud Controller and the Cloud Gateway. The platform was developed in Java through a set of interfaces. The main idea is that the developer can take advantage of SDCP-SDK to delegate the authorization process to the platform. Also, the access to the cloud resources is provided by the SDK. The new application will be deployed in Cloud Gateway, the entity responsible for loading the applications. On the other hand, the abstractions of blobstore, columnar data and notification systems are also possible to extend using the SDCP-SDK. For instance, it is possible to the developer write a new plugin for a specific provider based on the SDCP-SDK, only implementing the methods described earlier. We developed a plugin for notification system based on PubNub in 8 hours and now, all developed applications with SDCP will benefit of this provider.

4. Privacy and confidenciality model

Undoubtedly, cloud computing has several advantages for enterprises, but two major issues need to be addressed: the cost/benefits of the solution and the privacy and confidentiality of the data stored over the cloud.

The first issue depends on the business, and several studies have been addressing the financial impact of cloud computing. Often associated with data tampering, privacy aspects are still a challenge in these scenarios. Our platform takes those two aspects into consideration, because we can store the information in multiple cloud players and, at the same time, we also tackle the privacy and confidentiality issue. The solution architecture was built taking into account that particular requirement. Our cloud has two main components: Cloud Controller and the cloud players. Thus, for instance, in storage service, we have the opportunity to store the information in a ciphered way.

At present, most cloud solutions do not offer an option to encrypt data when it is uploaded to the cloud. Some companies are already offering this service, for instance, AWS Storage Gateway, but we believe this should be a client service to give more confidence in the cloud solution. Our proposed platform has an encryption/decryption layer on the client side, i.e. the cipher and decipher operations are executed on-the-fly on the enterprise side.

Moreover, this privacy issue is independent of the cloud vendor and the data can be easily sent and accessed in multiple cloud players at the same time. The end-user can do that more easily when writing, specifying a list of cloud providers they intend to use. Use of a common interface should be adopted by services and this will be a contribution to Cloud resources standardization. Another important issue regarding the architecture is that it can be deployed in a hybrid infrastructure. Nonetheless, the Cloud Controller can be deployed in a public or private cloud. Several applications may want to extend the functionality of

the Cloud Controller because it may be relevant to host some information in the public cloud.