A Personalised Integrated Care Platform

(Grant Agreement No. 689209)

D5.4 First Integrated Data Management Subset in Public Cloud

Date: 31-08-2017

Version 1.2

Published by the PICASO Consortium

Dissemination Level: Public

Co-funded by the European Union's Horizon 2020 Framework Programme for Research and Innovation under Grant Agreement No 689209

Document control page

Document file: D5.4_First_Integrated_Data_Management_Subset_in_Public_Cloud_v1.2_final.docx
Document version: **1.2**
Document owner: TUK

Work package: WP5 – Privacy Enhanced Integrated Data Management
Task: T5.2 – Shared Memory Manager
Deliverable type: [DEMONSTRATOR]

Document status: ☒ approved by the document owner for internal review
☒ approved for submission to the EC

Document history:

Version	Author(s)	Date	Summary of changes made
0.1	Peter Kostelnik (TUK)	12-01-2017	Initial Content
0.2	Jan Hreno (TUK), Peter Kostelnik (TUK)	22-01-2017	Added Clinician Manager and LinkWatch Integration
1.0	Peter Kostelnik (TUK)	01-02-2017	Comments from Reviewers. Final version ready to be submitted to the European Commission
1.1	Peter Kostelnik (TUK), Marek Skokan (TUK)	30-08-2017	Updates reflecting the demonstrator in the context of current PICASO architecture
1.2	Peter Kostelnik (TUK), Marek Skokan (TUK)	31-08-2017	Comments from internal review. Final version ready to be submitted to the European Commission

Internal review history:

Reviewed by	Date	Summary of comments
Matts Ahlsen (CNET)	31-01-2017	Accepted with minor comments and discussion.
Evangelos Vlachogiannis (FIT)	01-02-2017	Accepted with minor comments and discussion.
Matts Ahlsen (CNET)	31-08-2017	Accepted with minor comments

Index:

1 Executive Summary

One of the core ideas of the PICASO project is to enable the uniform access to the clinical data distributed across various clinical systems in hospitals or personal patient monitoring data. WP5 is dedicated to design and implement the components enabling the joining and fusion of such a distributed data. This deliverable describes two demonstrators illustrating, (1) example, how data management components are implemented and (2) example, how data fusion provided by data orchestration is used in Clinician Manager. The next versions of data management components will be described in D5.6 and D5.7.

Both demonstrators include the core Data Management Components, the Metadata Registry and Data Orchestration.

2 Introduction

2.1 Purpose, context and scope of this deliverable

The purpose of this deliverable is to demonstrate the two example cases of Picaso Distributed Data Management, implementing the design described in deliverable D5.2 Shared Memory Manager.

2.2 Content and structure of this deliverable

Section 3 describes the basic use case of Data Management demonstrator. Demonstrator illustrates the full implementation of Distributed Data Management architecture and data fusion.

Section 4 describes the demonstrator presented at the first review. Demonstrator implements the basic use case of using LinkWatch measurements from remote patient monitoring, which are dynamically visualized in Clinician Manager.

Both demonstrators fully implement the architecture and design described in D5.2 Shared Memory Manager.

3 Data Management Demonstrator

3.1 Architecture and Implementation

The purpose of demonstrator is to illustrate the overall usage of PICASO Distributed Data Management approach. Design and implementation of Data Management Demonstrator follows the architecture design specified and described in deliverable D5.2 Shared Memory Manager. Overall demonstrator architecture is illustrated in figure 1.

Figure 1: Overall demonstrator architecture.

Demonstrator consists of four separately running components:

- Simulated Picaso Care System Public Cloud – CSPUC, containing Metadata Registry keeping track of all patient data.
- Two simulated Picaso Care System Private Cloud – CSPRC 1 & 2. Both Data Node applications, each running as separate CSPRCs simulate the simple clinical applications enabling to insert the patient related data. Data Node applications serve as the source of patient data.
- Simulated CSPRC 3 containing the application for reading the patient data inserted via Data Node applications. Data related to selected patient are the product of fusion of data retrieved from Data Node applications with support of Metadata Registry in public cloud and locally running Data Orchestration components in related private clouds.

3.1.1 Data Node in CSPRC

Demonstrator contains one CSPRC implementation for inserting the patient data, which is deployed twice, with different configuration, to simulate two independent local Picaso clinical systems. CSPRC application for inserting patient data is referred as Data Node application.

Data Node application consists of three components:

- Picaso App. Application enabling the entry of patient data. There are three available predefined patient identifiers (PID-1, PID-2 and PID-3), each representing specific patient. Data Node enables to insert three types of example entry: DIAGNOSIS, OBSERVATION and DICOM for selected

patient identifier. DIAGNOSIS and OBSERVATION are just simple identifiers of the entry type, DICOM entry contains also the pointer to three example samples of real DICOM data. Once the entry is inserted, the central Metadata Registry is notified with the new metadata representing the entry and it is stored in local CSPRC storage represented by simulated ODS stub.

- ODS stub is implemented as simple file-system based storage containing and providing all entries inserted via CSPRC Picaso Application. ODS stub also contains the DICOM images for three example DICOM MRI samples. All data stored in ODS stub are available via related local Data Orchestration Component.
- Data Orchestration Component is responsible for retrieving the data stored in related local ODS stub to applications requesting the data from this local CSPRC. Data Orchestration components deployed in CSPRCs provide capability of serialization of information into XML and JSON formats.

Data Node applications are implemented as Java Web Applications using REST interface implementing the local Data Orchestration API for retrieving the data from local ODS stubs.

3.1.2 Metadata Registry in CSPUC

The purpose of Metadata Registry is to keep track of all data inserted via all local clinical systems. Metadata Registry is central component deployed in public cloud, available to all Data Orchestration components in Picaso platform.

Metadata Registry was designed to be able to use any underlying storage for metadata. In this demonstrator, Metadata Registry storage is implemented using GraphDB [GraphDB, 2016] semantic repository.

Metadata Registry is implemented as Java Web application available via REST API interface for inserting and querying the metadata storage.

3.1.3 Data Fusion App in CSPRC

The Data Fusion App (CSPRC 3: App in Figure 1) consists of two components:

- Picaso Application enabling to retrieve all patient related data by selecting the patient identifier (PID-1, PID-2 and PID-3). For each request, this application demonstrates the fusion of from both Data Node applications via its local Data Orchestration component. For each request, this application shows:
  - All results for selected patient identifiers. The results are retrieved from related Data Node applications and interpreted – transformed into human readable form. For DICOM samples, the related results show also the interpretation of DICOM images, which can be viewed and browsed.
  - The action log containing the list of all actions performed by underlying logic, including the Metadata Registry query and calls for real data related to each matching metadata record. Action log contains the links to the API calls of related Data Orchestration components.
- Data Orchestration component responsible to physically collect and fuse the data related to selected patient.
  - Data Orchestration component queries the Metadata Registry for all metadata matching selected patient identifier. Matching metadata are ordered in time.
  - For each metadata record in obtained results, the real source of information – the related Data Node Data Orchestration component – is asked for real data represented by metadata record.

- Retrieved chunks of real data are interpreted and rendered. All action performed by Data Orchestration are logged in action log, which is also shown in application view. Note, the logs will be passed to the Activity Log component in next version of Data Orchestration.

Data Fusion Application is implemented as Java Web Application.

3.2 Demonstrator usage

3.2.1 Data Node Application in CSPRC

Data Node application provides very simple interface to insert the patient data. Interface is illustrated in Figure 2. Once the entry is saved, the confirmation view (see Figure 2) contains the links to related metadata record and real data stored in local ODS stub.

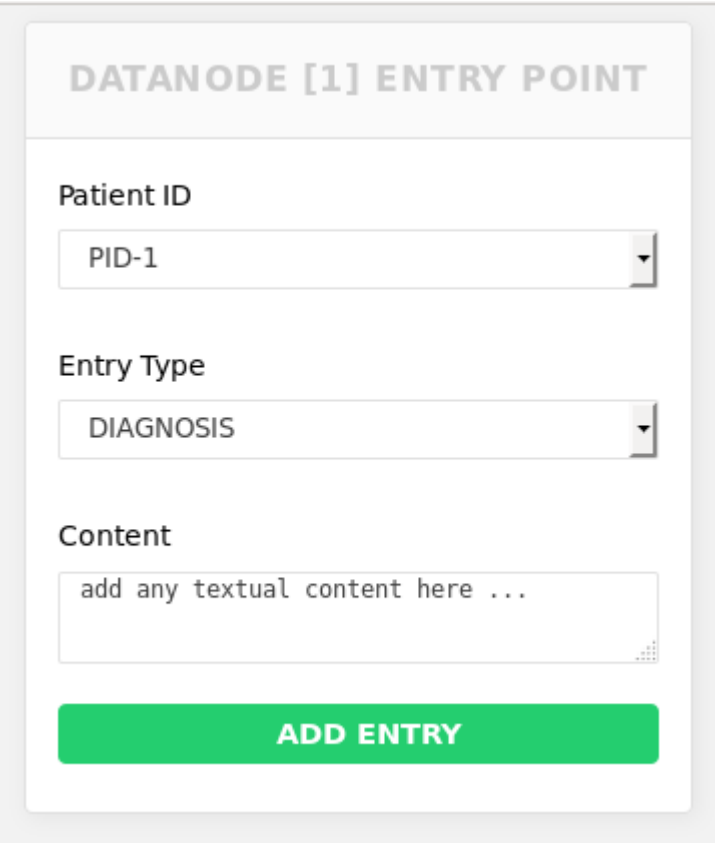

Figure 2: Example of data entry interface.

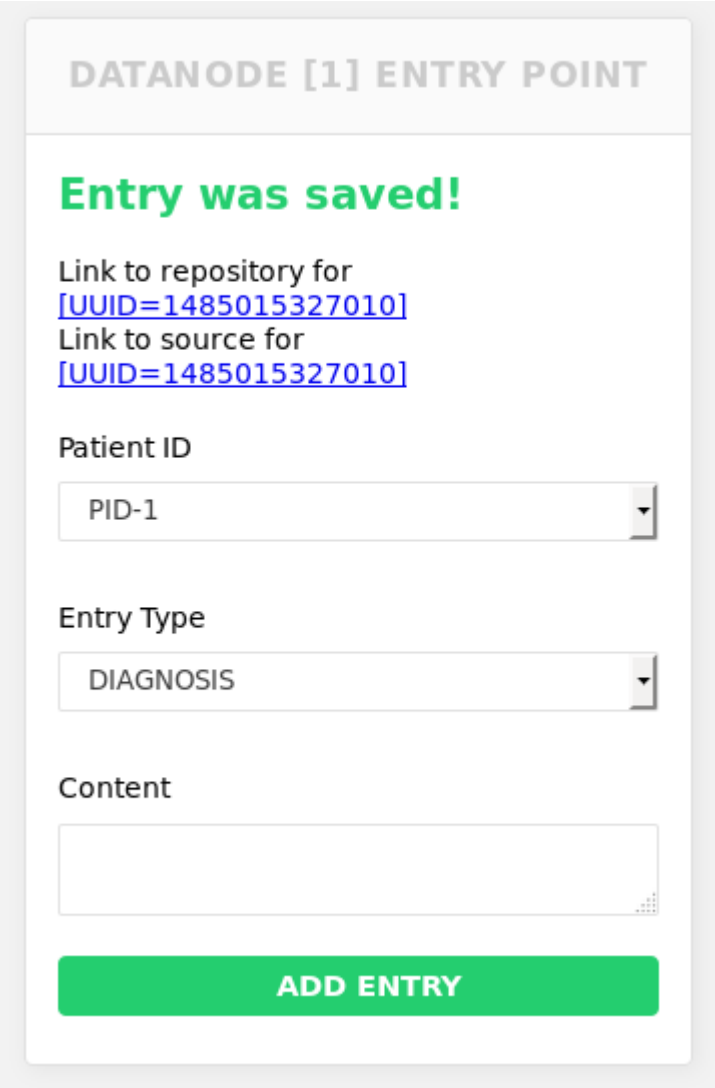

Figure 3: Example of data entry confirmation view.

Link to the repository is the link to Metadata Registry API returning the reference to related metadata record. By clicking the link, the user receives the full machine readable semantic information related to this record, which can be hard to read. The metadata record in form of N3 tripples can be simplified in more human readable way as follows:

picaso:1-1485015918215

rdf:type picaso:PatientDiagnosisData ;

picaso:patientId "PID-1”^^xsd:string ;

picaso:timestamp “1485015918215”^^xsd:long ;

picaso:source http://picaso.eu/example#data-source-1 .

Generally, each metadata record contains the patient identifier, type of entry (diagnosis, observation, DICOM), reference to source and timestamp.

The data entry is stored in JSON format as follows:

{"data": {"entry": {

```
  "created": "2017-01-21T17:25:18",
  "patient": {"id": "PID-1"},
  "entry-type": "DIAGNOSIS",
  "content": "example diagnosis entry"
}}}
```


3.2.2 Data Fusion Application in CSPRC

The purpose of Data Fusion Application is to retrieve and interpret all information related to selected patient identifier. The application view contains simple interface for selecting the patient identifier and the results view. The results view consists of interpreted results collected from all related Data Node applications ordered by timestamp and the action log listing all actions and REST API calls performed by local Data Orchestration component (to Metadata Registry and then to Data Orchestration components related to each processed metadata record).

Each result contain the link to the original real data retrieved from related source exported in JSON and XML. Links point to Data Orchestration components API in source CSPRC of the result. The example of results retrieved for selected patient identifier are illustrated in Figure 4.

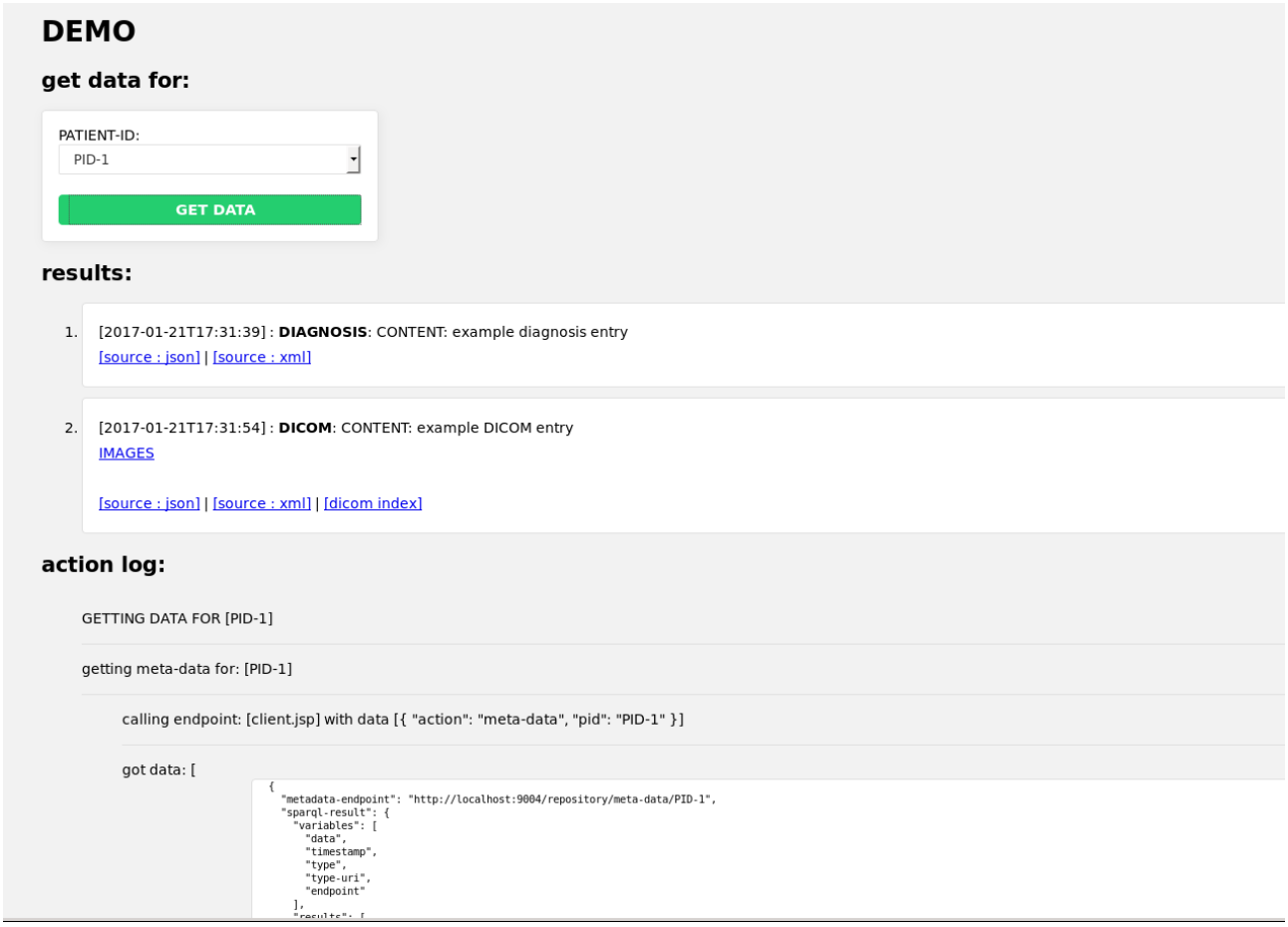

Figure 4: Example of interpreted results of data fusion.

If the result contains the DICOM data, clicking the IMAGES link shows the browser of all DICOM images in the sample related to this entry. Each picture contains the links to the original DICOM standard representation of picture and the link to the original source in related CSPRC. DICOM entry result also contains the link to the original index of sample images. DICOM images browser is illustrated in Figure 5.

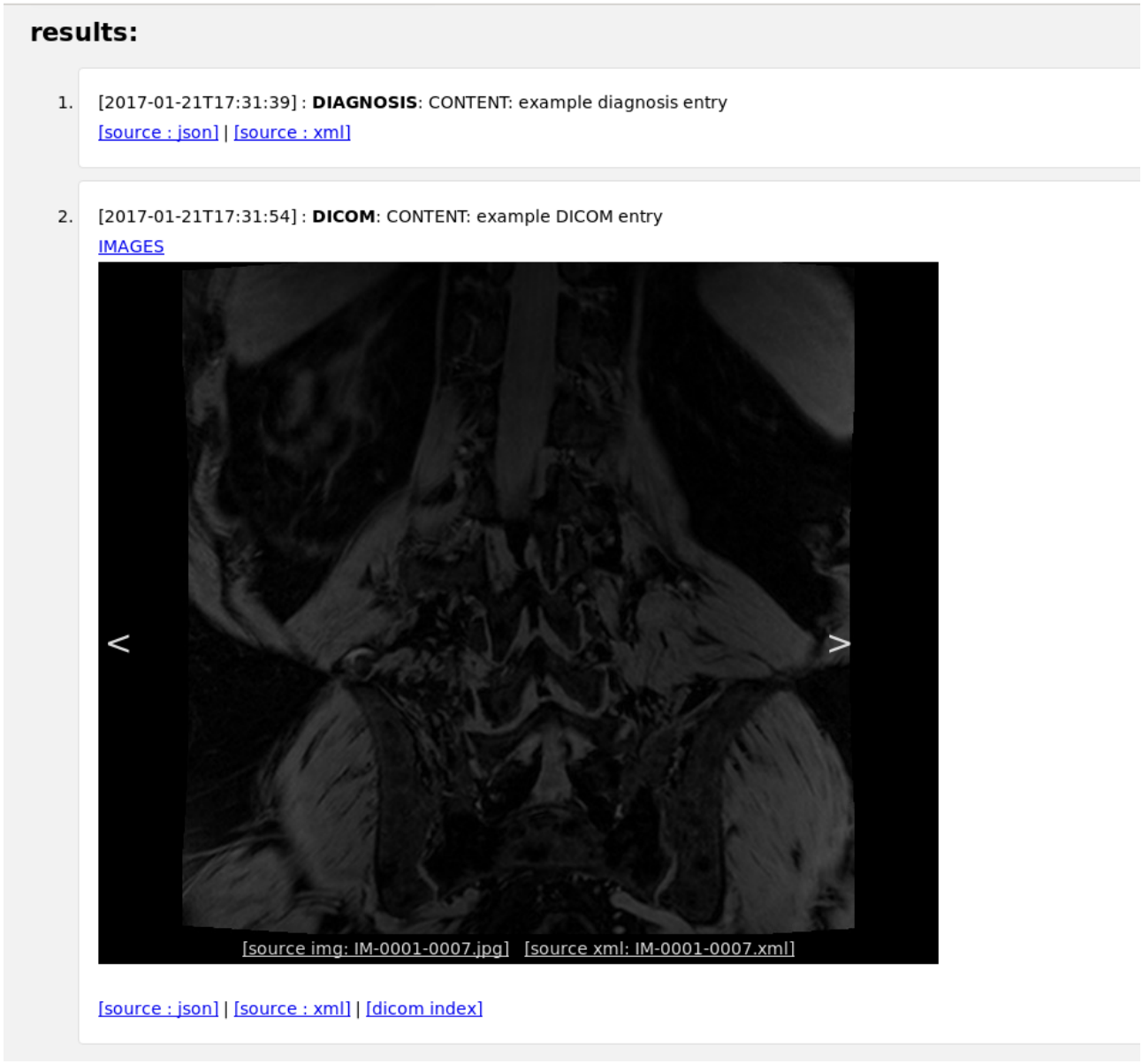

Figure 5: Example of interpreted browsable DICOM images sample.

4 LinkWatch measurements in Clinician Manager Demonstrator

4.1 Architecture and Implementation

The purpose of demonstrator is to illustrate the integration of Clinician Manager and LinkWatch remote patient monitoring measurements via Data Management Components. Design and implementation of demonstrator follows the architecture design specified and described in deliverable D5.2 Shared Memory Manager and D6.2 First Decision Support and Interaction Tools. Overall demonstrator architecture is illustrated in figure 6.

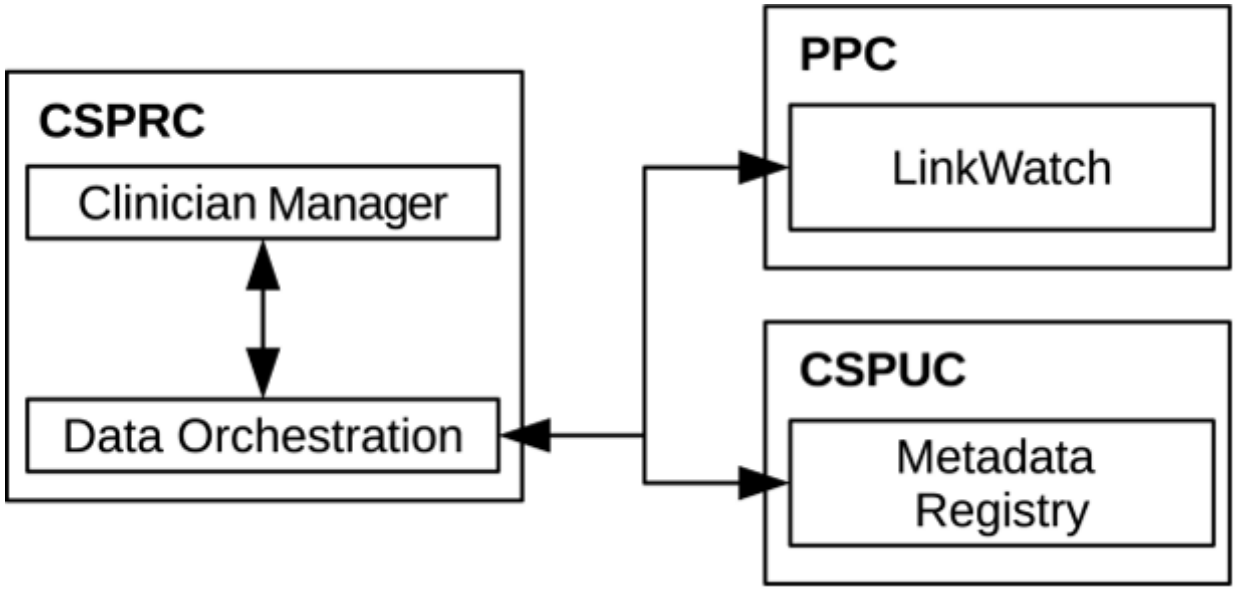

Figure 6: Architecture of Clinician Manager and LinkWatch demonstrator.

Demonstrator consists of four separately running components:

- Simulated Picaso Care System Public Cloud – CSPUC, containing Metadata Registry keeping track of all patient data.
- Simulated Picaso Care System Private Cloud – CSPRC, containing the Clinician Manager serving as user interface and Data Orchestrator used to retrieve the requested LinkWatch data.
- Simulated Patient Private Cloud - PPC containing the testing LinkWatch instance.

4.1.1 Simulated CSPRC

Simulated CSPRC contains two components:

- Clinician Manager – the visualization tool, described in more details in D6.2 First Decision Support and Interaction Tools. The purpose of Clinician Manager demonstrator was to mock up the ideas of Clinician user interfaces and interactions with his/her Picaso tools. The second purpose was to test the ideas of integration with home monitoring component providing clinical home measurement data. The Clinician Manager provides the graphical overview of health-related home monitoring measurements. Clinician Manager requests the local Data Orchestration component to retrieve the patient profile (patient basic information) data and the measurements data. This information is visualized in the graph.

- In its first iteration the Data Orchestration providing API services to retrieve the patient profile and LinkWatch measurements using LinkWatch API services. For each request, the Data Orchestrator queries the Metadata registry for records representing required information, then it calls the LinkWatch API services deployed in simulated PPC to physically retrieve the data and exports it in machine readable format accepted by Clinician Manager.

To satisfy the data requests of Clinician Manager, Data Orchestrator implements two API services, directly communicating with LinkWatch API. The primary purpose of the demo is to show the behaviour of Clinicial Manager. However, in architecture, the LinkWatch measurements will be available via local ODS storage related to particular LinkWatch instances. As in time of implementation of this demonstrator, the ODS implementation was not available (it was in its population state), the demonstrator communicates with LinkWatch instance directly. Note, the next iteration of Data Orchestration will consult ODSes only and such state will be described in next version of the deliverable D5.6 Second Integrated Data Management Subset in Public Cloud.

Local Data Orchestration API implements two services, described below.

Get patient profile

Returns the profile of the patient directly from LinkWatch endpoint available at:

https://linkwatchrestservicetest.azurewebsites.net/api/v1/patient

This service consumes the patient token to identify the patient. Result produced by this endpoint is serialized as JSON:


```
{
  "ManualDevice": true,
  "UserAccountId": 2,
  "Smoker": true,
  "Patient": {
    "DateOfBirth": "1960-01-01T00:00:00",
    "Status": "Active",
    "PatientId": "11892829",
    "Height": 152,
    "Gender": "",
    "Age": 57
  },
  "ManualInput": true,
  "Diabetes": 2,
  "Person": {
    "Email": "peter.rosengren@cnet.se",
    "Address": null,
    "FirstName": "Robotnik",
    "Phone": null,
    "LastName": "Moskowitch",
    "MobilePhone": null
  }
}
```


Get patient data

The Data Orchestrator service requests the Metadata Registry to retrieve all sources of related LinkWatch instalations. This demo contains metadata record for example patient and one LinkWatch endpoint, however, it is possible to simulate multiple sources of LinkWatch data.

Metadata are processed and for all LinkWatch sources identified in metadata, the query is executed, data are processed and serialized in machine readable format accepted by Clinician Manager. Then, Clinician Manager visualises the data in graph.

Data Orchestration component contains the full model of data available via LinkWatch instance. LinkWatch provides data from multiple devices, each device may return particular measurement types and each measurement type may return the data in several units. Each device, measurement type and unit is represented via unique identifier. The full model of LinkWatch measurement data with identifiers is listed in table 1.

Table 1: Devices/Measurement/Unit model

Device	Measurement type	Unit
BloodPressure [528391]	• Systolic [150021] • Diastolic [150022] • PulseRate* [149546]**	• MmHg [266016] • mmHg [266016] • bpm [264864]
PulseOximeter [528388]	• Saturation [150456] • PulseRate* [149530]**	• % [262688] • bpm [264864]
WeighingScale [528399]	• Weight [188736]	• g [263875] ; pound [263904]
GlucoseMeter [528401]	• Glucoe Wholeblood [160184]	• mmol/l [266866]; mg/dl [264274]
StepCounter [528488]	• Steps [8454260] • ActiveTime [68185]	• Step [268800] • sec [264320] ; min [264384] ; hour [264352] ; day [264416]
SleepQuality [528408]	• Sleep [8455148] • ActiveTime [68185]	• min [264384] ; hour [264352] • min [264384] ; hour [264352]
PeakFlow [528405]	• PEF [152584] • FEV1 [152586] • FEV6 [152175]	• liter/min [264992] • liter [263744] • liter [263744]
RespiratoryRate [528397]	• Resp_rate [151562]	• breaths/min [264928]
HeartRate [528525]	• HeartRate [8454258]	• bpm [264864]

The patient data service accepts the following parameters to specify the query:

- patient token to identify the data
- the list of measurement types, in demonstrator, it was agreed to visualize patient measurements by measurement type, each measurement type specified in service execution is retrieved from all devices providing specified measurement type, the devices providing requested measurement types are identified using the model in table 1.
- date-time interval to specify the timespan of required measurements

The final result is the product of data fusion from several executions of LinkWatch endpoint providing the observation data for specified device and time interval. The LinkWatch endpoint providing the device data is available at:

https://linkwatchrestservicetest.azurewebsites.net/api/observation/value/deviceId/between

Retrieved results are:

- grouped by the LinkWatch source identified in metadata
- the device model, that provided requested measurement type; for each device, there is attached the machine readable metamodel of this device including all information listed in table 1, to enable automatic interpretation of measurements data (e.g. human readable information, transformation of units, etc)
- for each device, there is attached the list of measurements ordered by timestamp

The example of retrieved result returned to Clinician Manager serialized in JSON:


```
[
  {
   "source-data" : [
     {
      "device-data" : {
       "device-model" : {
        "device-name" : "BloodPressure","device-measurements" : [
         {
          "measurement-name" : "Systolic","measurement-units" : [
          {"unit-name" : "mmHg","unit-id" : "266016"}
         ],"measurement-id"   : "150021"
         },
         {
          "measurement-name" : "Diastolic","measurement-units" : [
          {"unit-name" : "mmHg","unit-id" : "266016"}
         ],"measurement-id"   : "150022"
         },
         {
          "measurement-name" : "PulseRate","measurement-units" : [
          {"unit-name" : "bpm","unit-id" : "264864"}
         ],"measurement-id"   : "149546"
         },
         {
          "measurement-name" : "PulseRateOxim","measurement-units":[
          {"unit-name" : "bpm","unit-id" : "264864"}
         ],"measurement-id"   : "149530"
         }
        ],"device-id" : "528391"
       },
       "data"       : [
        {"Typeid"      : 528391,"Comment" : null,"Measurements" : [
         {"TypeId" : 150021,"UnitCode" : 266016,"Value" : "115"},
         {"TypeId" : 150022,"UnitCode" : 266016,"Value" : "85"}
        ],"Equipmentid" : "00091FFFFE8019AC","Id" : 311676,
         "Inputtype" : "Web Service",
         "Timestamp" : "2017-01-20T05:33:31.000Z",
         "Contexts" : null
        },
        {
         "Typeid"      : 528391,"Comment" : null,"Measurements" : [
         {"TypeId" : 150021,"UnitCode" : 266016,"Value" : "118"},
         {"TypeId" : 150022,"UnitCode" : 266016,"Value" : "80"}
        ],"Equipmentid" : "00091FFFFE8019AC","Id" : 311851,
         "Inputtype" : "Web Service",
         "Timestamp" : "2017-01-22T12:32:00.000Z",
         "Contexts" : null
        }
       ]
      }
     }
   ],"endpoint"  : "https://linkwatchrestservicetest.azurewebsites.net",
   "source-token" : "link-watch-demo"
  }
]
```

4.1.2 Metadata Registry in CSPUC

The purpose of Metadata Registry is to keep track of all data inserted via all local clinical systems. Metadata Registry is central component deployed in public cloud, available to all Data Orchestration components in Picaso platform.

Metadata Registry was designed to be able to use any underlying storage for metadata. In this demonstrator, Metadata Registry storage is implemented using MySQL database containing the records notifying that there exist LinkWatch measurement data for the patient.

The metadata records providing information about the LinkWatch sources related to testing patient were inserted manually.

Metadata Registry is implemented as Java Web application available via REST API interface querying the metadata storage.

4.1.3 LinkWatch in PPC

In demonstrator, there was used the publicly available testing instance of LinkWatch deployed at https://linkwatchrestservicetest.azurewebsites.net, providing data for one testing user. LinkWatch API services to retrieve the patient measurement data are described in:

https://confluence.fit.fraunhofer.de/confluence/display/PIC/LinkWatch+Measurement+DB+API

4.2 Demonstrator usage

The demonstrator provides the Clinician Manager user interface providing the views for actualization of patient profile.

In left menu, click the *Patient Overview* link to change the view.

Reload button in *Patient Info* view actualized patient profile from LinkWatch endpoint.

Patient Monitoring Overview view provides the interface for visualization of predefined measurement types: Systolic (mmHg), Diastolic (mmHg), Pulse Rate (bpm) and Weight (kg). This data can be actualized in time interval specified using date-time input field below the graph by clicking the *Reload for selected dates* button. Once button is clicked, data are actualized from running LinkWatch instance.

The Clinician Manager views are illustrated in figure 7.

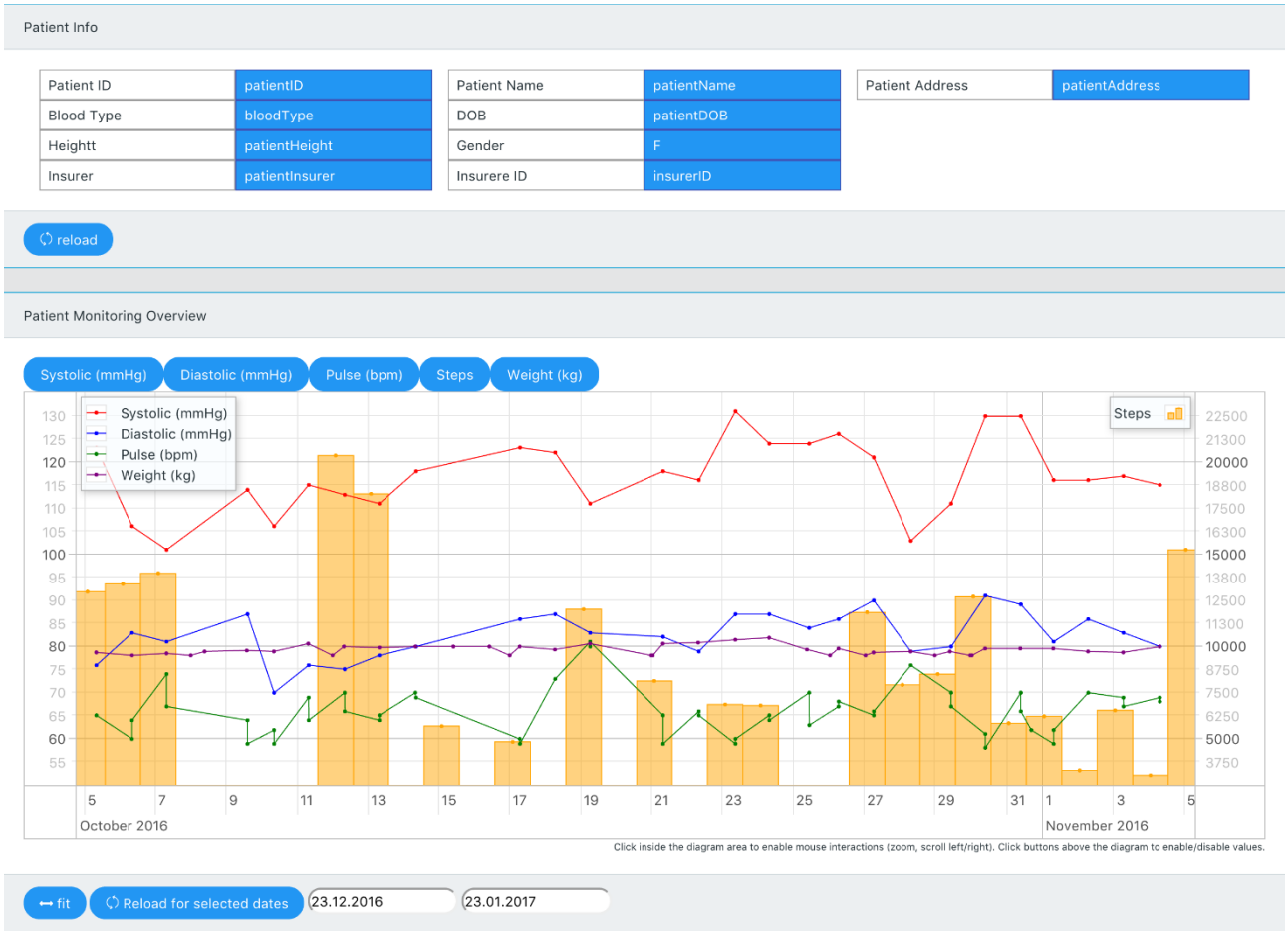

Figure 7: Clinician Manager user interface.

4.3 Running the demonstrator

Demonstrator is publicly available at the endpoint:

http://147.232.202.111/#/clinician-manager/all/all

5 List of Figures and Tables

5.1 Figures

5.2 Tables

6 References

(GraphDB, 2016) Ontotext GraphDB, http://ontotext.com/products/graphdb/