Initial anonymized ingestion prototype

Authors
AleksanderStroiński, Tomasz Parkoła (Poznań Supercomputing and Networking Center)

May 2014

This work was partially supported by the SCAPE Project. The SCAPE project is co-funded by the European Union under FP7 ICT-2009.4.1 (Grant Agreement number 270137).

Executive Summary

The Wielkopolskie Center of Pulmonology and Thoracosurgery hospital identified several challenges and requirements in the context of medical data handling, including technical limitations such as lack of storage space and computing power for advanced data processing as well as need for long-term professional archiving system. Therefore it was necessary to develop new approaches for medical data preservation. The solution has been prepared in the framework of the SCAPE project and is composed of various components for scalable medical data preservation and processing, including dedicated data transfer services and anonymization. The proposed toolset is able to reliably and securely transfer medical data from the hospital environment into the data center facilities of Poznan Supercomputing and Networking Center. Moreover, it is relatively easy for any interested party to reuse the solution, as the tools and protocols used are publicly available and conformant with the well-known standards in the medical sector, such as the DICOM transmission protocol and HL7 gateway specification. The technical environment for the anonymized ingestion is composed of scalable tools (e.g. Hadoop, HBase, HDFS), anonymization and personalization tool as well as DICOM and HL7 toolsets. The application of the tools has been proven by the successful execution of several development-level ingestion experiments and will be further analysed in the remaining part of the project.

Table of Contents

1 Introduction

Extension of the SCAPE project brought two new scenarios in the context of data preservation and large-scale processing. The new scenarios are different from the previous in two ways. First, two new data types are investigated in the context of preservation, these are the medical data and video data. Second, a new kind of processing environment has been applied as those scenarios are executed using facilities located in HPC (High-performance computing) and data centers. This document discusses the scenario relating to medical data preservation and processing, with special focus on external data storage, and in particular the medical data anonymization and ingestion processes.

In the context of medical data there are two SCAPE partners focused on the related stories and technical aspects:

- Wielkopolskie Center of Pulmonology and Thoracosurgery (WCPT) which is a hospital located in Poznan, Poland. The hospital acts as a content provider for various kinds of medical data, including X-ray scans (RTG), computed tomography (CT) and other examination results (e.g. in textual format).
- Poznan Supercomputing and Networking Center (PSNC), which is an HPC and data center located in Poznan, Poland. PSNC acts as solution provider in the context of software and hardware facilities for the integration of WCPT's working environment with its HPC and/or data center environment.

This document discusses medical data handling in the context of two core building blocks of the handling procedure, namely ingestion and anonymization. It introduces the rationale and data workflow of the anonymized ingestion prototype, as well as the toolset used to implement medical data transfer, storage and processing activities.

1.1 Challenges with Medical Data

The medical data user stories have emerged from the requirements and challenges identified by the WCPT hospital. The WCPT hospital currently has multiple software components deployed and running within its premises. Unfortunately due to technical limitations it is not possible to preserve the whole hospital documentation in electronic form, therefore only a subset of produced data is stored for the long period. Even if the data were archived in the WCPT storage facilities however, it is not possible for WCPT users to access the data on-line because special procedures are needed to be followed in order to obtain specific digital assets (e.g. archive administrators need to locate the data manually). Additionally, in order to increase teaching and research capabilities of WCPT it is envisioned to provide simple access to the medical data sets and create new ways for large-scale processing of medical data.

It is important to note, that the medical data scenario is highly influenced by the WCPT policy related to sensitive personal data. The policy states that no personal sensitive data can be stored outside of the WCPT facilities, neither in the plain form nor using encryption techniques. Due to this requirement it is necessary to apply anonymization techniques in the context of medical data before ingesting them to external storage facilities. As a result of the analysis related to the medical data, the following key issues at WCPT have been identified:

- Limited storage facilities (WCPT is facing ~20TB of produced data each year)
- No real-time access to full patients history (history kept for only 2 years)
- Need to improve educational activities via medical data exposure, using simple web interface

- Need to perform large-scale analysis of the medical data recorded in the textual format to obtain statistics and analysis at a general level.
- Medical data stored in external facilities, such as data centers, cannot contain personal sensitive data
- The communication between WCPT and PSNC needs to be encrypted.

1.2 User Stories

Based on the above analysis four user stories have been identified and investigated in the context of medical data preservation:

- Data storage – WCPT needs external data storage due to limited resources onsite. WCPT policy forbids transferring sensitive personal data outside its premises. Therefore before ingesting medical data to PSNC, WCPT Hospital Archive System (HAS) component needs to anonymize and/or encrypt specific data.
- Full access – WCPT user needs full access to patient's history, including data coming from the Medical Data Center (MDC) hosted at PSNC and sensitive personal data held only at WCPT. In order to achieve this goal it is necessary to involve DICOM personalization component within the HAS. This component adds personal data to DICOM files, which are further presented to the WCPT user.
- Educational – teachers at medical universities need a simple and straightforward way to showcase specific diseases together with their description prepared by professionals in the field (e.g. radiologists). MDC provides such an opportunity for a broad range of users, as it contains anonymized data which can be shared with the teachers at universities as well as with students.
- Scientific – to support research activities related to medical data it is required to calculate statistics on textual data provided by WCPT. This scenario involves large scale processing of textual data with the use of technologies identified by the SCAPE project (Map-Reduce approach and NoSQL database).

The remainder of this document describes a number of aspects for medical data anonymization and ingestion process which are applicable to these user stories. In the second chapter, details about the medical data characteristics are provided, such as the used file formats and standards, data organization and expected data volumes. The third chapter presents a workflow for handling medical data, including the ingestion process and data storage structure. The fourth chapter details information about the tools used in the anonymization and ingestion processes, including the DICOM toolset, HL7 toolset and anonymization tool. Finally, the last chapter provides a short summary of the work on anonymization and ingestion prototype.

2 Characteristics of the medical data

In order to implement medical data user stories it was required to take into consideration specific characteristics related to medical data such as the file formats or data volumes. This chapter describes important characteristics and requirements related to medical data which is the subject of preservation. It also provides information on the medical data organisation for the needs of the data transfer.

2.1 Preserved information and file formats

Medical data which is to be stored at PSNC facilities are a subset of all medical information stored in the specialized hospital information systems (HIS). This subset corresponds to the specific scenarios and requirements identified by WCPT, and contains:

- Disease type according to ICD10[1] supplemented with disease description and manual description added by the specialist at WCPT hospital.
- Procedures according to the ICD9[2] (an international classification system for surgical, diagnostic and therapeutic procedures) supplemented with the description of the procedure prepared by the specialist at WCPT hospital.
- Examination results from the specialized laboratories. An example can be morphology, biochemistry, urinalysis or gasometry.
- Description of the EKG and date of examination.
- RTG examination results, including DICOM files, date of examination and description of the results prepared by the specialist at WCPT hospital.
- CT (computed tomography) examination results, including DICOM files, date of examination and description of the results prepared by the specialist at WCPT hospital.
- Description of the USG examination prepared by the specialist at WCPT hospital.
- Epicrisis and recommendations for further treatment.
- Information about patient, including complaints, allergies, addiction, blood type and contact with tuberculosis.
- Additional medical observations.

Depending on the data type the information listed above is stored either in HL7 v3 format[3] or in DICOM format[4]. HL7 format is used for storing textual information such as laboratory results or descriptions of examination results. DICOM format is used to store results from the RTG or CT examination, so it is medical imaging information.

2.2 Data organisation

Medical data needs to be appropriately organised and structured for well-working transmission as well as efficient processing and preservation. In the investigated scenario each single HL7 file contains textual information which corresponds to a single visit of a patient in the hospital. If the patient has examinations, which result in imaging files, then the results are stored in multiple DICOM files. Each DICOM file for a particular examination has the same unique series identifier and unique image identifier. The unique series identifier is stored in the DICOM tag SeriesInstanceUID (<0020,000E>), which by definition needs to be unique for each patient’s examination. The unique image identifier is stored in the AffectedSopInstanceUID DICOM tag (<0000,1001>), which by

[1] http://apps.who.int/classifications/icd10/browse/2010/en

[2] http://www.cdc.gov/nchs/icd/icd9.htm

[3] http://www.hl7.org/

[4] http://medical.nema.org/

definition needs to be unique for each image captured in a particular examination. Additionally, all of the imaging files contain accession number, which is a unique identifier of the referral. It is stored in the DICOM tag corresponding to AccessionNumber (<0008,0050>) and it is also included in the corresponding HL7 file, providing linkage between HL7 files and DICOM files. Thanks to such an approach DICOM files and HL7 files can be transmitted to the data center independently, with no ordering requirements (it does not matter what is sent first – DICOM files or HL7 files). Finally, because a single HL7 file represents a visit and a patient can have several visits in the hospital, the whole patient's medical history is stored in one or many HL7 files. The HL7 file, which concerns a certain patient, has a reference to the previous visit (the previous HL7 file), if there was such. Thanks to that the whole medical history of a patient's visits is also preserved in this context.

2.3 Data volume

In the context of data volume the largest datasets are produced during the examinations which results in medical imaging data. For instance, WCPT conducts 150 CT examinations each week and multiple RTG examinations as well. Each CT examination has a size of about 500 megabytes and may contain up to 2000 DICOM files. Thus for CT examinations alone, the WCPT hospital produces approximately 75 gigabytes of image data per week and 3,6 terabytes per year (stored in approx. 14,5mln files). All of the information is important in the context of patient treatment, scientific analysis and education at the medical universities.

Moreover, according to the Polish law, specific parts of the medical data must be stored by the hospital for at least 50 years in case of relapse of disease. In fact, most of the data considered in the framework of the SCAPE project are already subject to this law. In order to meet these demands, WCPT hospital expects from the SCAPE platform to securely transfer and store medical data with an easy and fast access interface to them. The estimated size of such data for only CT examinations is therefore 180TB and 725mln files. This estimation is only the lower limit because it assumes that the resolution of the images and the number of images per single examination will be the same as currently, and the number of CT machines will remain the same as well. It is highly probable that these factors will rapidly grow with time and the final size of data will be larger than the estimated values.

3 Workflow for the medical data handling

In order to store/access medical data at the data center or HPC center facilities it is required to follow a certain data flow procedure, so that all of the necessary information (e.g. linking between certain parts of data) is stored and available for further use. This section presents the vision of the data workflow and details about the storage procedure.

3.1 Medical data workflow architecture

Basis on identified challenges and WCPT requirements the medical data scenario has been designed as depicted in the Figure 1. The main idea is that there are two main actors in this context:

- WCPT (on the right side) with already existing software components (visible at the bottom) such as PACS (Picture Archiving and Communication System) and HIS (Hospital Information System). WCPT will also host the Hospital Archive System for handling medical data before transferring them to PSNC (Medical Data Center). It will also have tools for accessing the data directly from the data center archiving services.
- PSNC (on the left side) hosts the Medical Data Center, which leverages SCAPE-identified software components (e.g. Hadoop, HBase) as well as technologies used in the context of cloud and data center environment (e.g. OpenStack, dcm4che adjusted to cloud needs).

Basically, there are two core technical components of the whole medical data scenario setup. These are: Hospital Archive System (HAS) and Medical Data Center (MDC). The HAS system is to be running at WCPT and it is composed of the following components:

- Transfer Service (TS) – responsible for HAS-MDC communication. It acts as a client of the MDC. For example, it uses dcm4che tool[5] (dcmsnd command) to send anonymized DICOM files to MDC.
- DICOM anonymization component – used for anonymization of the medical data before transferring them to MDC via the TS.
- DICOM personalization component – used for personalization of the medical data before giving access to WCPT users. The idea is that whenever the WCPT user (granted permission to see sensitive personal data) wants to access a patient's history, the DICOM files coming from the MDC (which are anonymized) will be supplemented with the sensitive personal data. In this respect, the anonymization and personalization process is transparent for the WCPT users.
- Encryption component – used for encrypting those parts of medical data which are the most sensitive, to send to the MDC. Currently this is the identifier of the patient's visit at the hospital.
- HIS-MDC – component which keeps references between HAS data identifiers and MDC data identifiers.

The Medical Data Center is running at PSNC data center and HPC facilities and covers the following components:

- HDFS PACS – a server application, which is a customized version of the Picture Archiving and Communication System based on dcm4che toolset. The tool was modified in a way that it can store the data (DICOM files) in HDFS (for processing purposes) and cloud storage (for archiving purposes).

[5] http://www.dcm4che.org/

- HDFS HL7 – a server application, which is a HL7 gateway. It receives textual data encoded in XML and stores them on HDFS (for further processing) and cloud storage (for archiving purposes).
- WWW – it is a web based component for accessing anonymized data in a simple and easy way. The main idea behind that is to allow interested parties to access the data in a way that it is possible to view special cases for particular diseases.
- A set of tools which are treated as backend technologies, such as OpenStack cloud environment, Cloudera components (Hadoop and HBase) as well as dcm4che toolset for handling medical data.

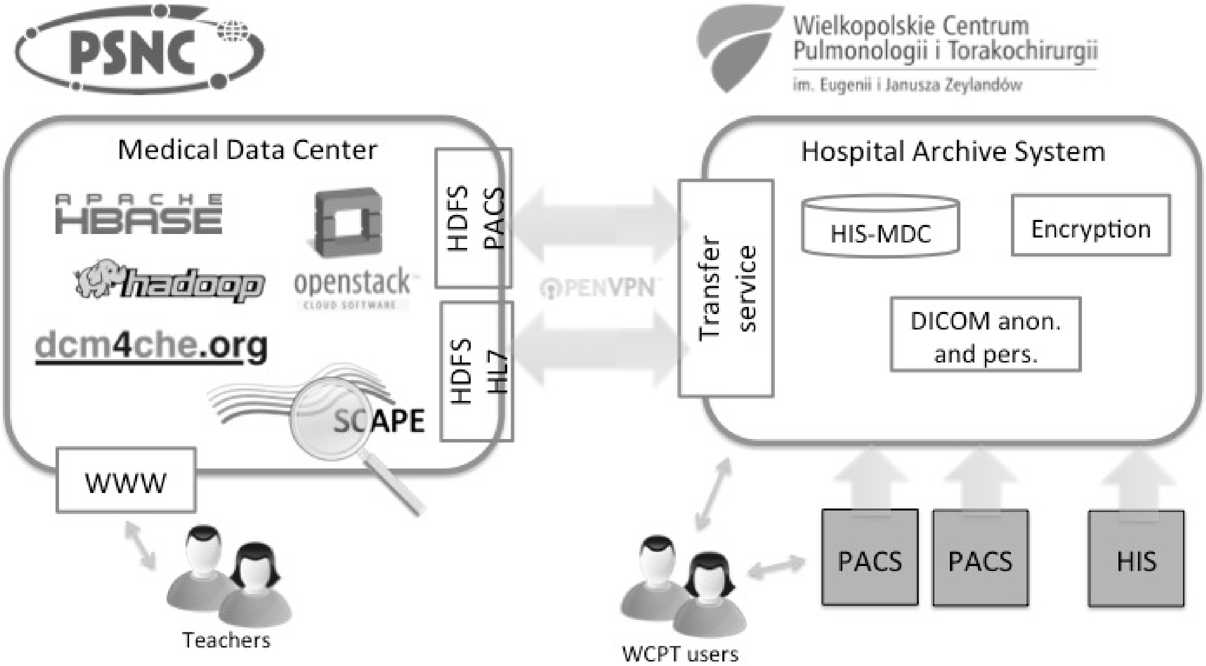

Figure 1 Medical data scenario - WCPT and PSNC technical components and communication

3.2 Ingestion procedure

Data produced at the WCPT hospital needs to be ingested into the PSNC's MDC. The ingestion procedure is the same for every piece of data such as DICOM files or HL7 files. It is depicted in Figure 2 in the context of the medical data scenario from Figure 1.

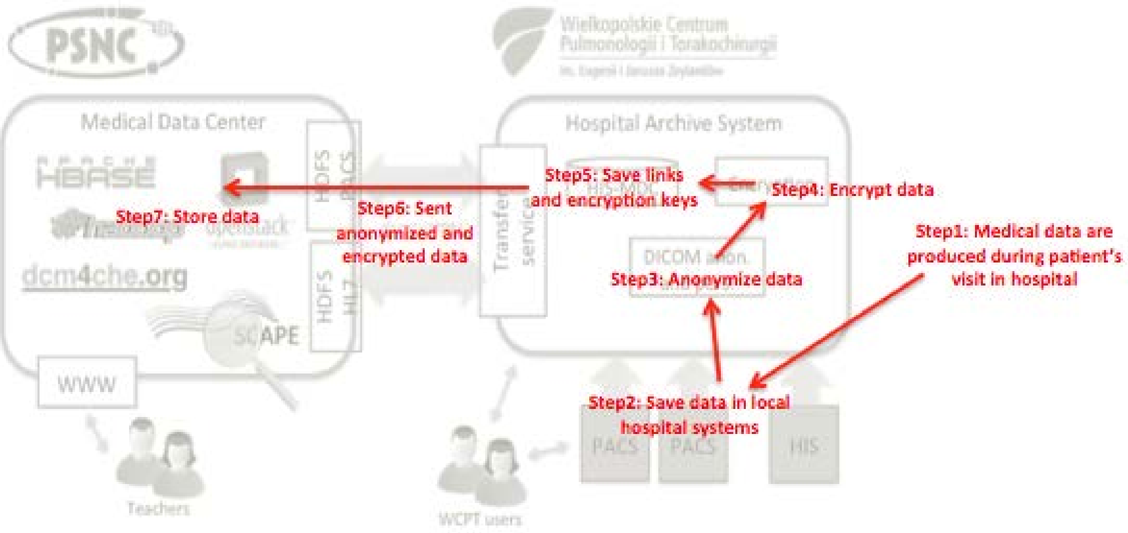
PSNC
Wielkopolskie Centrum Pulmonologii i Torakochirurgii
Medical Data Center
HBASE
dcm4che.org
SCAPE
HDFS
PACS
HDFS
HL7
WWW
Teachers
Transfer service
Hospital Archive System
Step5: Save links and encryption keys
Step4: Encrypt data
Step1: Medical data are produced during patient's visit in hospital
Step7: Store data
Step6: Sent anonymized and encrypted data
DICOM anon.
Step3: Anonymize data
Step2: Save data in local hospital systems
PACS
PACS
HIS
WCPT users

Figure 2 Ingestion procedurę steps in the context of medical data scenario

The following steps are undertaken in the context of medical data ingest:

- Step1: medical data is produced during the patient's visit at the hospital.
- Step2: medical data is saved in the local hospital systems (in HIS or PACS).
- Step3: medical data is anonymized with the dicom-anonymizer tool (see section 4.3-Anonymization tool for details).
- Step4: medical data identifiers are encrypted.
- Step5: encryption keys and links to personal sensitive data are stored to enable reverse anonymization and/or encryption in the future, if necessary. This step is crucial for linking anonymized and encrypted data sent to the data center with medical data stored in the local hospital systems.
- Step6: anonymized and encrypted data are sent to remote data center (see sections 4.1 and 4.2).
- Step7: medical data is stored at the data center.

The ingestion procedure is designed in such a way that it is easy to further access the data as defined in the other scenarios (e.g. educational and scientific). The core rules for storing the data are described in the following sections.

3.3 Data storage structures

Data storage mechanism is based on the Cloudera toolset[6], more specifically it is based on HDFS file system[7] and HBase component[8]. Data is stored in directories on HDFS. HBase is used to provide quick access and linkage to the data required in the medical data access scenarios.

3.3.1 Storage structure

The first level in the HDFS directories tree is dedicated for organizations like WCPT hospital. It means that data from particular organization (e.g. from WCPT hospital) will be saved in a dedicated directory on HDFS. Therefore different organizations will have different dedicated directories on HDFS. Because DICOM tag SeriesInstanceUID is unique for each examination and

[6] http://www.cloudera.com/

[7] http://hadoop.apache.org/

[8] http://hbase.apache.org/

AffectedSopInstanceUID is unique for each image in particular examination, those have been used to name directories and file names on HDFS. In the directory of a certain organization DICOM files are grouped by the examination series identified by the SeriesInstanceUID DICOM tag (<0020,000E>). Because each DICOM file coming from the same examination has the same SeriesInstanceUID, the files from one examination will be stored in the same directory. Each DICOM file (from this series) is then saved in this directory with the name corresponding to its unique identifier stored in the AffectedSopInstanceUID DICOM tag (<0000,1001>). Finally, full path to an exemplary DICOM file in HDFS is as follows:

/{organisation_name}/{series_instance_uid}/{affected_sop_instance_uid}

where:

- *{organisation_name}* is the name of an organization, e.g. WCPT,
- *{series_instance_uid}* is the unique identifier of the examination series
- *{affected_sop_instance_uid}* is unique identifier of the DICOM file within particular series.

3.3.2 Linking DICOM and HL7 files

HBase is used to link together stored DICOM files with HL7 data and also to provide fast response for queries coming from the user interface components of the MDC. In the context of ingest process HBase tables contain mapping between accession number from the DICOM files and from the HL7 files as well as information on links between examination and corresponding DICOM files. There are two tables defined in HBASE for this named **accession_numbers** (see Table 1) and **series_instance_uids** (see Table 2).

key	hospital_name_a:1	hospital_name_a:2	hospital_name_b:1	…
3206/KT/2011	1.2.840.119.2.278.1	1.2.840.119.2.278.2		
3207/KT/2011			1.2.840.119.2.278.3	
…	…	…	…	…

Table 1 Accession numbers table in HBase

The table named **accession_numbers** (example presented in the Table 1) is for connecting HL7 files with all related examination results (which are in form of DICOM files). In the key column of this table there are accession numbers from the HL7 files. The family column with hospital names contains identifiers of related examinations (InstanceSeriesUIDs DICOM tag). Index number after the hospital name in the column indicates the ordering number of the examination related to a given HL7 file. Therefore having an accession number it is possible to obtain all of the related examination identifiers (in scope of one patient's visit).

Key	hospital_name_a:1	hospital_name_a:2	hospital_name_b:1	…
1.2.840.119.2.278.1	1.2.3644.893.139	1.2.3644.893.140		
1.2.840.119.2.278.2			1.2.3644.893.150	
…	…	…	…	…

Table 2 Series instanceuids table in HBase

The table named **series_instance_uids** is for connecting given examination identifier with all the DICOM files created as a result of this examination. Therefore the key column of this table contains identifiers of examination (InstanceSeriesUID) and the values contains family column named hospital_name, which contains AffectedSOPInstanceUID (single image identifiers). Index after the hospital name indicates ordering number of the image in the examination.

4 Communication and anonymization tools

Two core components are used in all SCAPE medical scenarios - the Hospital Archive System (HAS) component located in WCPT hospital and Medical Data Center (MDC) located in PSNC. In order to ensure communication between both components several applications were implemented or existing open source tools were adapted. Figure 3 depicts the location of particular tools and communication interfaces with respect to these core components, which are described in the remainder of this section.

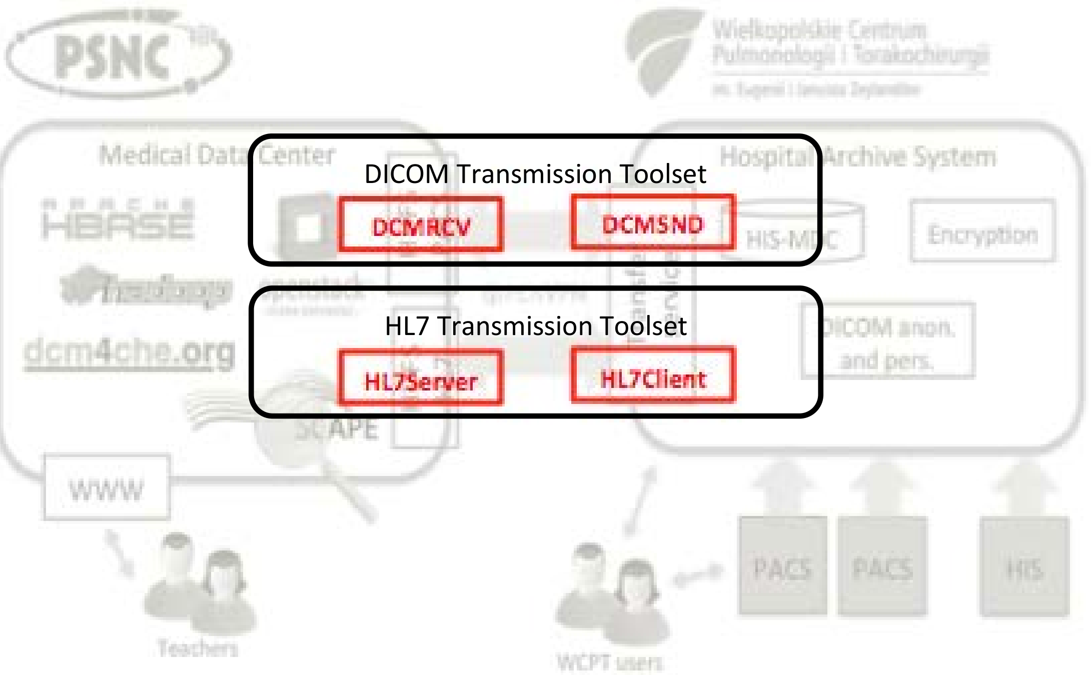

Figure 3 Tools and communication interfaces in the medical scenario context

4.1 DICOM transmission toolset

DICOM transmission toolset consists of the server (DCMRCV) and client application (DCMSND).

4.1.1 DCMSND

DCMSND is part of dcm4che[9] package. This application acts as a DICOM SCU (Service Class User - according to DICOM standard) to send DICOM objects to a DICOM SCP (Service Class Provider User - according to DICOM standard). Below you can find the most important options in the context of the medical data scenario and exemplary invocation of the tool. For full documentation please refer to http://www.dcm4che.org/.

Usage:
dcmsnd [Options] <aet>[@<host>[:<port>]] <file>|<directory>...

Options:

-h,--help	*print all options*
-keystore<file>	*file path to keystore*
-keystorepw<password>	*password for keystore file*

[9] http://www.dcm4che.org/

-tls<NULL|3DES|AES>	*enable TLS connection*
-truststore<file>	*file path to truststore*
-truststorepw<password>	*password for truststore file*

Example:

```
dcmsnd DCM@mdc.scape.psnc.pl:7183 directory -keystorekeystores/wcpit-keystore.jks -
keystorepwpassword_to_keystore -truststorekeystores/wcpit-truststore.jks -
truststorepwpassword_to_truststore -tls AES
```

DCMSND loads DICOM Object(s) from the specified DICOM file(s) or a directory structure and sends them to the specified remote AE (Application Entity - according to DICOM standard). If a directory is specified, all DICOM objects in files under that directory and its sub-directories are sent. If <port> is not specified, DICOM default port 104 is assumed. If no <host> is specified, localhost is assumed.

4.1.2 DCMRCV

DCMRCV is a modified version of the DCMRCV from dcm4che package. DCMRCV has been extended with a new feature to save DICOM files into HDFS file system and cloud storage for backup copy. Below you can find the exemplary invocation of the tool with most important usage options (from the point of view of the medical data scenario). For full documentation please refer to http://www.dcm4che.org/. The source code of the modified tool can be found on PSNC's git repository: https://git.man.poznan.pl/stash/projects/SCAP/repos/dicom/browse/dcm4che.

Usage:

```
dcmrcv [Options] [<aet>[@<ip>]:]<port>
```

Options:

-dest<dir>	*store received objects into files in specified directory <dir>.*
-h,--help	*print all options*
-keystore<file>	*file path to keystore*
-keystorepw<password>	*password for keystore file*
-tls<NULL|3DES|AES>	*enable TLS connection*
-truststore<file>	*file path to truststore file*
-truststorepw<password>	*password for truststore file*

Example:

```
dcmrcv DCM@:7183 -dest . -keystorekeystores/psnc-keystore.jks -
keystorepwpassword_to_keystore -truststorekeystores/psnc-truststore.jks -
truststorepwpassword_to_truststore -tls AES
```

Executing the DCMRCV utility will run a DICOM Server listening on the specified <port> for incoming requests. If no local IP address of the network interface is specified, connections on any/all local addresses are accepted. If <aet> is specified, only requests with a matching called AE title will be accepted.

4.2 HL7 transmission toolset

HL7 transmission toolset consists of the server application (HL7SRV) and client application. It is used to transfer HL7 files between WCPT and PSNC. The tools are described in the following part of this section.

4.2.1 HL7SND

HL7SND tool sends HL7 3.x messages to HL7 gateway (server). It uses HTTP protocol as specified in the HL7 documentation. The source code of the tool can be found on the PSNC's git repository: https://git.man.poznan.pl/stash/projects/SCAP/repos/hl7/browse/hl7sender.

Usage:

```
hl7snd [Options]
```

Options:

-url<url:port>	*specify url and port to HL7 Server*
-file<HL7file>	*file path to HL7 message stored in xml*
-keystore<file>	*file path to keystore*
-keystorepw<password>	*password for keystore file*
-truststore<file>	*file path to truststore*
-truststorepw<password>	*password for truststore file*

Example:

```
hl7snd -url https://mdc.scape.psnc.pl:9183 -file hl7_file -keystorekeystores/wcpit-keystore.jks
-keystorepwpassword_to_keystore -truststorekeystores/wcpit-truststore.jks -truststorepw
password_to_truststore
```

4.2.2 HL7RCV

HL7RCV stores HL7 files in HDFS file system as described in the previous chapter. It acts as a HTTP server and listens for incoming requests to store HL7 data.

Usage:

```
dcmrcv [Options]
```

Options:

-port<port>	*specify port*
-keystore<file>	*file path to keystore*
-keystorepw<password>	*password for keystore file*
-truststore<file>	*file path to truststore*
-truststorepw<password>	*password for truststore file*

Example:

```
hl7rec -port 9183 -keystorekeystores/psnc-keystore.jks -keystorepwpassword_to_keystore -
truststorekeystores/psnc-truststore.jks -truststorepwpassword_to_truststore
```

HL7RCV utility will run an HL7 gateway (server), which listens for incoming requests on the specified <port>.

4.3 Anonymization tool

As stated in previous sections WCPT policy requires existence of anonymization and personalization functions which can be executed on the data sent to or received from MDC. In order to fulfil this demand the DicomAnonymizer tool was developed to provide anonymization and personalization functionality. Using the DicomAnonymizer WCPT is able to process (anonymize) each DICOM file before sending it to PSNC. A personalization feature is needed when retrieving data from PSNC –

WCPT components can then add necessary personal data before sending them to interested WCPT user (so that it has full patient information). DicomAnonymizer is written in Java programing language, and can therefore be executed on any operating system which is able to run a Java Virtual Machine (e.g. Windows, Linux, MacOS). The source code has been published on the github repository: https://github.com/openplanets/dicom-anonimizer.

DICOM anonymization tools already exist, the list of most well-known tools is presented below.

- DICOM Anonymizer[10] - this tool replaces the patient names in all the DICOM files in a folder (and sub-folders) with other specified values. The tool also contains an option to erase institutional information. It works either as a batch or GUI application. There is no possibility to define which tags (DICOM attributes) should be removed. The tool supports anonymization process only and there is no option to reverse it (to personalize).
- DicomCleaner[11] - it is a free open source tool with a user interface only (no command line interface is available). The user is given the control over what to remove and replace (many options in interface) but there is no option to predefine a list of tags to remove. There is also no possibility to reverse anonymization (personalize).
- DICOM Anonymizer[12]- this tool allows dragging and dropping of individual DICOM files or one or more folders to recursively search for DICOM files to be anonymized. List of tags to anonymize can be specified in configuration file, which by default corresponds with DICOM Standard Attribute Level Confidentiality (Supplement 55). The tool is a GUI application only (no command line interface is available).
- DICOMAnonymize[13] - this tool is a freeware Windows application with user interface only. It supports anonymization of the patient name only.
- DVTk DICOM Anonymizer[14]- in the DVTk DICOM Anonymizer user can define what type of anonymization should happen (basic or full), but there is no options to prepare own configuration of anonymization process – only those two predefined options are available. The tool can anonymize individual DICOM files or a folder with many DICOM files. There is no possibility to personalize DICOM files.

Unfortunately because none of the already existing tools provides whole set of functions needed by WCPT scenario it was decided to build a new tool which will provide command line interface for configurable anonymization and personalization process of the given DICOM file(s). The new tool called DicomAnonymizer is based on the dcm4che toolset, which is a collection of open source applications and tools related to medicine. One part of dcm4che is the DICOM standard implementation, including API for handling/modifying DICOM files. dcm4che is implemented in Java programming language and runs on Java Development Kit 1.4 or newer.

4.3.1 Configuration

By default DicomAnonymizer tool is configured according to “Supplement 55: Attribute Level Confidentiality (including De-identification)” released by National Electrical Manufacturers

[10] http://sourceforge.net/projects/dicomanonymizer
[11] http://www.dclunie.com/pixelmed/software/webstart/DicomCleanerUsage.html
[12] http://doradiology.com/DICOManonymizer/index.html
[13] http://www.image-systems.biz/products/free-dicom-tools/dicomanonymize.html
[14] http://dicom.dvtk.org/index.php

Association in 2002[15]. The document describes and defines which DICOM attributes should be removed from DICOM file to ensure Basic Application Level Confidentiality Profile (basic level of anonymity). It is important to note that attributes listed in the Supplement 55 may not be sufficient to guarantee confidentiality of patient identity, in particular, identifying information may be contained in DICOM Private Attributes and inside data (e.g. saved in the image itself). The default configuration can be adjusted to specific needs of the user, as the configuration file contains information about the DICOM attributes that will be removed during anonymization. In the case of WCPT medical scenario it is assumed that only sensitive personal information will be removed during anonymization, namely the following DICOM tags:

- PerformingPhysiciansName(0008,1050)
- NameofPhysicianReadingStudy(0008,1060)
- OperatorsName(0008,1070)
- PatientsName(0010,0010)
- PatientID(0010,0020)
- OtherPatientIds(0010,1000)
- OtherPatientNames(0010,1001)
- ReferringPhysiciansName(0008,0090)
- ReferringPhysiciansAddress(0008,0092)
- ReferringPhysiciansTelephoneNumbers(0008,0094)
- PerformingPhysiciansName(0008,1050)
- NameofPhysicianReadingStudy(0008,1060)
- OperatorsName(0008,1070)
- PatientsName(0010,0010)
- PatientID(0010,0020)
- OtherPatientIds(0010,1000)=1
- OtherPatientNames(0010,1001)=1

4.3.2 Execution

As previously mentioned DicomAnonymizer can anonymize and personalize DICOM files. In order to execute anonymization (depicted on Figure 5) function on specific DICOM file the user needs to run the following command in the OS shell:

```
java –jar <DicomAnonymizer_jar_file> -anonymize <dicom_file> <anonymized_tag_file>
```

where

- *<DicomAnonymizer_jar_file>*indicates the path to DicomAnonymizer jar file
- *<dicom_file>*indicates the path to the DICOM file which needs to be anonymized. It is important to note that this file will be modified (no copying is done by the anonymization tool).
- *<anonymized_tag_file>* is the name of text file to which all anonymized data removed from the DICOM file will be outputted

Attributes to be removed from the given DICOM file are specified in the configuration file which contains key-value pairs: tag and its corresponding flag in format:

[15] http://medical.nema.org/Dicom/supps/sup55_03.pdf

<tag>=<flag>

where *<tag>* is a human understandable name of the DICOM attribute with its numeric representation in brackets and *<flag>* is one of two values: 0 or 1. Value 1 indicates that corresponding tag will be removed during anonymization and value 0 indicates that corresponding tag will not be removed. If a specific tag is not specified in the configuration file value 0 of its corresponding flag is default. Below you will find an example of several lines of such configuration file.

```
…
PatientsName(0010,0010)=1
PatientID(0010,0020)=1
…
PatientsBirthDate(0010,0030)=1
PatientsSex(0010,0040)=0
…
```

Figure 4 Several example lines of configuration file

The output text file resulting from the anonymization process is created in the same directory where the anonymized DICOM file is located. The output file contains all the information that is removed from the DICOM file during anonymization. The anonymization process, including input and output files, is depicted on Figure 5. The input files are the DICOM file to be anonymized and configuration file, while the outputs are anonymized DICOM file and output text file with information (DICOM attributes and their values) removed from the DICOM file.

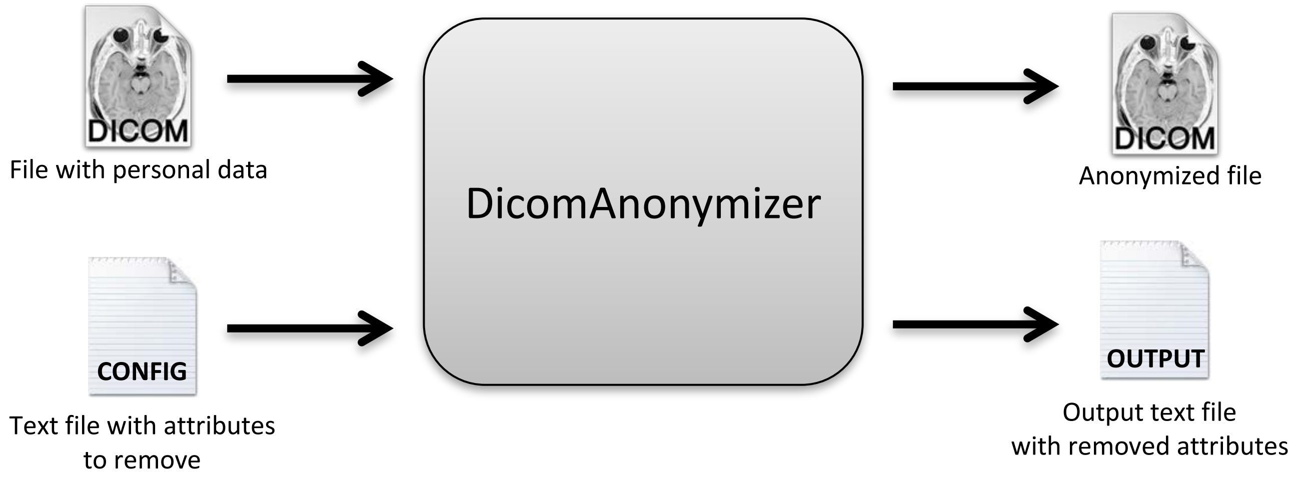
DICOM
File with personal data
CONFIG
Text file with attributes to remove
DicomAnonymizer
DICOM
Anonymized file
OUTPUT
Output text file with removed attributes

Figure 5 Anonymization tool - input and output files

In order to execute personalization function on specific DICOM file the user needs to run the following command in the OS shell:

java –jar <DicomAnonymizer_jar_file> -personalize <dicom_file><anonymized_tag_file>

where

- *<DicomAnonymizer_jar_file>* indicates the path to DicomAnonymizer jar file

- *<dicom_file>* indicates the path to the DICOM file which needs to be personalized. It is important to note that this file will be modified (no copying is done by the tool).
- *<anonymized_tag_file>* is the name of text file from which personal data will be read and added to the resulting DICOM file. The format of this file is identical as the *<anonymized_tag_file>* of the anonymization process.

The personalization process is depicted on the Figure 6. The input files include anonymized DICOM file and text file with DICOM attributes to be added to the DICOM file, while the outputs include modified DICOM file.

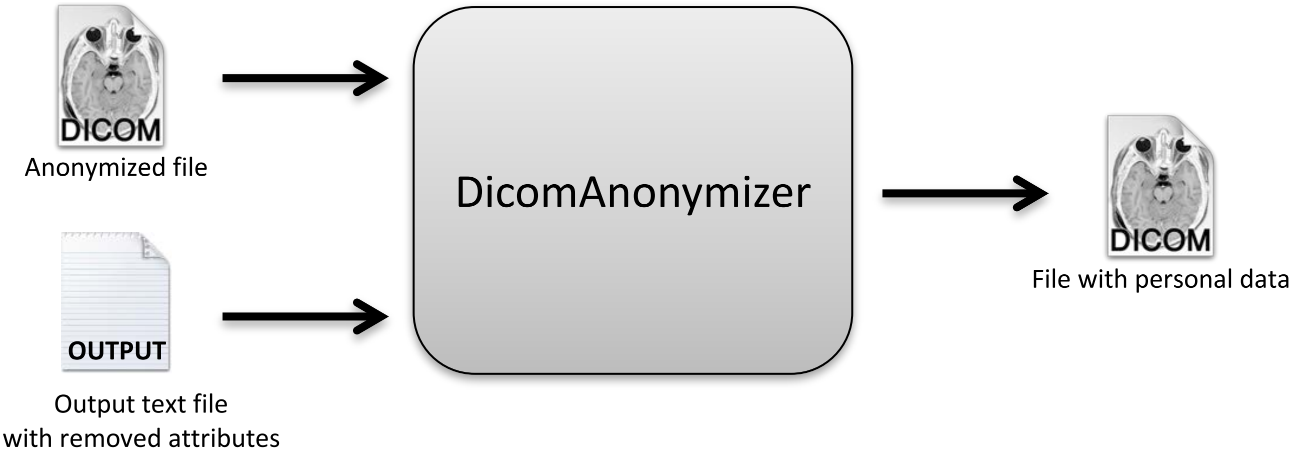

Figure 6 Personalisation tool - input and output files

Moreover, both personalization and anonymization can be executed not only on a file level, but also on the directory level. The following commands are intended to work on directory – anonymization and personalization is executed for each file in the directory and subdirectories.

- *-anonymize_dir<directory>* - anonymizes all DICOM files in a specified directory and its subdirectories plus saves anonymized tag files for each DICOM file. The name of the tag file is the same as anonymized DICOM except file extension (tag file has txt extension).
- *-personalize_dir<directory>* - personalizes all DICOM files in specified directory and its subdirectories. It is assumed that tag files are in the same directory as DICOM files plus they have the same name (except extension) as the DICOM file to which personal information should be added.

Beside those commands there is also *–help* argument which displays help (the list of available arguments for the tool).

5 Summary

Anonymized ingestion is a multi-step process of storing the data not only on the HDFS file system, but also for storing supporting information in the HBase component. The whole process is composed of several steps, including anonymization of the DICOM files and production of the anonymized HL7 files. The data transfer itself is done using dedicated tools developed within the SCAPE project and for the purposes of the medical data scenarios. The two core toolsets are necessary to transfer data from the WCPT hospital to PSNC storage facilities, namely DICOM toolset and HL7 toolset. Each of them contains pairs composed of the server and client application. The server is deployed on the PSNC side, and the client is used to copy the data from WCPT to PSNC. For the needs of the further work in the SCAPE project it is crucial to use HBase component, as it stores the information that allow for fast and easy retrieval of necessary data in the context of educational, access or scientific scenario. The ingestion into HDFS storage process has been tested on the PSNC's development cluster, with all services for integrating WCPT's remote working environment available. It was possible to upload multiple GB of data. Further work on the anonymization and ingestion process will cover activities related to necessary improvements such as bug-fixes or documentation.