Title:	*Final Deployment*
Author:	*Work Package 8 ("Application 2: organ transplant management")*
Editor:	*Javier Vázquez-Salceda and Sergio Álvarez (UPC), László Zs. Varga and Kifor Tamás (MTA SZTAKI)*
Reviewers:	*All project partners involved*
Identifier:	*D8.3.1*
Type:	*Deliverable*
Version:	*V1.0*
Date:	*November 30th , 2006*
Status:	*Final*
Class:	*Public*

Summary

This document describes the final deployment of the OTM demonstration system in months 12-24 of the project. OTM runs in a SOA environment and uses several services developed by different partners of the EU PROVENANCE project. Some of the services represent organizations from real life such as transplant authority or hospitals.

Copyright © 2004 – 2005 by the PROVENANCE Consortium
The PROVENANCE project receives research funding from the European Community's Sixth Framework Programme

Members of the PROVENANCE Consortium:

IBM United Kingdom Limited	United Kingdom
University of Southampton	United Kingdom
University of Wales, Cardiff	United Kingdom
Deutsches Zentrum für Luft- und Raumfahrt e. V.	Germany
Universitat Politecnica de Catalunya	Spain
Magyar Tudomanyos Akademia Szamitastechnikai es Automatizalasi Kutato Intezet	Hungary

Foreword

This document describes the final deployment of the demonstration applications in Work Package 8 *"Application 2: organ transplant management"*. The document provides:

- short description of the OTM application and its components,
- detailed description of the final deployment of the OTM application,
- user manual of the OTM application.

The primary audience of this document includes: A) Grid computing practitioners seeking to understand how Provenance technologies might be applied and, B) Information technology practitioners in the health care domain interested in applying Provenance to their own medical systems.

Table of Contents

List of Acronyms

Acronym	*Description*
AR	Abstract level capability user requirements
CR	Constraint user requirements
CSL	Client Side Library
DNI	Spanish National Identity Number
EHCR	Electronic Health Care Record
EHCRS	Electronic Health Care Record Store
GMPID	Global Medical Patient Identifier
GP	General Practitioner
GUI	Graphical User Interface.
HCR	Health Care Record
HLA	Human Leukocyte Antigens test
IT	Information Technology
LMPID	Local Medical Patient Identifier
OCATT	Organització Catalana de Transplantaments (Catalan Transplant Organisation)
ONT	Organización Nacional de Transplantes (Spanish National Transplant Organisation)
OTA	Organ Transplant Authority
OTM	Organ Transplant Management
PDF	Portable Document Format, Adobe Inc.
PID	Patient Identifier
PS	Provenance Store
SR	Software Requirements
TR	Technical level capability user requirements

1 Introduction

This document describes the final deployment of the demonstration applications in Work Package 8 *"Application 2: organ transplant management"*. This document describes the components of the OTM demonstrator:

- the core OTM application,
- the EHCR Store application (EHCRS),
- the OTM GUI,
- the test OTM client of the EHCR Store,
- the EHCR Auth (the authorization service),
- databases of the core OTM application, EHCR Store and EHCR Auth.

The primary audience of this document includes: A) Grid computing practitioners seeking to understand how Provenance technologies might be applied and, B) Information technology practitioners in the health care domain interested in applying Provenance to their own medical systems.

1.1 Purpose of the Document and Overview

The purpose of the document is to describe the final deployment of the demonstration application in the organ transplant scenario. This application is composed by two inter-connected applications: the Organ Transplant Management application (OTM) and the Electronic Health Care Record Store application (EHCRS).

In order to achieve this goals, the document aims to:

- Summarize components of the OTM and EHCRS applications
- Introduce actors involved in the demonstration
- Give the access points of all the components involved in the demonstration, including access points of components of provenance service and provenance tool as well

This document does not aim to:

- Describe the OTM and EHCRS applications in details.
- Describe provenance service and provenance tool used in the demonstration.
- Describe the demonstration of the applications.

1.2 Links to other Provenance Documents

The contents of this document are based on the following existing Provenance project documents:

- Requirements expressed for the OTM application in the WP2 Requirements deliverables D2.1.1 and D2.2.1.,
- The Provenance architecture document D3.1.1,
- Project internal note on "Representing Provenance in the OTM application" [Miles05],

- Deliverable D8.1.1. "Specification of mapping to provenance architecture, and domain specific provenance handling ",
- Deliverable D8.2.1. "Evaluation Report",
- Deliverable D9.3.3a "Client Side Library Design and Implementation",
- Deliverable D6.1.1 "Tools Description Document.

Further supporting documents are provided in the references section.

1.3 Short description of the OTM and EHCRS applications

Electronic systems for transplant management should cover 2 aspects:

1. *Transplantation Management*: information systems used by medical staff during the process of a transplant incident (a single patient receiving an organ or tissue) to access existing case or background data, share it with colleagues, carry out matchmaking and/or otherwise provide decision support.

2. *Medical Record management*: the storage, access and modification of medical patient care records for patients in a given geographic region. Gathering, access and modification of such data is regulated by European, national and regional laws and forms an underlying information system for any treatment process management system.

Therefore the demonstration application developed in Work Package 8 is actually composed of two interconnected applications:
- transplant management is provided by the Organ Transplant Management (OTM) application,
- medical record management is provided by the Electronic Health Care Record (EHCR) application.

Although in this document we will refer to each of these applications separately, both function together and can be seen from the user perspective as a single application, with the OTM application directly accessing and making use of the EHCR functionality.

Following subsections present a short description of each of the applications.

1.3.1 Transplantation Management and Post-processing: The OTM Application

The Organ Transplant Management (OTM) Application aims to speed up the allocation process of solid organs to improve graft survival rates. Its policy implements the Spanish guidelines for organ and tissue procurement and Spanish regulations for allocation, as Spain is world leader in the area, followed as a model by other countries. OTM uses standard web service technology and has been adapted to be provenance-aware, by interacting with the provenance stores in order to keep track of the distributed execution of the allocation process for audit purposes.

Figure 1 summarizes the different administrative domains (solid boxes) and units (dashed boxes) that are modeled in the OTM application. Each of these interact with each other through Web Service interfaces (circles) that send or receive messages. The Organ Transplant Authority (OTA) is an administrative domain with no internal units. In a transplantation management scenario, one or more hospital units may be involved: the hospital transplant unit, one or several units that provide laboratory tests and the unit that is responsible for the patient records (which will use the EHCR application services, see section 1.3.2). The diagram also shows some of the data stores that are involved: apart of the patient records, these include stores for the transplant units and the OTA

recipient waiting lists (WL). Hospitals that are the origin of a donation also keep records of the donations performed, while hospitals that are recipients of the donation may include such information in the recipient's patient record. The OTA has its own records of each donation, stored case by case.

More specifically, Figure 1 also shows an example of a typical interaction between these actors: a transplant management scenario starts with a potential donor in Hospital A's transplant unit. In order to evaluate the donor, this unit may request the patient records inside the hospital and order a number of tests, some of them to internal laboratory units and others to some specialized external laboratories. Once the donor is evaluated and, if valid, the transplant unit contacts the OTA, which sends first the offer to hospital C. As the transplant unit in hospital C rejects the donation, the OTA sends the offer to hospital B, which has a potential recipient for the organ offer (as in the case of Hospital A, all the medical data needed for the recipient was previously collected by hospital B by interacting with the ECHR application and the testing laboratories). During extraction and implantation, direct communication between hospital A and hospital B and also between the OTA and the hospitals occurs.

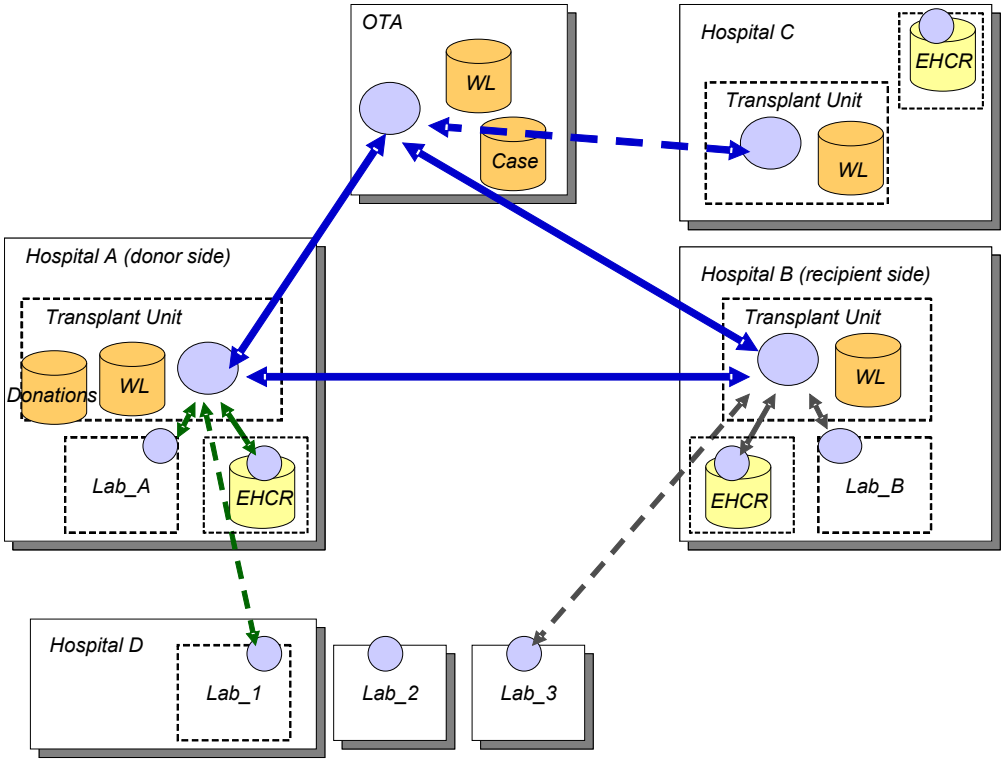


Figure 1 -- Actors in the OTM application.

By transforming OTM into a *provenance-aware* application, OTM is augmented with a capability to produce at run-time an explicit representation of the process actually taking place (examples can be seen in section A.14). Such representation can be then queried and analysed in order to extract valuable information to validate, e.g., the decisions taken in a given case, or to make an audit of the system over a period of time.

1.3.2 The EHCRS Application

The Electronic Health Care Record System (EHCRS) provides a way to manage electronic health records distributed in different institutions. The architecture provides the structures to build a part of or the entire patient's healthcare record drawn from any number of heterogeneous databases systems in order to exchange it with other healthcare information systems.

The EHCRS architecture has two external interfaces:

1. a Web Service that receives and sends messages (following ENV13606 pre-standard format [ENV13606]) for remote medical applications; and
2. a Web Service for local medical applications that can be used to access the EHCR store directly. The application also uses an authentication Web Service to authorize request messages from remote health care parties.

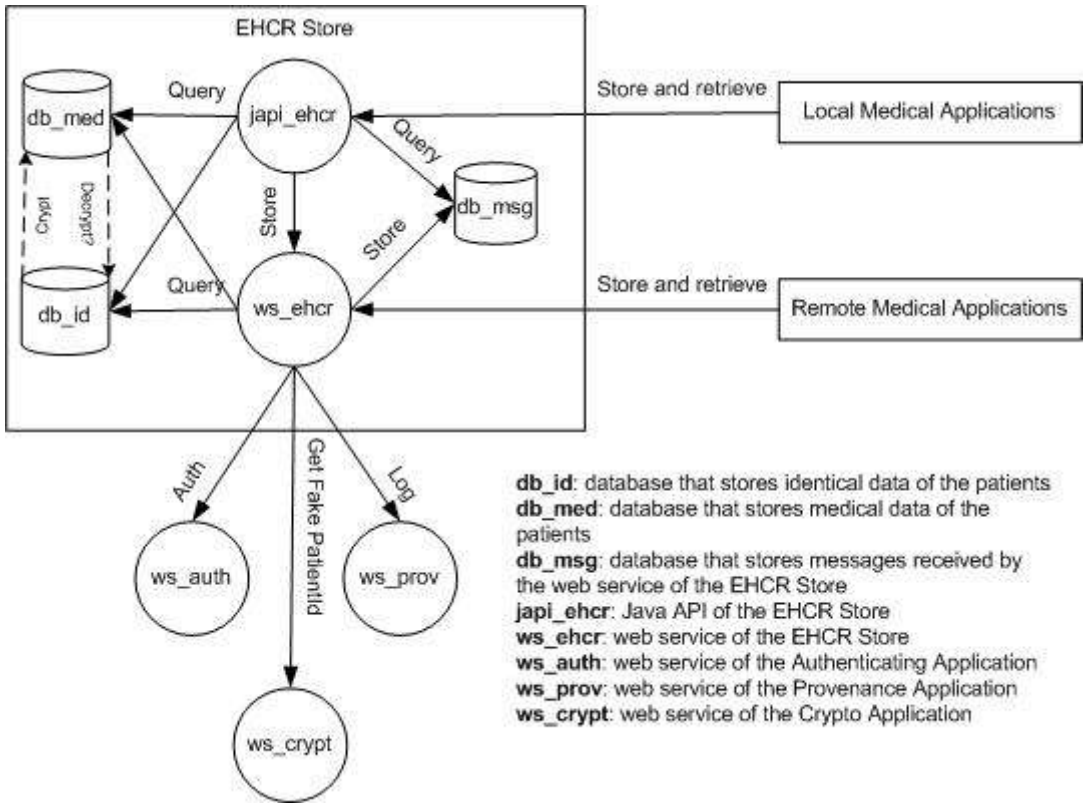


Figure 2 – EHCRS application.[1]

1 In the case of the OTM application, OTM services may access to EHCR stores as local or remote medical applications, depending if there is local network access between the OTM service and the EHCR store or not.

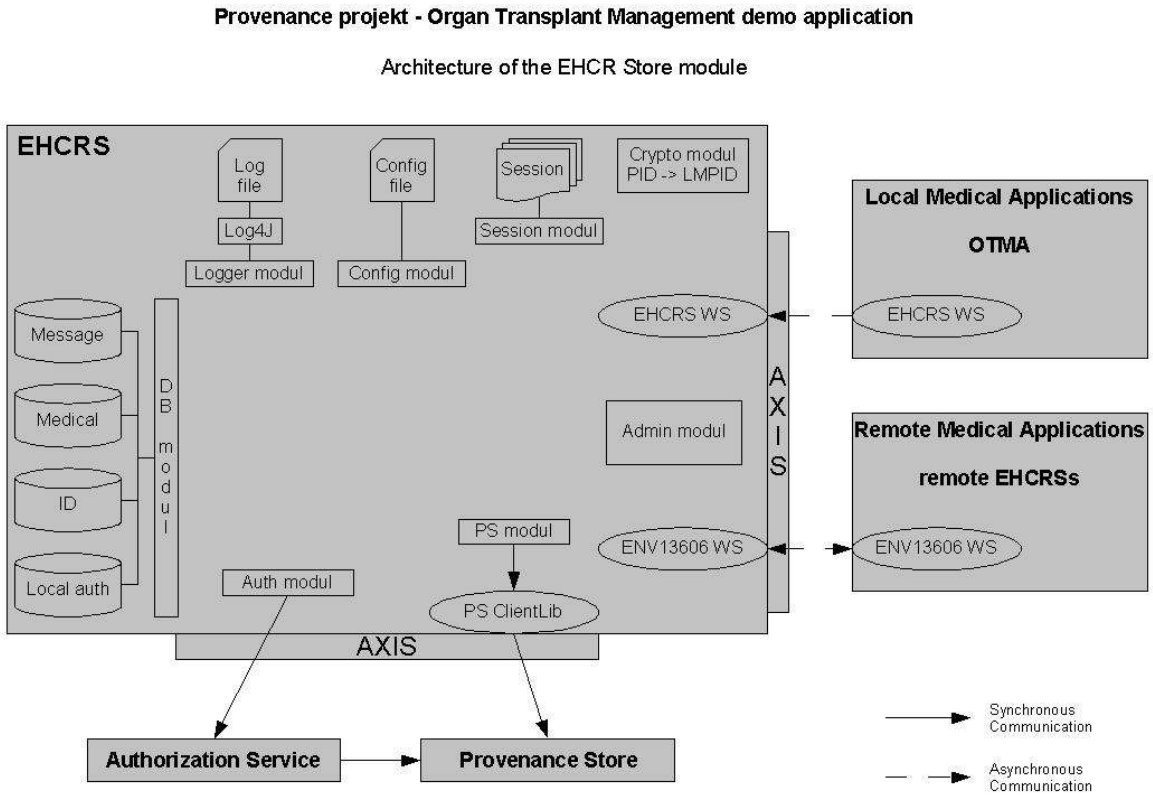


Figure 3 – EHCRS architecture.[2]

The EHCRS application is used by the OTM application as its primary store of patient care data. But it is important to note that the EHCRS application was intended to be not only the application to store medical records for the needs of the OTM application, but a generic system for storing and collating health care records across multiple health care providers, usable by other health care grid applications.

Making the EHCRS application *provenance-aware* provides a way to have a unified view of a patient's medical record with its provenance (i.e. to connect each part of the medical record with the processes in the real world that originated it and/or the individuals, teams or units responsible for each piece of data).

1.4 Summary

In summary, the OTM demonstrator is divided into two parts: an underlying health care record management element (EHCRS) and the OTM application itself. These two applications are seen by the user as a single application that provides both the transplantation management and the medical record management integrated in OTM's user interface.

2 In the case of the OTM application, OTM services may access to EHCR stores as local or remote medical applications, depending if there is local network access between the OTM service and the EHCR store or not.

2 Description of the OTM Demonstrator

This section provides a brief technical description about the OTM Demonstrator and its main components. Section 2.1 explains technical details about the OTM application before applying Provenance. Section 2.2 describes the Provenance interface used in the adaptation of the OTM Application. Section 2.3 introduces the EHCRS application, the data management part of OTM, which follows ENV13606 European pre-standard communicating with external medical institutions. Section 2.4 is similar to 2.3, but referring to the adaptation of the EHCRS application. Section 2.5 explains how the OTM and EHCRS applications interact between themselves. Finally, section 2.6 shows how the OTM application uses the Provenance Tool.

2.1 The OTM Application

The OTM Application is a distributed version of a Organ Transplant Management application being developed for the CARREL@FIS project. As mentioned in section 1.3.1, each organisational unit (a transplant unit in a hospital, an OTA, the testing laboratories) is a component of the OTM application composed by:

- a core subsystem that controls the event workflow for the component and the interactions with other components (such as the Provenance Store and the EHCRS application),
- a communication subsystem with other components, based in webservice communication.

The components in the OTM application receive and send interaction messages between themselves, and process the requests according to their programmed behaviours, using their core subsystems to control the execution workflow.

The OTM Application has been developed in Java 1.5 and the modular components are deployed as web services. The user interface of the OTM Application is the OTM GUI, a web application developed in Java 1.5 using the Google Web Toolkit. This application serves DHTML web pages that allow the user to interact with the OTM components representing the user's organizational unit. Communication between user and the OTM components consists of three steps:

Each DHTML web page uses Javascript to communicate with a Java Servlet using XML-RPC.

Each Java Servlet processes the user actions calling an OTM component using the component's webservice interface defined by its communication subsystem.

The OTM GUI has been designed as a client lightweight interface with no important information processing being carried out at the client side. Most of the event processing is handled at the server side. Therefore only the server side (the OTM Application components) had to be adapted to become provenance-aware.

2.2 The Interface to the Provenance System through the Client Side Library

In order to make the OTM application provenance-aware, the OTM components record several p-assertions for any key event in the system. Figure 4 shows a diagram displaying a subset of a typical recording schema for a traditional OTM case workflow:

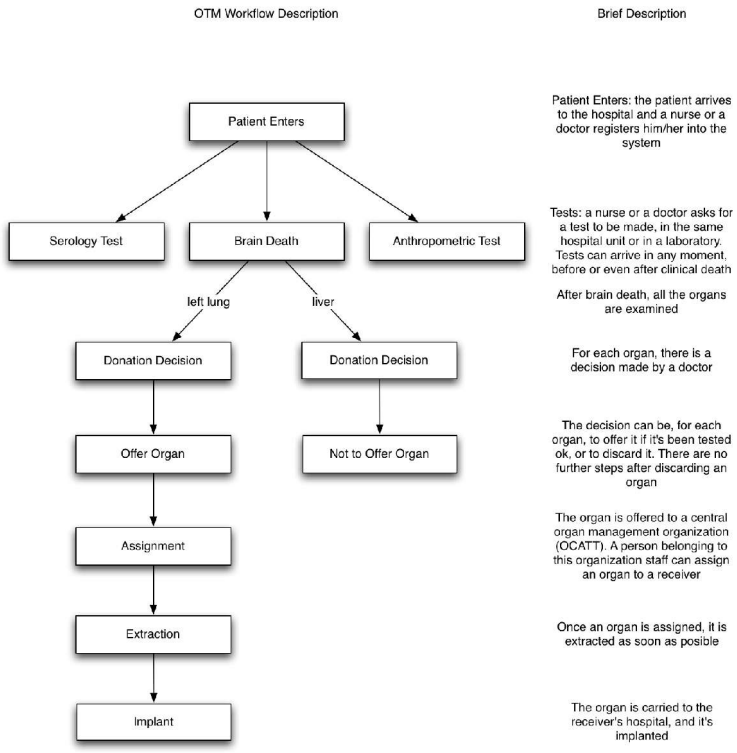


Figure 4 Example of OTM workflow (a subset)

Once one of these key events is triggered, the OTM application uses the wrapper to record a p-assertion in the Provenance Store. The information stored in a p-assertion typically includes the event type, the session of the user that triggered the action, a time stamp, the parameters describing the event, and the relationship of that p-assertion with others related (in the form of relationship p-assertions).

The Client Side Library [D9.3.3, D9.3.3a] is a Java API that provides a complete set of classes and methods to manage, record and retrieve data to and from a Provenance Store. The CSL makes recording in the Provenance Store an easy task, but as the OTM application records many p-assertions (approximately 200 during a typical patient case) with a very similar structure and common parameters, the OTM Application uses a wrapper implementing some higher-level routines for recording p-assertions. Most of the wrapper recording routines consist of the following steps:

1. Create a view record
2. Add one interaction p-assertion and zero or more relationship p-assertion(s) to the view record
3. Record the view record into the Provenance Store

Using the wrapper routines simplifies the provenance recording in the OTM application as it provides higher-level routines adapted to the application domain. It also reduced the dependencies between the OTM code and the different versions of the CSL interface during its development: each time a significant change was made in the CSL, it only impacted the wrapper, not the OTM application code.

An extra role of the wrapper is to ease recording of the communication events between the OTM application and the EHCRS application and itself. The EHCRS needs special Provenance information in order to make bindings between the OTM events and EHCRS ones. The wrapper includes several methods to handle this, using the p-header component of the Provenance structure. This p-header

contains information generated by the OTM Application and is used by the EHCRS application (see section 2.4.).

2.3 The EHCR Subsystem

The EHCR Store serves as the data management part of OTM. It follows ENV13606 European pre-standard communicating with external medical institutes. This subsystem is also provenance aware and using the Client Side Library to assert p-assertions.

There are two main interfaces of EHCR Store:

- EHCRS interface is for local OTM (and possibly other local medical applications).
- ENV13606 interface is for remote EHCR Stores (and possibly other remote medical applications, prepared for ENV13606)

2.4 The Interface to the Provenance System through the Client Side Library

Although the Client Side Library (CSL) provides a good set of functions to record and query p-assertions to and from a provenance store, the EHCRS application uses a wrapper similar to the OTM one to simplify recording of the p-assertions. Doing this reduces the steps needed for lines of code needed: four to create an assertion and one for each extra p-assertions in the same view record.

In case of EHCRS receiving a message, the recording steps are as follows:

1. extract the p-header from the message received
2. create a new view record, based on the p-header
3. add one interaction p-assertion and zero or more actor state or relationship p-assertion(s) to the view record
4. record the view record into the provenance store

In case of EHCRS sending a message, the recording steps are the following:

1. create a view record
2. create a p-header, based on the view record
3. add one interaction p-assertion and zero or more actor state or relationship p-assertion(s) to the view record
4. record the view record into the provenance store

When an interaction p-assertion is added to a view record, the wrapper also adds an actor state p-assertion containing a timestamp to the same view record.

2.5 The Interface between the OTM and the EHCR Subsystem

Communication between OTM and EHCRS is deployed over SOAP/XML and HTTP as normal, however a limited generic interface corresponding roughly to the Agent Communication Language FIPA-ACL [FIPA02b, FIPA02c] is used for all messages between OTM and EHCRS services. Using

this device all messages are generated using general classes of message such as *Inform, Request, Agree* etc. which characterize the type of message being sent (following Speech Act theory).

More concretely, both OTM and EHCRS provide the following list of web service operations:

- `AGREE`: an acknowledgment over a (previous) given message.
- `REFUSE`: a refusal over a (previous) given message.
- `INFORM DONE`: response to a (previous) message, informing that the requested action has been done
- `FAILURE` : response to a (previous) message, informing that there was a failure while generating a result.

Apart of the above operations, the EHCRS webservice interface provides extra operations:

- `LOGIN_REQUEST` and `LOGOUT_REQUEST`: logging in and out used for authenticating the doctor and the OTM,
- `STARTCASE_REQUEST`: starting a case of patient, to open a subsession for the patient avoid using the public patient ID in messages where sensitive medical data can travel,
- `GETEHCR_REQUEST` and `SETEHCR_REQUEST`: requesting and updating electronic health care record (EHCR) of the patient.

2.6 The Interface between the OTM Application and the Tool

The Provenance Tool, as a web application, is linked using hypertext in both the OTM GUI and the EHCRS GUI. Once inside the Tool, the user can follow the specific instructions for Provenance Tool to navigate and configure the portal.

Apart of the usage of the Provenance Tool as an independent application, there are some key user actions of the OTM GUI which give direct access to a specific portlet of the Tool, parametrized with some contextual queries related to a given user action. For instance, when the user is checking a specific donation case record, a *"Check donation case history"* button is shown as one of the contextual action buttons in the OTM GUI. This button redirects to the Tool webpage, with the donation case Provenance query already executed. The user can then navigate through the Tool portlets, so he can check the result of the query in text or graphical visualizations.

3 Deployment of the OTM Application

Source of OTM and EHCRS is stored in the CVS of EUProveance. After a checkout ANT can be used to compile, build and deploy the components (web applications, web services and databases). OTM and EHCRS can be handled separately.

3.1 Deploying OTM

The OTM application has two main components that can be deployed separately: the core system (OTM) and the graphical interface (OTM GUI). Code is divided in two ANT projects: the `OTMApplication` project and the `OTMGui` project. Each instance of OTM represents an specific organisational unit (such as hospitals, laboratories or transplant management units), and therefore has to be installed in each one of the entities. OTM GUI provides a web interface to interact with the distributed OTM system, and one deployed OTM GUI can be shared by more than one OTM instace.

The `OTMApplication` and `OTMGui` projects can be compiled using ANT. Both projects use a `build.properties` file which define some important parameters that should be changed for each specific instance, such as the hospital name,the list of its units for OTM and the URL of the EHCR Store to be used as data source.

The result of the ANT build for `OTMApplication` is a WAR file that has to be deployed in a Tomcat server, copying it into the webapps directory. After a few seconds, it will be automatically deployed. The ANT `build.xml` file of the project also has some tasks for automatic deployment and context refresh in the case the Tomcat server is running locally.

The `OTMGui` project ANT file generates a www directory and an `otmgui.jar` file. This file should be copied into the WEB-INF/lib directory of the OTM context in the Tomcat server. The www directory is a web page that can be deployed in any web server. The ANT `build.xml` file for this project includes tasks that do this process automatically, deploying everything into the OTMA context of a local Tomcat server.

3.2 Deploying EHCRS

In case of EHCRS, an ANT property file is provided to choose which tasks (e.g. compile sources, create database, install web application, ...) should be performed during the three basic ANT processes, namely *install*, *uninstall* and *reinstall*.

All instances of EHCRS have its own property file for deploy process. In this file the access points of the other components (such as Provenance Service) should be set.

During the deployment, an EHCRS database can be created and initialized along with the user of this database. The EHCRS core and the 'Test OTM' applications can be built into a web archive (WAR) file, which can be then installed into a Tomcat web server.

After this process both the EHCRS and 'Test OTM' applications should be fully functional. This can be checked with an specific web page to check the installed web applications (see figure 5).

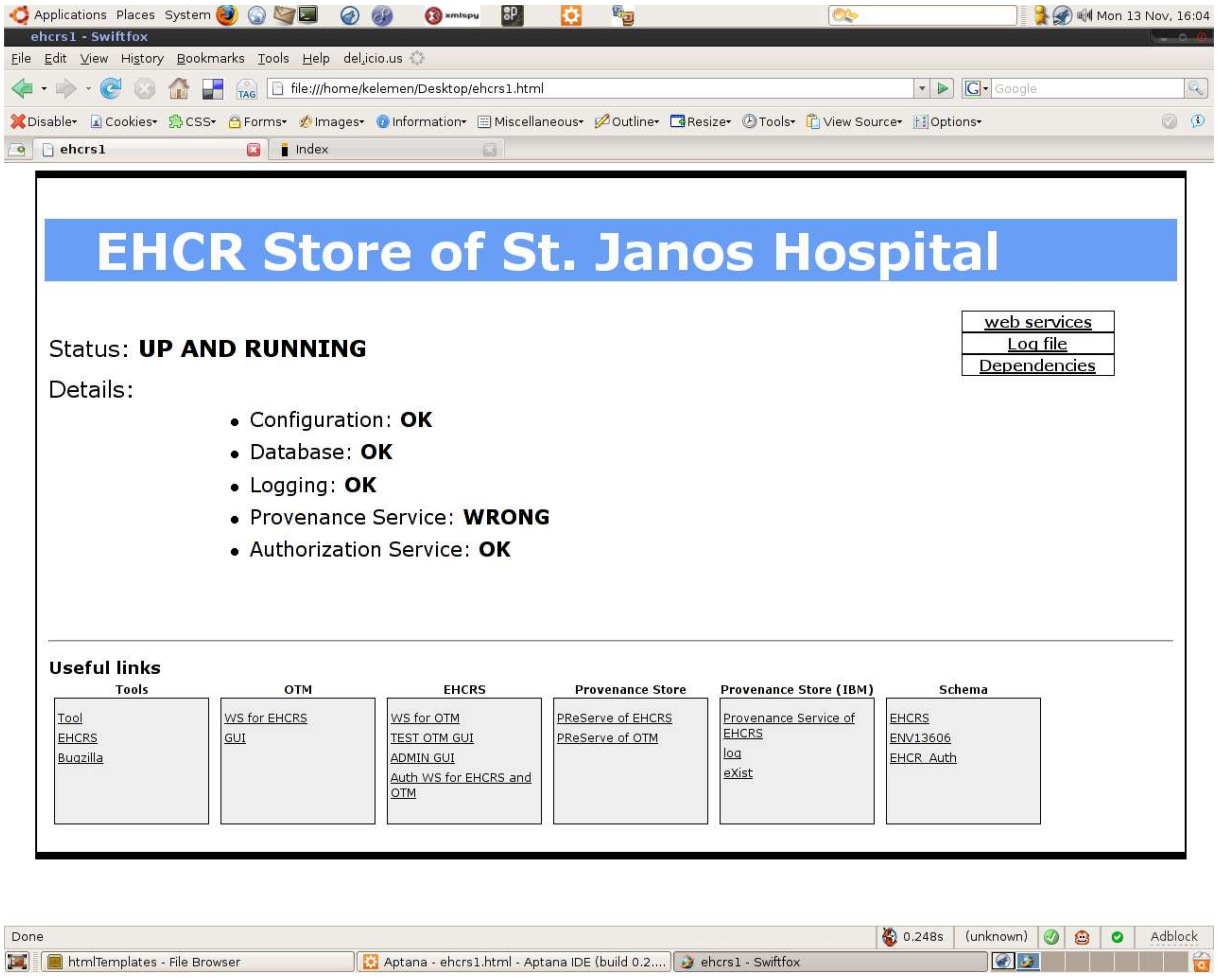


Figure 5 Interface to check the running services.

For more details on installation, please see the `install.txt` file in the root directory of the OTM-EHCR distribution

3.3 Deploying EHCR Auth

In the case of EHCR Auth the deployment process involves both the installation of the EHCR Auth web (service) application and the installation of the application's eXist database system.

The deployment of the EHCR Auth application can be done with the help of the Ant build tool, resulting in a web archive (war) file, which can be directly installed as a Tomcat server application.

The deployment of the eXist database system should be carried out separately, as it is described in the eXist manuals.

A very important step of the deployment is the editing of the appropriate properties by which the various environmental settings (like the address of the Provenance Store to be used) can be given.

For details of installation, see the `readme.txt` and `install.txt` files in the root directory of the code distribution.

4 Deployed OTM Demonstrator

OTM runs in a SOA environment and uses several services developed by different partners of the EUProvenance project. These services can run on different computers. As mentioned in previous chapters, these services represent organizations from real life such as transplant authority or hospitals.

4.1 Humans

The human actors in this prototype application are three doctors and four patients. The tables below:

- describe real name, user name and password of the doctors
- describe real name, type of the medical ID and the medical ID itself of the patients

The doctors already introduced in the OTM demonstrator are the following:

Name	***Username***	***Password***
Marta Sánchez Sánchez	msanchez	msanchez
Javier Vázquez Salceda	jvazquez	jvazquez
Sergio Alvarez Napagao	salvarez	salvarez

The patients already introduced in the OTM demonstrator are the following:

Name	***Medical ID type***	***Medical ID***
Ramón Pérez Pérez	Catalonian Insurance Number	36347834E
Laura Gómez Ruiz	Catalonian Insurance Number	47347342F
Carlos García Quiñones	Catalonian Insurance Number	34378433B
Mr. Anderson	EU Insurance Number	12345678A

4.2 Organizations

In the demonstrator four organizations are already modeled: three hospitals (Hospital de Sant Pau, Hospital de la Vall d'Hebron and Hospital Clinic) and the Organización Catalana de Trasplantes (OCATT), which is the transplant authority in Catalonia.

The services provided by OCATT in the system include:

- authenticating doctors
- authenticating applications

- registering patients
- identifying patients
- organ transplant management mediation between the hospitals.

Services in the hospitals include

- querying and updating the EHCR of a patient
- organ transplant management services.

4.3 Applications

The demonstrator consists of the following applications:

- **OTM**: The Organ Transplant Management application

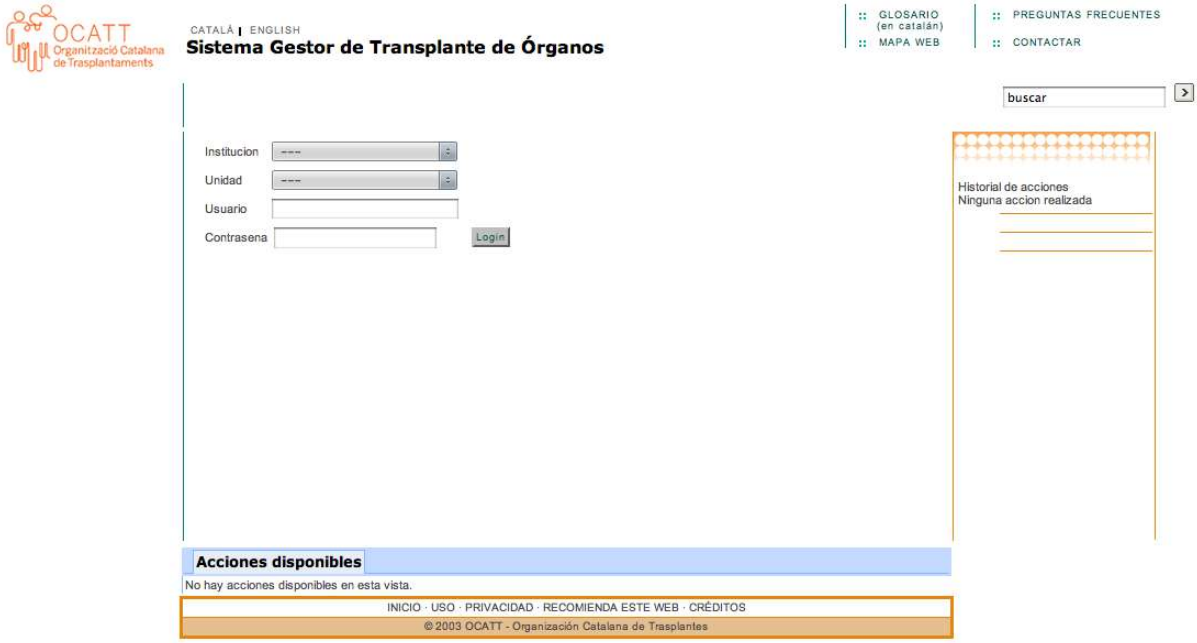


Figure 6 The OTM application GUI.

- **EHCRS**: the EHCR store. It provides services for querying and updating the EHCR of a patient. It has no GUI.
- **EHCR_Auth**: the application providing the authentication services to the other applications

Figure 7 The EHCR_Auth application GUI.

- **TEST_MA**: an application that provides a GUI for testing EHCRS without the use of OTM.

Figure 8 The TEST_MA application GUI.

EHCR_Auth, EHCRS and OTM are provenance aware applications, while TEST_MA is not. All the installed provenance aware applications use a provenance store to document the processes they are doing.

Apart of these applications, the **Provenance Tools** (developed in Workpackage 6) are used to helps humans inspect and query the content of the provenance stores.

Most of the application components are installed on three computers: ibm04.ilab.sztaki.hu, copan.lsi.upc.edu and tulum.lsi.upc.edu. For each computer there are four server applications which host all the demonstrator components. An additional MySQL database is installed in tulum.lsi.upc.edu and acts as a cache database for OTM in case the EHCR Store is not accessible at a certain moment.

Figure 9 shows a high-level view of the server applications installed in ibm04.ilab.sztaki.hu.

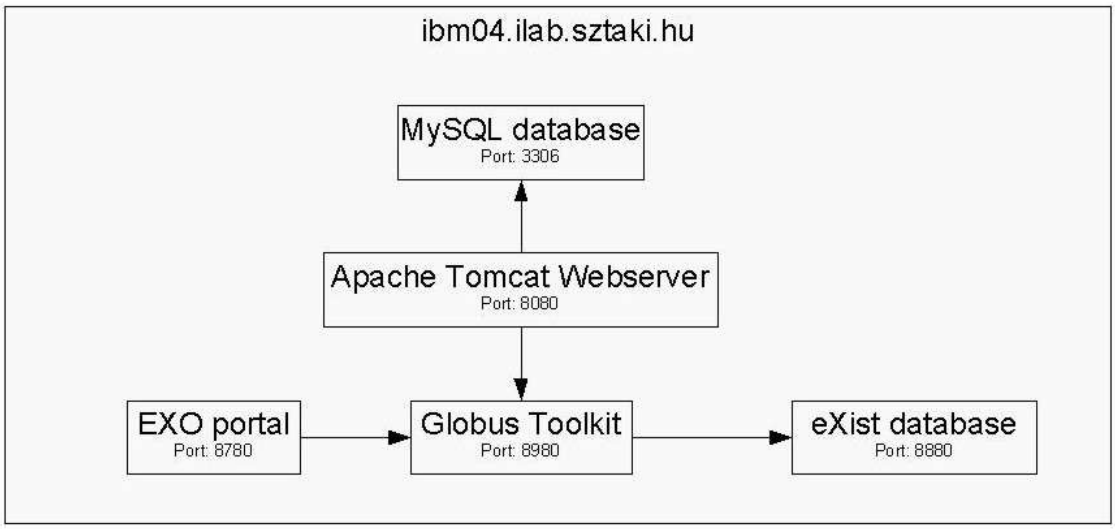


Figure 9 Server applications and communications between each other in ibm04.ilab.sztaki.hu

Figure 10 shows the connections between the components of each OTM instance. To simplify the diagram, in figure 10 all instances are deployed in the same computer, but it can also be deployed on several computers.

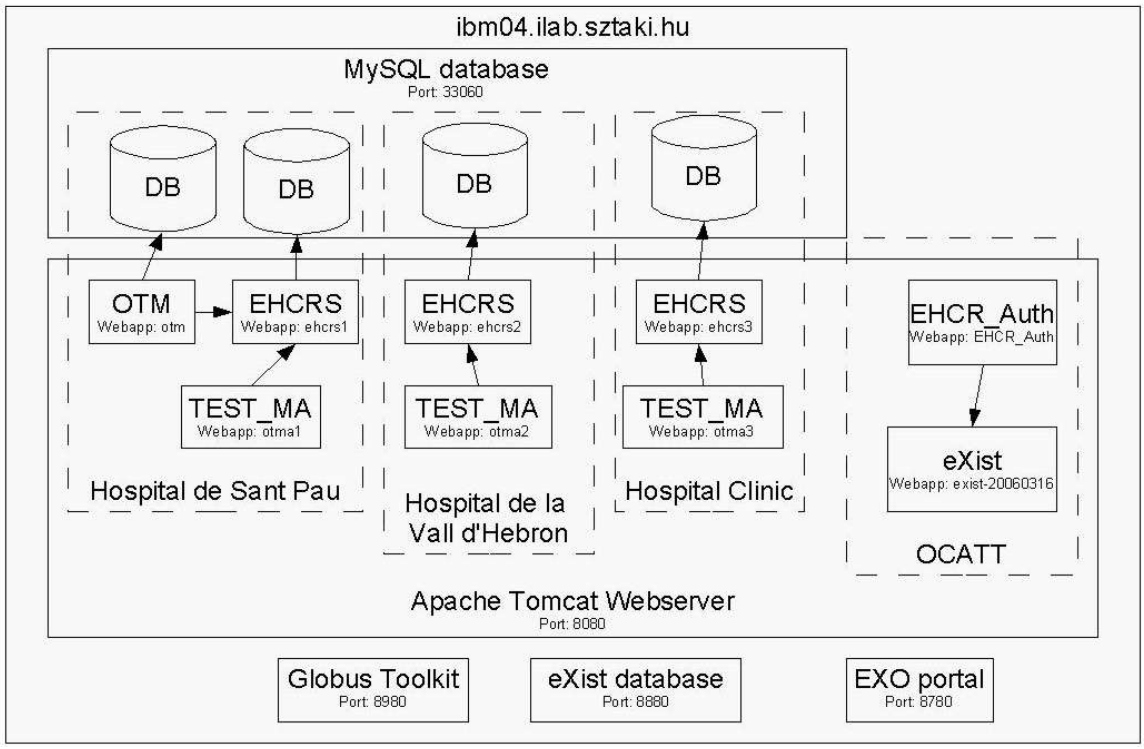


Figure 10 Organizations and communications inside them.

Finally, figure 11 shows an example of the connectivity between OTM instances when deployed in the same computer.

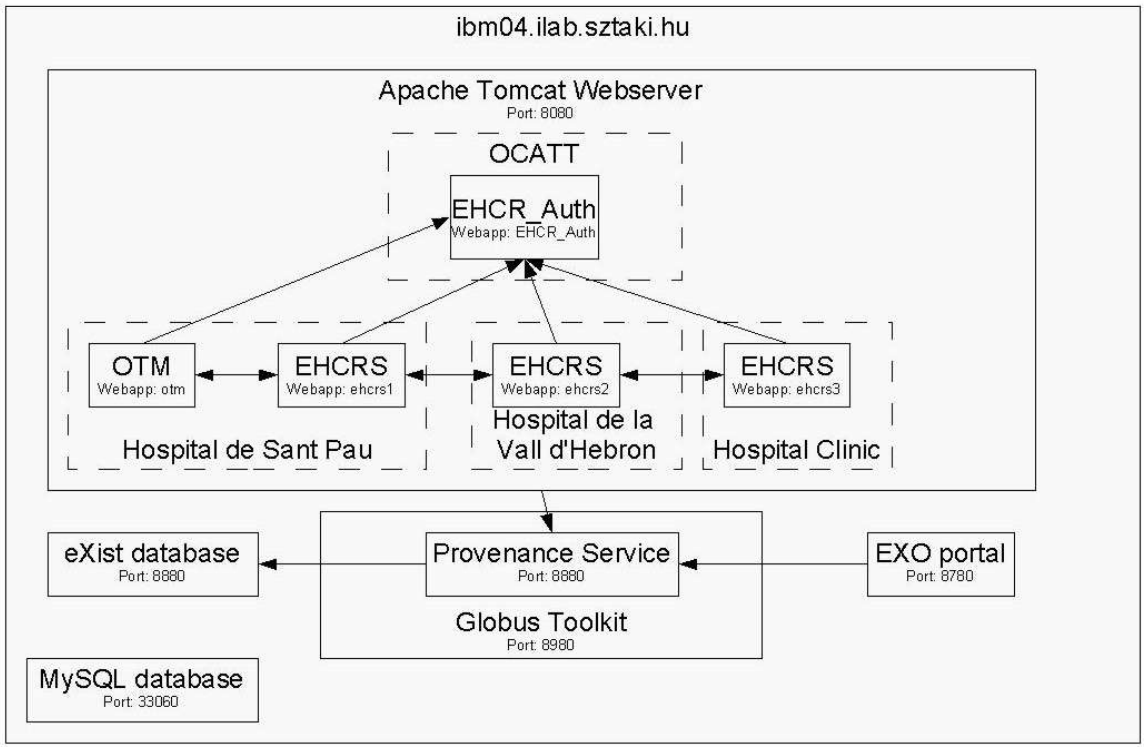


Figure 11 Organizations and communications between each other.

4.4 Access points

The following table lists all the access points to the installed components in the OTM demonstrator.

Owner	*Name*	*URL*
OCATT	EHCR_Auth GUI	http://ibm04.ilab.sztaki.hu:8080/EHCR_Auth/userLogin.jsp
OCATT	eXist database	http://ibm04.ilab.sztaki.hu:8080/exist-20060316 Collection: ehcrauth
Hospital de Sant Pau	EHCR Store	http://ibm04.ilab.sztaki.hu:8080/ehcrs1
Hospital de Sant Pau	TEST_MA	http://ibm04.ilab.sztaki.hu:8080/otma1
Hospital de Sant Pau	OTMA	http://tulum.lsi.upc.es/otm/services/
Hospital de Sant Pau	OTM GUI	http://tulum.lsi.upc.es/otm/org.gridprovenance.otm.www.OTMGui/OTMGui.html

Owner	*Name*	*URL*
Hospital de la Vall d'Hebron	EHCR Store	http://ibm04.ilab.sztaki.hu:8080/ehcrs2
Hospital de la Vall d'Hebron	TEST_MA	http://ibm04.ilab.sztaki.hu:8080/otma2
Hospital de la Vall d'Hebron	OTMA	http://copan.lsi.upc.es/otm/services/
Hospital de la Vall d'Hebron	OTM GUI	http://copan.lsi.upc.es/otm/org.gridprovenance.otm.www.OTMGui/OTMGui.html
Hospital Clinic	EHCR Store	http://ibm04.ilab.sztaki.hu:8080/ehcrs3
Hospital Clinic	TEST_MA	http://ibm04.ilab.sztaki.hu:8080/otma3
Hospital Clinic	OTMA	http://ibm04.ilab.sztaki.hu:8080/otm/org.gridprovenance.otm.www.OTMGui/OTMGui.html
Provenance Store	Provenance Service	http://ibm04.ilab.sztaki.hu:8980
Provenance Store	eXist database	http://ibm04.ilab.sztaki.hu:8880 Collection: ProvenanceStore
Tool	Tool	http://ibm04.ilab.sztaki.hu:8780/portal

Appendix A User Manual

A.1 Installing applications

For details please see `install.txt` in root directory of the OTM distribution.

A.2 Initial status of the actors

After the application are installed the doctors are registered in OCATT and in every EHCR Stores. Patients are not registered in OCATT. EHCR Stores, TEST_MAs and OTMs are registered in OCATT. Provenance Store is empty.

A.3 Start the medical history of Mr. Anderson

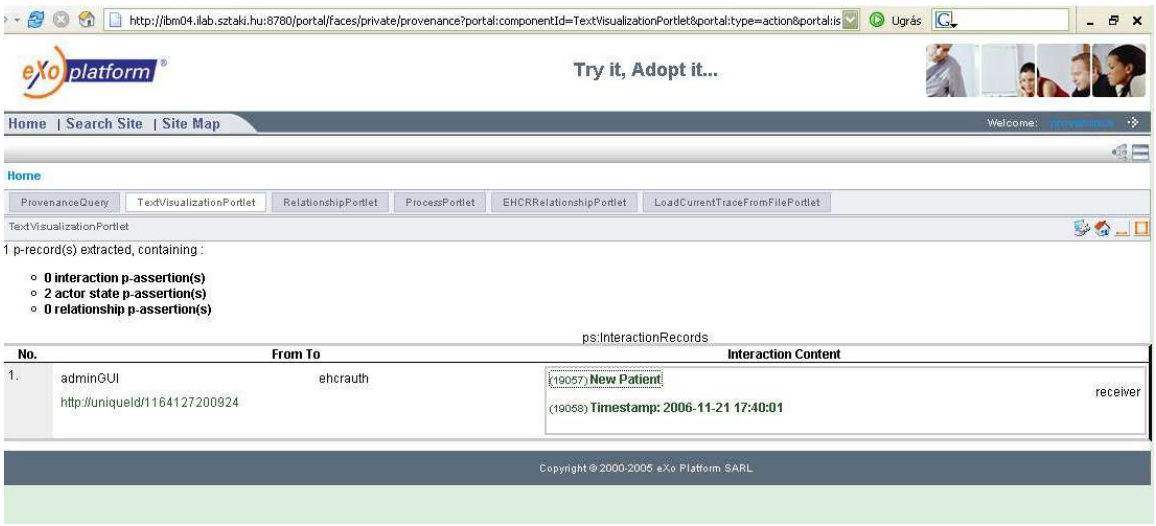
1. Open OCATT in a browser.
2. Login with “ehcrauth” as username and password to “patient administration”

3. Fill the form on page “Patient Administration”
   1. Patient name: “Mr. Anderson”
   2. Insurance type: “Catalonian Insurance Number”
   3. Insurance number: 363478234E

4. Click on ”Insert” button.

Mr. Anderson is now registered in OCATT. You can check this event in the Provenance Store with the Tool.

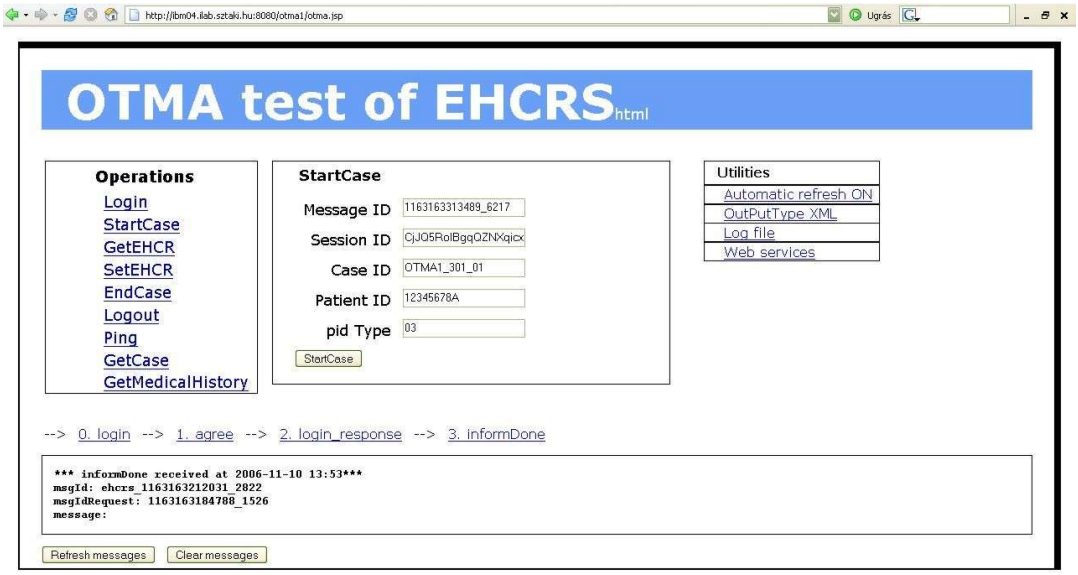
1. Open the Tool in a browser.
2. Use the Tool according to the documentation of the Tool.
3. You can see that *“New Patient is created”* in the Text Visualization Portlet



A.4 Treat Mr. Anderson

1. Open TEST_MA of Hospital de la Vall d'Hebron in a browser.
   1. Choose “Automatic refresh ON” if your browser supports AJAX.
2. Login with “msanchez” as username and password.

3. Keep pressing “Refresh Messages” button (or wait if “Automatic refresh” is ON) till you see the “informDone” answer.
4. Choose “StartCase” menu item in the “Operations” menu (on the left side).
5. Choose “03_301_01” as case ID and click on “StartCase” button.



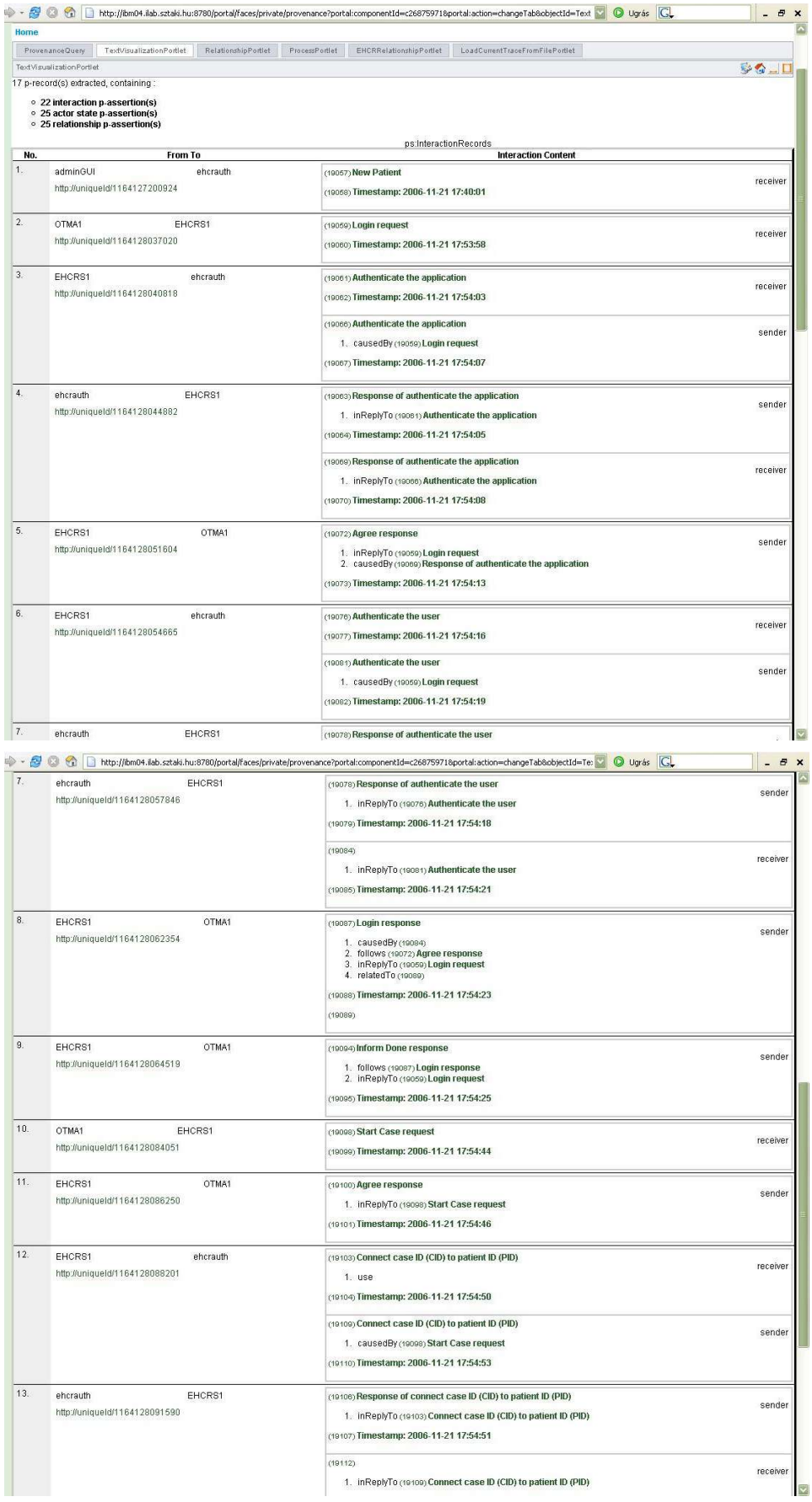
6. Keep pressing “Refresh Messages” button (or wait if “Automatic refresh” is ON) untill you see the “informDone” answer.
7. Choose “SetEHCR” menu item in the “Operations” menu.
8. Edit the XML representing the EHCR as you wish and click on “SetEHCR” button.
9. Keep pressing “Refresh Messages” button (or wait if “Automatic refresh” is ON) till you see the “informDone” answer.
10. Choose “Logout” menu item in the “Operations” menu.
11. Click on “Logout” button.

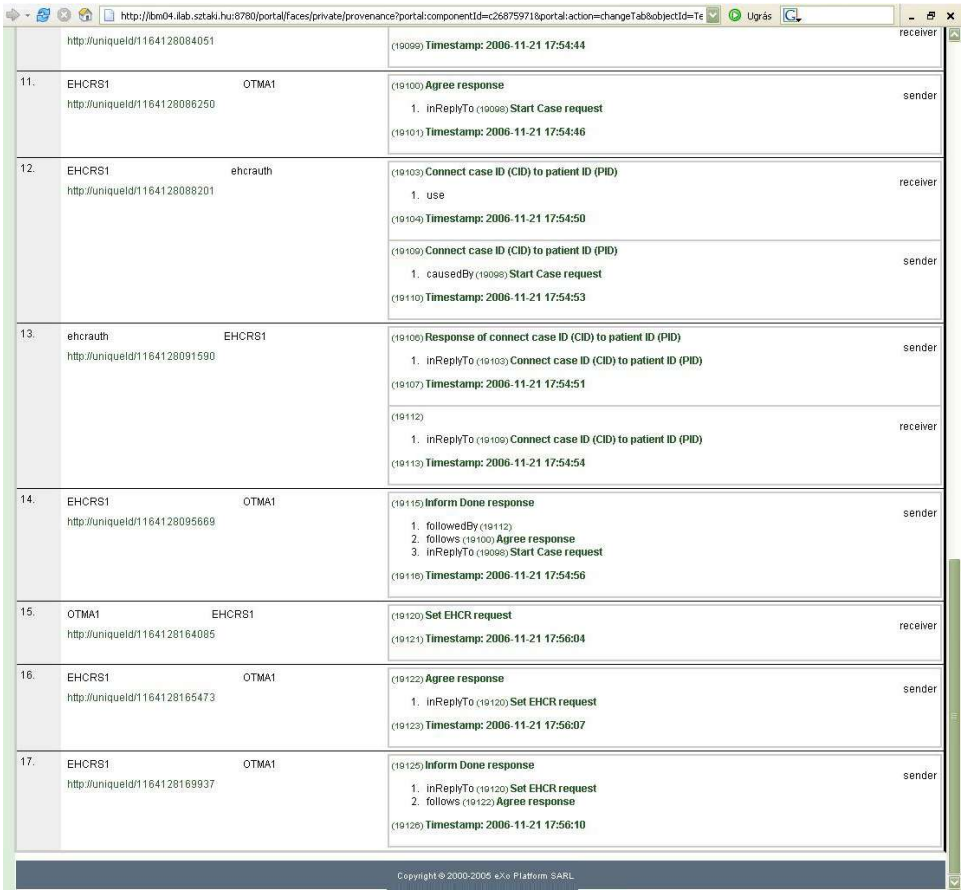
You can check the following sub-processes in Provenance Store with the Tool:

a) After the 3nd step: the login process in Relationship Portlet

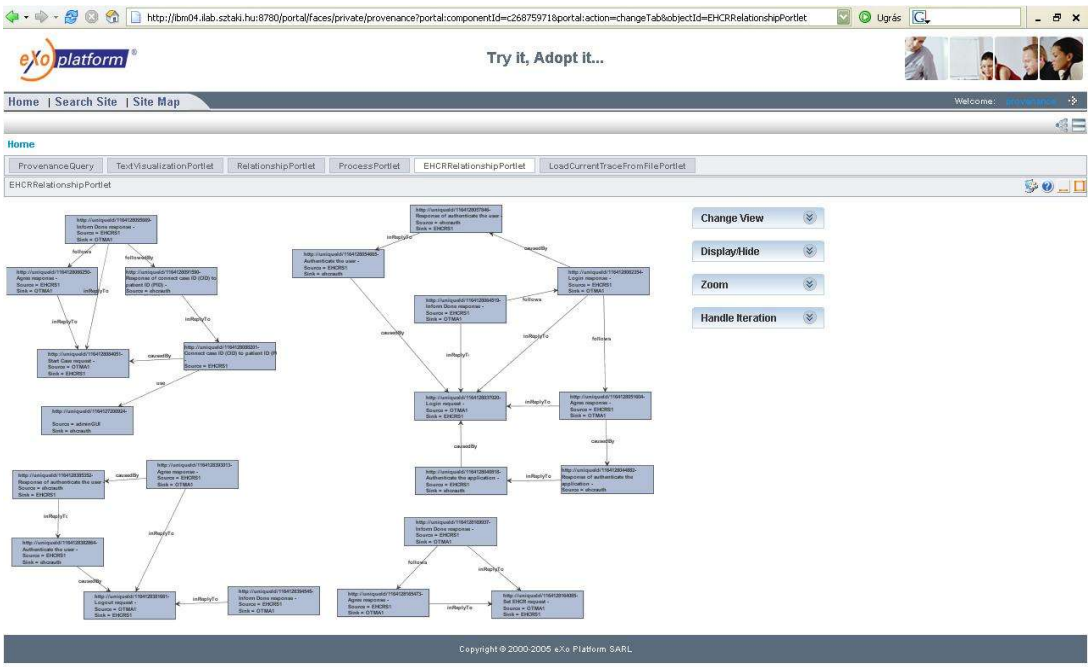
b) After the 6th step: the new case process in Relationship Portlet. You can see that the new case is connected to the new patient event.

c) After the 9th step: the setEHCR process in the Text Visualization Portlet. You can see that the setEHCR process is connected to the new case process.


http://ibm04.ilab.sztaki.hu:8780/portal/faces/private/provenance?portal:componentId=c26875971&portal:action=changeTab&objectId=Text
Ugrás
Home
ProvenanceQuery
TextVisualizationPortlet
RelationshipPortlet
ProcessPortlet
EHCRRelationshipPortlet
LoadCurrentTraceFromFilePortlet
TextVisualizationPortlet
17 p-record(s) extracted, containing :
22 interaction p-assertion(s)
25 actor state p-assertion(s)
25 relationship p-assertion(s)
ps:InteractionRecords
No.
From To
Interaction Content
1. adminGUI ehcrauth
http://uniqueId/1164127200924
(19057) New Patient
receiver
(19058) Timestamp: 2006-11-21 17:40:01
2. OTMA1 EHCRS1
http://uniqueId/1164128037020
(19059) Login request
receiver
(19060) Timestamp: 2006-11-21 17:53:58
3. EHCRS1 ehcrauth
http://uniqueId/1164128040818
(19061) Authenticate the application
receiver
(19062) Timestamp: 2006-11-21 17:54:03
(19066) Authenticate the application
sender
1. causedBy (19059) Login request
(19067) Timestamp: 2006-11-21 17:54:07
4. ehcrauth EHCRS1
http://uniqueId/1164128044882
(19063) Response of authenticate the application
sender
1. inReplyTo (19061) Authenticate the application
(19064) Timestamp: 2006-11-21 17:54:05
(19069) Response of authenticate the application
receiver
1. inReplyTo (19066) Authenticate the application
(19070) Timestamp: 2006-11-21 17:54:08
5. EHCRS1 OTMA1
http://uniqueId/1164128051604
(19072) Agree response
sender
1. inReplyTo (19059) Login request
2. causedBy (19069) Response of authenticate the application
(19073) Timestamp: 2006-11-21 17:54:13
6. EHCRS1 ehcrauth
http://uniqueId/1164128054665
(19076) Authenticate the user
receiver
(19077) Timestamp: 2006-11-21 17:54:16
(19081) Authenticate the user
sender
1. causedBy (19059) Login request
(19082) Timestamp: 2006-11-21 17:54:19
7. ehcrauth EHCRS1
(19078) Response of authenticate the user
http://ibm04.ilab.sztaki.hu:8780/portal/faces/private/provenance?portal:componentId=c26875971&portal:action=changeTab&objectId=Te
Ugrás
7. ehcrauth EHCRS1
http://uniqueId/1164128057846
(19078) Response of authenticate the user
sender
1. inReplyTo (19076) Authenticate the user
(19079) Timestamp: 2006-11-21 17:54:18
(19084)
receiver
1. inReplyTo (19081) Authenticate the user
(19085) Timestamp: 2006-11-21 17:54:21
8. EHCRS1 OTMA1
http://uniqueId/1164128062354
(19087) Login response
sender
1. causedBy (19084)
2. follows (19072) Agree response
3. inReplyTo (19059) Login request
4. relatedTo (19088)
(19088) Timestamp: 2006-11-21 17:54:23
(19089)
9. EHCRS1 OTMA1
http://uniqueId/1164128064519
(19094) Inform Done response
sender
1. follows (19087) Login response
2. inReplyTo (19059) Login request
(19095) Timestamp: 2006-11-21 17:54:25
10. OTMA1 EHCRS1
http://uniqueId/1164128084051
(19098) Start Case request
receiver
(19099) Timestamp: 2006-11-21 17:54:44
11. EHCRS1 OTMA1
http://uniqueId/1164128086250
(19100) Agree response
sender
1. inReplyTo (19098) Start Case request
(19101) Timestamp: 2006-11-21 17:54:46
12. EHCRS1 ehcrauth
http://uniqueId/1164128088201
(19103) Connect case ID (CID) to patient ID (PID)
receiver
1. use
(19104) Timestamp: 2006-11-21 17:54:50
(19109) Connect case ID (CID) to patient ID (PID)
sender
1. causedBy (19098) Start Case request
(19110) Timestamp: 2006-11-21 17:54:53
13. ehcrauth EHCRS1
http://uniqueId/1164128091590
(19106) Response of connect case ID (CID) to patient ID (PID)
sender
1. inReplyTo (19103) Connect case ID (CID) to patient ID (PID)
(19107) Timestamp: 2006-11-21 17:54:51
(19112)
receiver
1. inReplyTo (19109) Connect case ID (CID) to patient ID (PID)



d) After the 9th step: the setEHCR process in Relationship Portlet. You can see that the setEHCR process is connected to the new case process.



You can repeat the process with (TEST) OTM of Hospital Clinic with jvazquez, with case ID "03_301_02" and with a different content of EHCR XML.

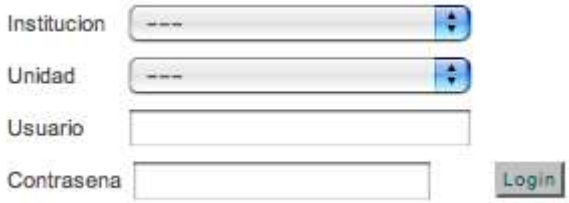
A.5 Query Mr. Anderson's whole EHCR from EHCRS3

1. Open TEST_MA of Hospital de Sant Pau in a browser.
    1. Choose “Automatic refresh ON” if your browser support AJAX.
2. Login with ”salvarez” as username and password.
3. Keep pressing “Refresh Messages” button (or wait if “Automatic refresh” is ON) till you see the “informDone” answer.
4. Choose ”StartCase” menu item in the “Operations” menu (on the left side).
5. Choose “03_301_03” as case ID and click on “StartCase” button.
6. Keep pressing “Refresh Messages” button (or wait if “Automatic refresh” is ON) till you see the “informDone” answer.
7. Click on the ”GetEHCR” menu item in the “Operations” menu.
8. Click on “GetEHCR” button.
9. Keep pressing “Refresh Messages” button (or wait if “Automatic refresh” is ON) till you see the “informDone” or “Failure” answer
10. Check the answers from EHCRS3:
    1. You can see that there are no data stored locally for Mr. Anderson.
    2. You can see EHCRS1 and EHCRS2 will be queried for more data.
    3. You can see the medical data from EHCRS1.
    4. You can see the medical data from EHCRS2.
11. Click on the „Logout” menu item in the “Operations” menu.
12. Click on „Logout” button.

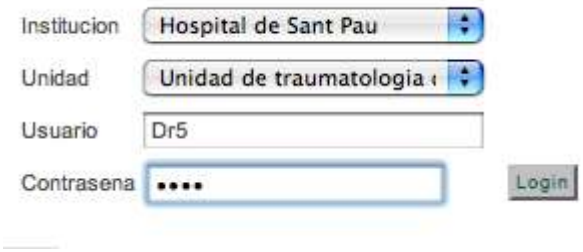
Check the medical history of Mr. Anderson in Provenance Store with the Tool (Relationship Portlet).

A.6 Logging into the OTM GUI

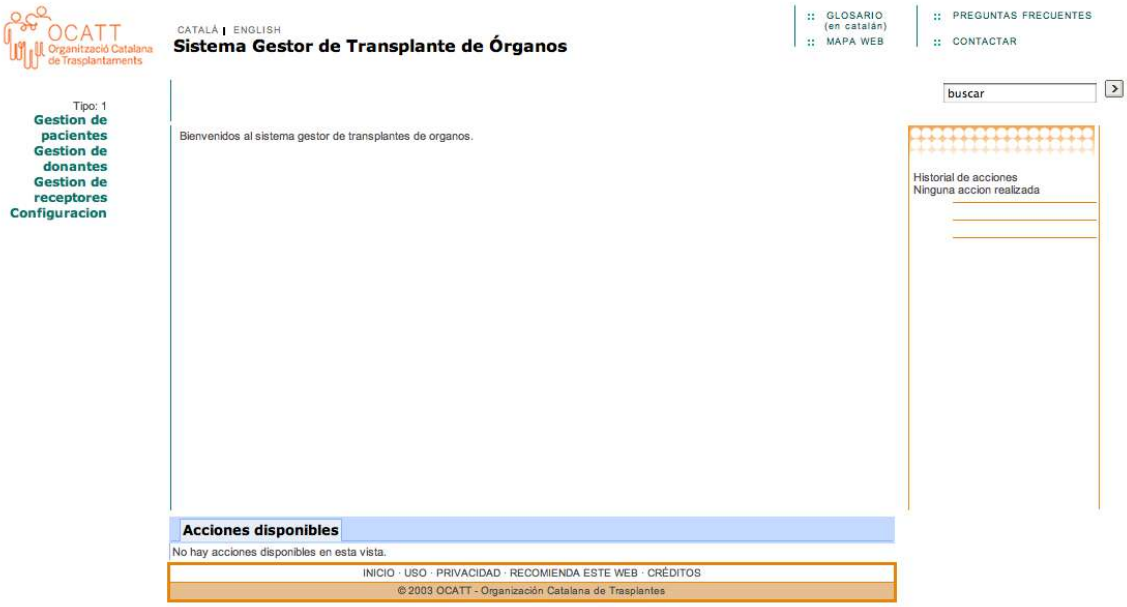
1. Open the OTM GUI in a browser.



2. Select the institution *('Institución')* and the unit (*'Unidad'*) to connect to a specific OTM Application.



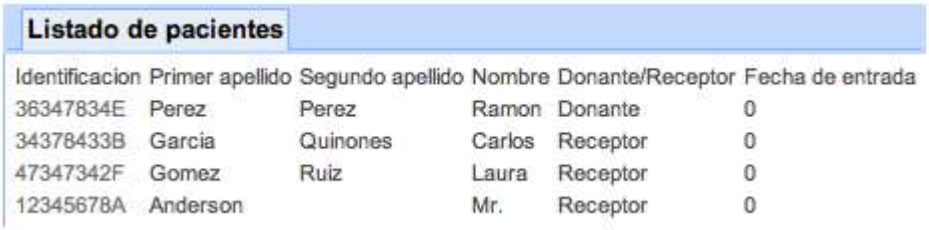
3. If the username and the password are correct and there are permissions to access the institution, the main menu page is shown.



A.7 Listing the patients

1. Select the Patient Management (*'Gestión de pacientes'*) submenu.

2. Click on "List patients" (*'Listar pacientes'*)



Listado de pacientes

Identificacion	Primer apellido	Segundo apellido	Nombre	Donante/Receptor	Fecha de entrada
36347834E	Perez	Perez	Ramon	Donante	0
34378433B	Garcia	Quinones	Carlos	Receptor	0
47347342F	Gomez	Ruiz	Laura	Receptor	0
12345678A	Anderson		Mr.	Receptor	0

A.8 Adding a new patient

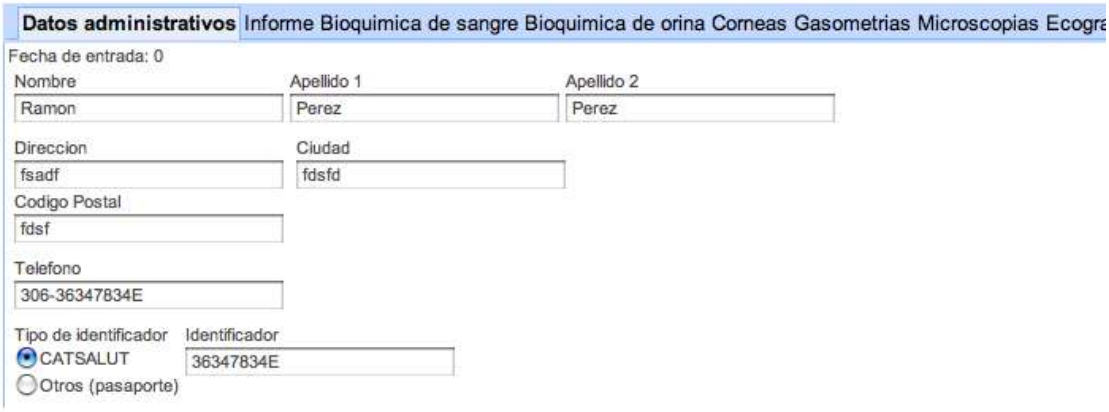
1. To add a new patient, select the Patient Management (*'Gestión de pacientes'*) submenu.

2. Select the "Add a new patient" (*'Dar de alta un paciente'*) option. An empty form will appear.

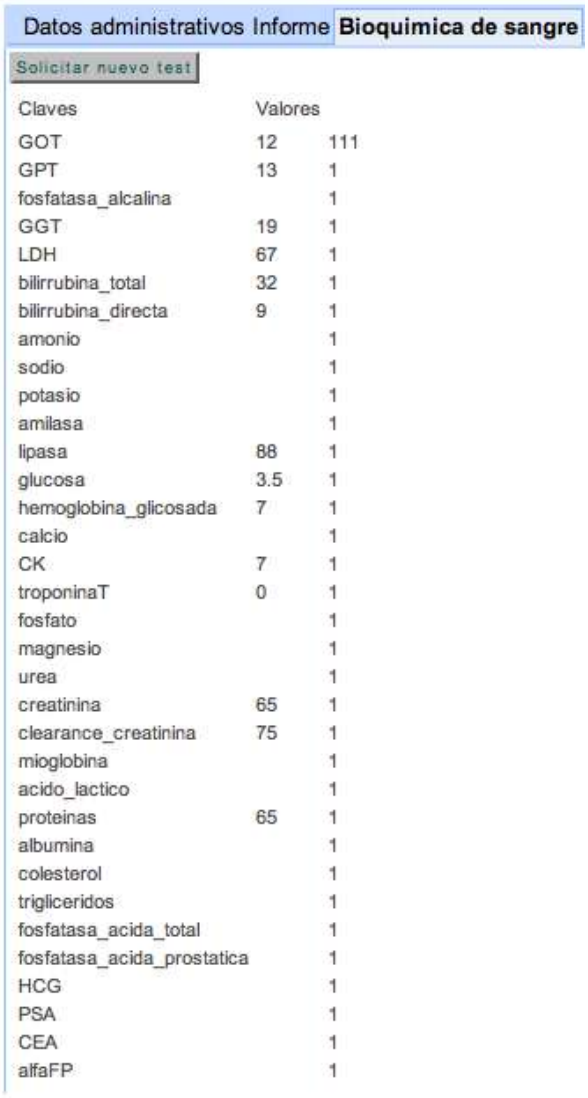
3. Fill in the form and click on the "Add patient" (*'Añadir paciente'*) button.

A.9 Managing a patient record

1. Click on the identification number of the desired patient in the patient list. The patient record appears on screen.



2. You can navigate through the records tab on top to see specific test results and other information.



Claves	Valores	
GOT	12	111
GPT	13	1
fosfatasa_alcalina		1
GGT	19	1
LDH	67	1
bilirrubina_total	32	1
bilirrubina_directa	9	1
amonio		1
sodio		1
potasio		1
amilasa		1
lipasa	88	1
glucosa	3.5	1
hemoglobina_glicosada	7	1
calcio		1
CK	7	1
troponinaT	0	1
fosfato		1
magnesio		1
urea		1
creatinina	65	1
clearance_creatinina	75	1
mioglobina		1
acido_lactico		1
proteinas	65	1
albumina		1
colesterol		1
trigliceridos		1
fosfatasa_acida_total		1
fosfatasa_acida_prostatica		1
HCG		1
PSA		1
CEA		1
alfaFP		1

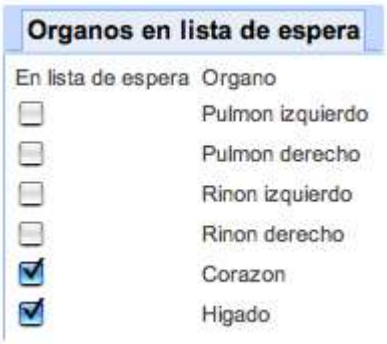
3. While inspecting a test result, you can order a new test of the same type by clicking on the "Request a new test" (*'Solicitar nuevo test'*) button.

A.10 Managing a patient's donation status

1. In a patient record, you can change the donation status. To do so click on the "Potential Donor" button (*'Donante potencial'*) or the "Potential Recipient" one (*'Receptor potencial'*)) on the action button list.

2. If the patient is already a potential recipient, you can manage the list of organs to be requested by clicking on the "Manage Waiting List" (*'Gestionar lista de espera'*) button.

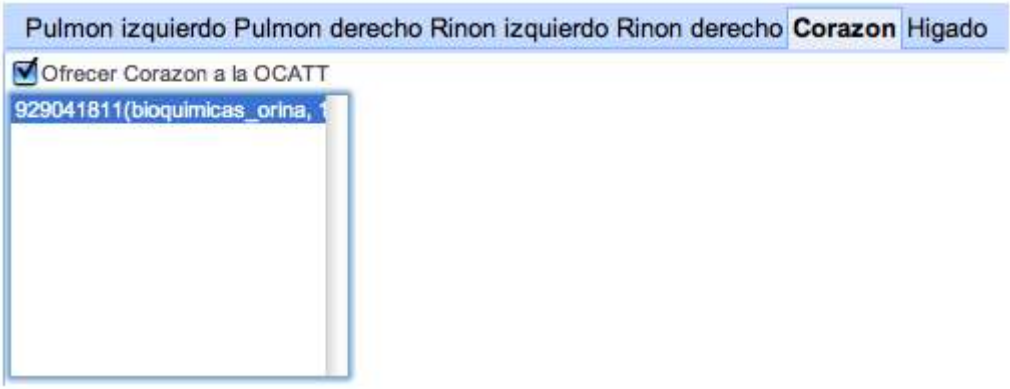
1. You will be asked for the list of organs. Just check or uncheck the organs on the list.



3. If the patient is already a potential donor, you can manage the donation process of the patient by clicking on the "Offer to OCATT" *('Ofrecer a la OCATT'*) button

1. Select the organs to be offered on the organs tab and check the offer confirmation

2. Select a test from the list if that test is supporting the offering decision.

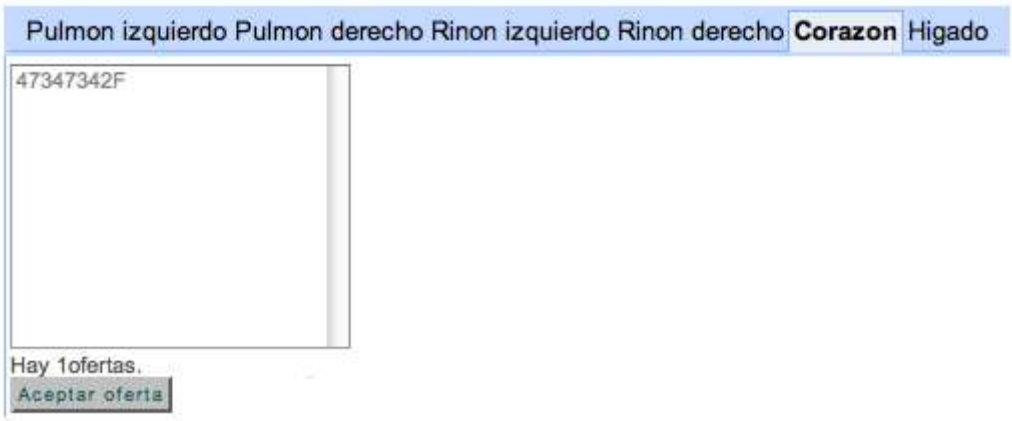


A.11 Assigning an organ to a patient

1. If you are a doctor or an organ transplant management staff of a hospital that has received an organ offer, you can manage such offer by selecting the "Manage Waiting List" suboption (*'Gestionar la lista de espera'*) on the main menu.

2. The recipient list has a list of the patients and a list of the asked organs for each one. Select a patient and an organ.

3. If there has been an offer for that organ to the hospital, you will see a list of the donors.



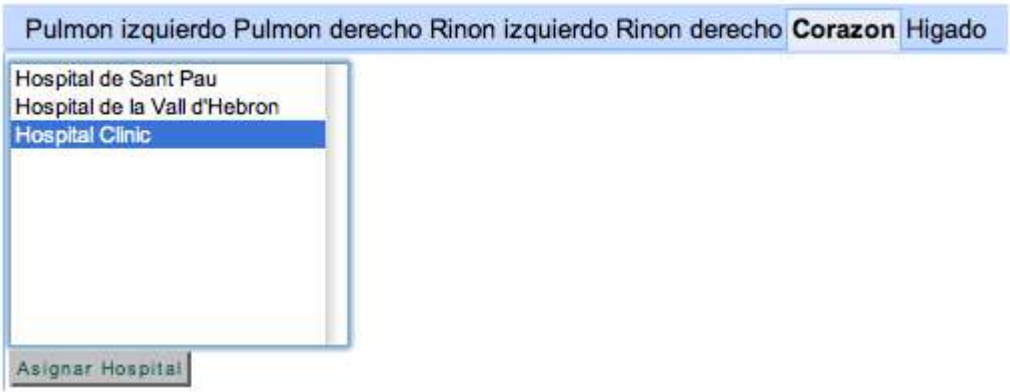
4. In order to accept the offer, click on the "Accept Offer" (*'Aceptar oferta'*) button. The organ will be then assigned to that recipient.

A.12 Selecting the organ destination

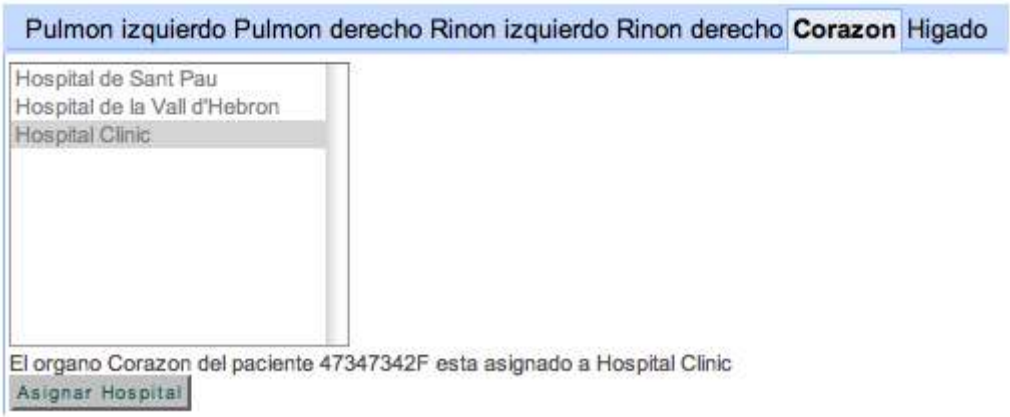
1. If you logged as a member of the OCATT, you will have access to the donor list from all hospitals. Select the "Manage Donors" (*'Gestión de donantes'*) submenu option.

2. You will see the donor list. Select the organ of the patient you want to manage.

3. The list of possible destinations will appear. Select a hospital from the list.



4. Once you click the "Assign Hospital" button *('Asignar hospital'*), the assignment will be sent to that hospital.



A.13 Managing test requests

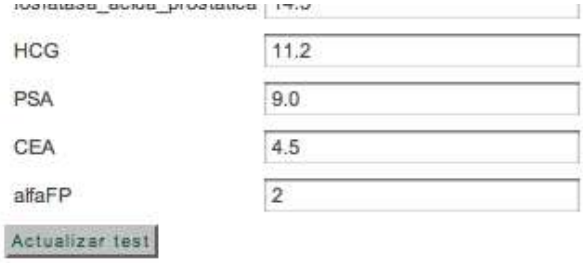
1. When you log in as a laboratory staff member, you can see the list of test requests (*'Listado de peticiones de test'*).

Listado de peticiones de test

Identificador de test	Identificador de paciente	Acabado	Fecha inicio	Fecha final	Institucion
168002925	Py6x7pBB3Lg=	Si	1160187497745	1160192823824	Hospital de Sant Pau
1954070591	46962871E	Si	1160202170859	1160202203369	Hospital de Sant Pau
1321879101	46962871E	Si	1160202231029	1160202254191	Hospital de Sant Pau
1761317462	FE3Vo1DDblg=	Si	1160211301664	1160211326629	Hospital de Sant Pau
1542086776	FE3Vo1DDblg=	Si	1160213051545	1160213696366	Hospital de Sant Pau
701941894	FE3Vo1DDblg=	Si	1160213749373	1160213763979	Hospital de Sant Pau
1253521070	sadffa	Si	1160365108881	1160365196716	Hospital de Sant Pau
1527939601	sadffa		1160552114624	0	Hospital de Sant Pau
1598092898	47347342F		1162273760006	0	Hospital de Sant Pau
1951512169	47347342F		1162273773426	0	Hospital de Sant Pau
1586974207	47347342F		1162273799384	0	Hospital de Sant Pau

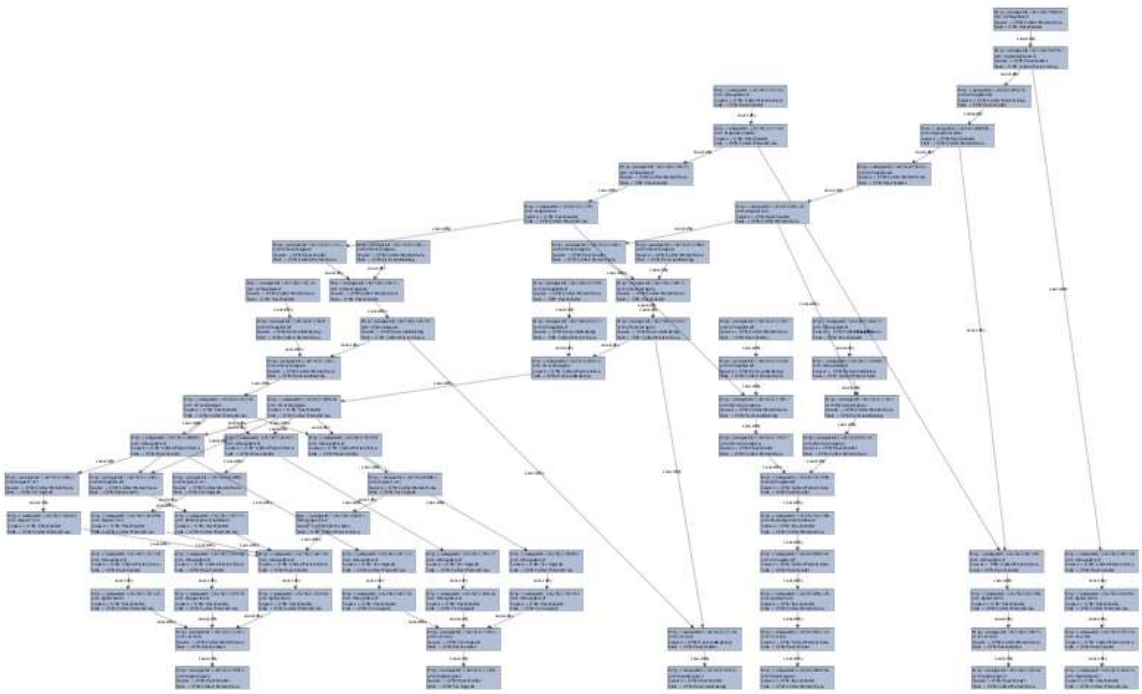
2. Click on an unfinished test request. You will see an empty form with the parameters of the request test type.

3. Fill in the form and click on the "Update Test" (*'Actualizar test'*) button. It will be updated in the system.



A.14 Querying events related to a patient

1. If you are inspecting a patient record in OTM Application, follow the hyperlink to the Provenance Tool.
2. Open the Relationship Portlet to see the graphical result of the query.[3]



3 Check the Provenance Tools user manual to learn more about querying the Provenance Store.

References

[D2.1.1] Árpád Andics (ed.) **D2.1.1: User Requirements Document**. Technical report, MTA SZTAKI, February 2005. http://www.gridprovenance.org/deliverables/GRID_PROVENANCE-UserRequirements-D211-Month6.pdf

[D2.2.1] Árpád Andics (ed.). **D2.2.1: Software Requirements Document**. Technical report, MTA SZTAKI, February 2005. http://www.gridprovenance.org/deliverables/GRID_PROVENANCE-SoftwareRequirements-D221-Month6.pdf

[D3.1.1] Paul Groth, Sheng Jiang, Simon Miles, Steve Munroe, Victor Tan, Sofia Tsasakou, and Luc Moreau. **D3.1.1: An Architecture for Provenance Systems**. Technical report, University of Southampton, November 2006. http://eprints.ecs.soton.ac.uk/13216/

[D6.1.1] Omer F Rana. **D6.1.1: Tools Description Document**. Technical report, University of Cardiff, September 2005. http://www.gridprovenance.org/deliverables/GRID_PROVENANCE-ToolsSpec-D611-Month12.pdf

[D8.1.1] Javier Vázquez-Salceda, Steve Willmott, and Kifor Tamás and László Zs. Varga. **D8.1.1: Application 2: Organ Transplant Management. Specification of Mapping to Provenance Architecture, and Domain Specific Provenance Handling**. Technical report, UPC, September 2005. http://www.gridprovenance.org/deliverables/GRID_PROVENANCE-OTMSpec-D811-Month12.pdf

[D8.2.1] Javier Vázquez-Salceda, Sergio Álvarez, László Zs. Varga, and Kifor Tamás. **D8.2.1: Application 2: Evaluation Report**. Technical report, UPC, November 2006. http://www.gridprovenance.org/deliverables/GRID_PROVENANCE-OTMEvaluationReport-D821-Month27.pdf

[D9.3.3] John Ibbotson and Sheng Jiang. **D9.3.3: Final Functional Prototype**. Technical report, IBM United Kingdom, November 2006. http://www.gridprovenance.org/deliverables/GRID_PROVENANCE-FinalPrototype-D933-Month27.pdf

[D9.3.3a] Sheng Jiang, Luc Moreau, Paul Groth, Simon Miles, Steve Munroe, and Victor Tan. **D9.3.3a: Client Side Library Design and Implementation**. Technical report, University of Southampton, November 2006. http://www.gridprovenance.org/deliverables/GRID_PROVENANCE-ClientSideLibrary-D933a-Month27.pdf

[ENV13606] CEN/TC251 WG I.: Health Informatics-Electronic Healthcare Record Communication- Part 1-4, Final Draft prENV13606-1 (1999-2000).

[FIPA02a] The Foundation for Intelligent Physical Agents "FIPA Abstract Architecture Specification", Specification Number SC00001, December, 2002, available online at http://www.fipa.org/specs/fipa00001/ .

[FIPA02b] The Foundation for Intelligent Physical Agents "FIPA Communicative Act Library Specification", Specification Number SC00037, December, 2002, available online at http://www.fipa.org/specs/fipa00037/ .

[FIPA02c] The Foundation for Intelligent Physical Agents "FIPA Message Structure Specification", Specification Number SC00061, December, 2002, available online at http://www.fipa.org/specs/fipa00061/ .

[Miles 05] Project Internal document "Representing Provenance Data in the OTM application", Simon Miles, Javier Vazquez, July, 2005. http://twiki.gridprovenance.org/bin/viewauth/Restricted/OrganTransplantPStructure (project internal web site).

[RFC1035] P. Mockapetris, "Domain Names – Implementation and Specification", Internet Engineering Task Force RCF 1035, November 1987, available online at http://www.ietf.org/rfc/rfc1035.txt .

[RFC2141] R. Moats, "URN Syntax", Internet Engineering Task Force RFC 2141, May 1997. available online at http://www.ietf.org/rfc/rfc2141.txt .

[W3C05] D. Booth, H. Haas, F. McCabe et. al. "Web Services Architecture, W3C Working Group Note", 11 February 2004, available online at: http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/ .