An Event-driven Health Service Bus

Despina T. Meridou
School of Electrical and Computer
Engineering, NTUA,
Heroon Polytechniou 9, 15773,
Athens, Greece
dmeridou@icbnet.ece.ntua.gr

Andreas P. Kapsalis
School of Electrical and Computer
Engineering, NTUA,
Heroon Polytechniou 9, 15773,
Athens, Greece
akapsalis@icbnet.ece.ntua.gr

Panagiotis Kasnesis
School of Electrical and Computer
Engineering, NTUA,
Heroon Polytechniou 9, 15773,
Athens, Greece
pkasnesis@yahoo.gr

Charalampos Z. Patrikakis
Department of Electronics
Engineering, TEI of Piraeus,
Petrou Ralli & Thivon 250, 12244,
Egaleo, Greece
bpatr@teipir.gr

Iakovos S. Venieris
School of Electrical and Computer
Engineering, NTUA,
Heroon Polytechniou 9, 15773,
Athens, Greece
venieris@cs.ntua.gr

Dimitra-Theodora I. Kaklamani
School of Electrical and Computer
Engineering, NTUA,
Heroon Polytechniou 9, 15773,
Athens, Greece
dkaklam@mail.ntua.gr

ABSTRACT

The enormous set of health and wellbeing data sources, as well as the diversity of the data, calls for an effective, time-aware integration paradigm that aids at the manipulation of the information by experts as a whole and not as individual pieces of knowledge. In this paper, we present the Health Service Bus, a service-based platform built on top of the Enterprise Service Bus architecture. Treating new information, either human-generated (e.g., doctors, dieticians, etc.) or device-generated (i.e., smart wristbands or connected scales) as events allows for in-time action and treatment. Platform interoperability is ensured both on service level, since any service irrespective of its specification can be plugged into the Health Service Bus seamlessly, and on data level, since health standards, such as HL7 FHIR and LOINC, are leveraged.

Categories and Subject Descriptors

I.2.1 [**Artificial Intelligence**]: Applications and Expert Systems – *medicine and science*; I.2.4 [**Artificial Intelligence**]: Knowledge Representation Formalisms and Methods – *semantic networks*; J.3 [**Computer Applications**]: Life and Medical Sciences – *medical information systems*

General Terms

Management, Performance, Design, Standardization.

Keywords

Health service bus; Enterprise service bus; Quad store; Health and lifelogging data ontology.

1. INTRODUCTION

The increasingly vast number of health-related connected devices and applications has led to the emerging need for a data integration paradigm, enabling the overall processing of health and well-being data on the basis of remote healthcare and home recovery. At the same time, in order to ensure the completeness of the patient's profile, the medical history, maintained by hospitals and medical centers, needs to be incorporated. In this paper, we present an Enterprise Service Bus based platform, named "Health Service Bus", which is able to aggregate health data for a particular user/patient and maintain them in a semantic format, ensuring seamless homogeneity and interoperability. New data entering the platform are ultimately treated as events, achieving on-the-fly access by any interested application, service or medical center. We assume that data are transferred by applying the HL7 FHIR standard and that they arrive at the platform expressed using the LOINC standard and then they are transformed in RDF based on a target vocabulary. Any interested party can take advantage of the initial and the transformed data by sending a request to the appropriate service of the Health Service Bus.

The rest of the paper is organized as follows: In Section II, work related to the Health Service Bus is discussed. Section III presents the architecture of the platform, whereas in Section IV the data storage and management techniques adopted in the Health Service Bus are described. Section V presents the client functionality offered by the platform and, finally, in Section VI, we summarize the conclusions and remarks of our work, as well as issues for future study.

2. RELATED WORK

In this paper, our view is to examine the use of an intelligent Enterprise Service Bus (iESB) [1] in the field of healthcare. The iESB used as a basis for the design of the proposed platform has been based on the service bus developed within the FP7 European Project ARUM [2], focusing in production management. The iESB of ARUM is a service-based, multi-agent platform for managing the production of highly-customized products, such as aircrafts and ships, especially in the ramp-up and small-lot modes. The service bus is implemented on top of JBossESB, an open-source Enterprise

Service Bus in Java programming language. The services that are deployed in the iESB provide scheduling and planning functionality, as well as functions of ontological data management and security. While in ARUM data are coming from the legacy systems of companies (e.g., MES and ERP systems), we are arguing that the same architecture could be used to serve the needs of a healthcare management system based on the collection of data from wearable and other smart devices.

The idea of using a service bus for healthcare management is not new: IBM has proposed in the past the use of IBM Healthcare Service Bus [3], a platform for enabling the integration of multiple healthcare-related services. The deployed services may belong to the category of consumers, such as applications that request a service, or to the category of providers, such as applications that provide a requested service. Each consumer or provider is assigned to a Binding Component, which collects all requests, transforms them to an internal format, understood by all the Binding Components of the system and then forwards it to a Normalized Message Router, through which the message eventually arrives to the appropriate producer or consumer. The integration of heterogeneous services is accomplished with the use of the WSDL, SOAP, and HL7 standards.

In [4], a Health Service Bus, based on the Mule open source ESB, is presented. Each application or service connected to the service bus is considered an Abstract Endpoint. The physical implementation of each Abstract Endpoint is called a Service Container and provides the interface of the service to the service bus. A translation service is responsible for transforming all health-related data in XML format. During the transformation process, from and to HL7 V2, HL7 V3 and OpenEHR, an ontology-based mapping tool, named OWLmt, is used.

While the above described healthcare-related platforms are service-based and aim at the interoperability of heterogeneous health services, they do not leverage the major advantages of the semantic languages OWL and RDF, which play a key role in the seamless integration of disparate data. Even though Ryan and Eklund [5] make use of ontologies, their use is limited to the data transformation process. Above all, in our approach, we can take advantage of the inference mechanisms supported by semantic languages and reason over the transformed healthcare and well-being data.

3. GENERAL DESCRIPTION OF THE HEALTH SERVICE BUS PLATFORM

The **Health Service Bus** platform has been designed having in mind three directives: (a) the idea of treating new data related to the health and well-being of a user of the platform as events arriving at a publish/subscribe endpoint, easing their effective handling and sharing within services deployed in the platform as well as within services and tools external to the platform, such as custom mobile applications, healthcare systems, etc., (b) the ability to easily connect and integrate with existing and future personal health monitoring systems and tools, including both single device solutions such as wearable devices that monitor specific living conditions and parameters such as calories burned, and middleware solutions capable of providing comprehensive Personal Health Records and (c) the capability of easily extending the connectivity of the presented platform, and linking it to similar platforms, taking advantage of the capabilities of existing decision support systems for Health, and

allowing professionals and practitioners to connect and make use of it, through the development of the necessary tools and
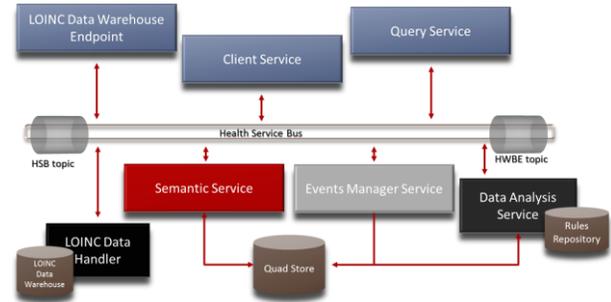
Figure 1. The HSB architecture

interfaces. The above have been reflected in a service-oriented platform, named "Health Service Bus", as depicted in Figure 1.

As described in the following sections, health and well-being data are collected for each platform user from wearable devices, such as smart watches, Bluetooth-enabled toothbrushes, smart pills with edible microchips, etc. We assume that data collected from external to the platform devices are formatted on the basis of the universal code system defined by LOINC. Once the LOINC data have entered the platform, they are semantically annotated on the basis of a well-defined knowledge base, namely the HLD Ontology described in Section IV.

Data produced by the HSB services, such as alerts on elevated blood pressure, on the basis of the received data are treated as events and sent to a dedicated topic of the Health Service Bus. All services are subscribed to this topic, named "Health and Well-being Data Events (HWDE)" topic, this way receiving immediately and being able to process newly fetched information.

The HSB services are able to communicate with each other with respect to internal issues in two ways. For point-to-point communication, a JMS queue is defined for each service. For example, when the Query Service wants to send a SPARQL request to the Semantic Service, it has to "place" the appropriate message to the queue of the latter. For the services to be able to communicate in a publish/subscribe fashion, a second topic has been defined, namely the Health Service Bus (HSB) topic. In this course, when new LOINC data arrive at the platform, the LOINC Data Handler, which acts as the entry point for the LOINC data, sends a message to the HSB topic through the Events Manager Service, notifying all the subscribed services.

4. DATA STORAGE AND MANAGEMENT

Data storage in the platform consists of the LOINC Data Warehouse (LDW), the Quad Store and the Rules Repository. As described below, the LOINC Data Handler is handling the LDW, while the Semantic Service, the Events Manager Service and the Data Analysis Service have direct access and update rights with respect to the Quad Store data. Finally, the Rules Repository is a Jena-based data structure of semantic rules, which are exploited by the Data Analysis Service for inference purposes.

We assume that data arriving at the platform have the form of FHIR messages, thus ensuring platform interoperability. Even in the cases, where the data collected by devices or connected

middleware do not comply with the FHIR and LOINC standards, translator modules from the proprietary format to FHIR and LOINC compliant formats will be used, making use of the device vendor's API. These translator modules are implemented as part of the LOINC Data Handler.

The **LOINC Data Handler** is the main entry point of the platform regarding new data. New data are collected directly from wearable and environmental devices and are assumed to be fully FHIR and LOINC compliant. The LOINC Data Handler component stores the data in their initial form in a document-type database. The motivation behind the selection of NoSQL databases was primarily performance and the semi-structured form of the data. Once newly arrived data are stored in the database, the LOINC Data Handler pushes a new event in the HSB Topic of the Health Service Bus via the Events Manager Service. At the same time, it sends a message to the queue of the Semantic Service in order to inform the service of the new batch of data that is available for transformation.

After the LOINC data have been stored in the NoSQL database, they are sent to the queue of the **Semantic Service** by the LOINC Data Handler. The role of the Semantic Service is twofold. On the one hand, once it receives the data in their initial format, it has to semantically annotate them by exploiting predefined mappings between the fields that a message may contain and the schema of the HLD ontology. In this way, we are moving from raw to meaningful data. At the same time, ubiquitous and seamless integration of data coming from multiple devices of different vendors is achieved. This processing stage includes transforming the LOINC data into RDF quads, this way organizing the RDF dataset in user-oriented graphs. Thereby, processing and querying the data for a specific user becomes faster and more effective. On the other hand, the Semantic Service is responsible for providing access to the semantic data, since not all HSB services have direct access rights to the data store (e.g., Query Service). Once the transformation is completed, the Semantic Service stores the meaningful data in a Jena-based Quad Store. Access to the Quad Store is realized by means of an API, enabling the application of generic queries expressed in SPARQL, the query language for semantic data, as well as of more user-specific queries by leveraging predefined functions (e.g., API methods for acquiring a user's medical history). Finally, the semantic data are linked to the respective records of the LOINC Data Warehouse, enabling, this way, to access and process the data in their initial form,

when necessary (e.g., in case a third-party entity, such as a hospital, wants to get access to the health data of a patient).

The target schema, according to which the semantic data are produced, is entitled "**Health and Lifelogging Data - HLD**" Ontology and is depicted in Figure 2. The HLD Ontology defines concepts relevant to health and lifelogging data aggregated for a person by a specific device. In this respect, each individual piece of information arriving at the LOINC Data Handler is represented by an individual of the *Observation* class and is linked to the appropriate individual of the *Person* class, denoting the data owner. Each Observation instance is linked to an *ObservationType*, defining this way the semantic type of the observation (e.g., blood pressure, heart rate, etc.). An individual of the ObservationType class is associated with up to three values, reflecting the defined *lower*, *normal* and *higher value*. The Observation class is also linked to the collected value through the *hasObservationValue* property, as well as the unit accompanying the value through the *hasObservationUnit* property, a timestamp through the *hasTimeStamp* property and the *Alert* class, which is instantiated in case of the need of an alert through the HWDE topic. An Observation is linked to the *DeviceAgent* that generates it and may happen while the data owner is occupied with an *Activity*, such as running or sleeping. Finally, the information captured for a given person includes any recorded symptoms through the *Symptom* class and the *hasSymptom* property, *medical conditions* (e.g., hearing impairment) that he may suffer from, the data owner's *age*, *sex* and *body measurements*, as well as any *goals* they may have set, such as loss of weight, body fat or the total number of calories burned during a day. As far as the body measurement of a person is concerned, a value and a unit are recorded through the *BodyMeasurementValue* and *BodyMeasurementUnit*, respectively, and the semantic type of the body measurement, such as waist or hip measurement, through the *BodyMeasurementType* class.

Next in the conceptual flow of data within the HSB is the **Data Analysis Service**, a service responsible for extracting additional information based on the transformed semantic data. The Data Analysis Service takes advantage of a Rules Repository, filled with health and well-being-related rules. These rules have been extracted from recommendations of the American Heart Association [5] and other similar health-related organizations. By applying the semantic rules to the Quad Store data, the Data Analysis Service can infer whether the blood pressure of a user
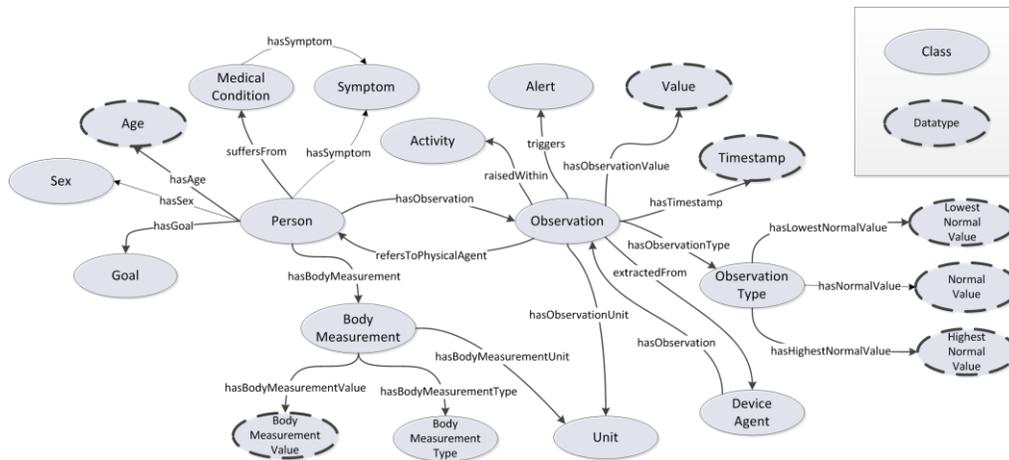
Figure 2. The Health and Lifelogging Data (HLD) Ontology

is low or high, if the amount of food and vegetables consumed within the day conforms to the ideal daily quantity based on the experts' recommendations, if the amount of time spent on an activity qualifies them as "active", etc. For instance, some experts' recommendations that led to the formulation of the HSB semantic rules are the following:

- The normal blood pressure values for an adult over 20 years old are 120/80 mm Hg (less than 120 systolic AND less than 80 diastolic). [6]
- A user should take up at least **30 minutes of moderate-intensity** aerobic activity at least **5 days per week for a total of at least 150 minutes of activity**. [7]
- A user should be involved in an activity for more than 10 minutes in order for this activity to be considered as exercise to boost their well-being. [8]

Overall, the Data Analysis Service enriches the Quad Store with information about the condition, actions and habits of the data owner and publishes this information to the HSB in the form of events via the Events Manager Service.

Events generated within the platform are sent to the queue of the **Events Manager Service**, which ultimately publishes them to the appropriate topic after verifying their validity. In particular, the Events Manager Service has to first parse the event and check whether the contained data do exist in the Quad Store, for example, in order to avoid references to wrong or obsolete object URIs. Then, the service has to make sure that such an event can originate from the sender of the event message based on internal security rules and logic. If all checks are successful, the event is forwarded to the appropriate topic, Apart from this, the event is stored in the Quad Store for future reference.

5. CLIENT FUNCTIONALITY OF THE PLATFORM

The functionality of the platform is exposed to third-party applications by means of several APIs. In this respect, three HSB services have been defined, namely the LOINC Data Warehouse Endpoint, the Client Service and the Query Service.

The **LOINC Data Warehouse Endpoint** provides wellbeing data in their original form. Its practical usefulness lies in the fact that third party APIs of different organizations (hospitals or medical institutions) many times need data in raw format in order to process them according to their own needs. In a similar way, professionals that are authorized to access specific user data may need to access the medical history with respect to a particular metric so that they can perform a more thorough diagnosis. The data are retrieved directly from the NoSQL database, which is ideal for allowing the retrieval of big chunks of data.

The **Client Service** provides client applications of the platform with updates. Notifications and updates regarding user activity are retrieved by the Client service. Events from the HWDE Topic of the Health Service Bus are constantly being pushed to any client application that uses the Client Service. Client Applications can range from specialized UIs for professionals that are authorized to track the progress of a specific platform user to dashboards that can be used from users to control their data, devices and profiles. In any case, the Client Service is meant to provide data that correspond to progress updates,

notifications (alerts) or combined knowledge that is extracted from the analysis of semantic data. Overall, it acts as the "connection point" between the HSB and any other service or application that is not JBossESB-based.

The HSB platform offers the capability of feeding external, custom-made applications with the LOINC data in their semantically enriched and reasoned-over format. Mobile application developers are able to send requests to the **Query Service** in the form of a SPARQL query, which is then forwarded to the queue of the Semantic Service. This way, developers are able to avoid the task of wearables data integration and homogenization and take advantage of the semantic data in the scope of building the functionality of their application.

6. CONCLUSION AND FUTURE WORK

The Enterprise Service Bus architectural paradigm provides an efficient way of integrating heterogeneous services and tools. In this paper, we examined the idea of adjusting a production-oriented ESB to the needs of healthcare. In HSB, homogenization is achieved by means of ontologies and access to integrated health data is given via services having well-defined functionality. The Health Service Bus can be exploited by a plethora of medical professionals as well as mobile application developers, facilitating this way tasks involved in the trending field of remote healthcare.

7. REFERENCES

[1] C. A. Marín, L. Mönch, L. Liu, N. Mehandjiev, G. V. Lioudakis, D. Kazanskaia, V. Chepegin, "Application of Intelligent Service Bus in a Ramp-up Production Context" in Proceedings of the Industrial Track of the Conference on Advanced Information Systems Engineering 2013 (CAiSE'13), CAiSE-IT 2013: International Conference on Advanced Information Systems Engineering (CAiSE) - Industrial Track; Valencia, Spain. CEUR Workshop Proceedings; 2013. p. 33-40.

[2] ARUM, Adaptive Production Management. [Online]. Available from: http://arum-project.eu/. Last accessed: 30/04/2015.

[3] IBM, Integrating healthcare services, Part 1: Using an Enterprise Service Bus for healthcare. [Online]. Available from: http://www.ibm.com/developerworks/library/j-hsb1/. Last accessed: 16/05/2015.

[4] A. Ryan and P. Eklund, "The Health Service Bus: An Architecture and Case Study in Achieving Interoperability in Healthcare". Studies in Health Technology and Informatics 160: pp. 922-926.

[5] American Heart Association. [Online]. Available from: http://www.heart.org/HEARTORG/. Last accessed 26/05/2015.

[6] American Heart Association, "Understanding blood pressure readings". [Online]. Available from: www.heart.org/HEARTORG/Conditions/HighBloodPressure/AboutHighBloodPressure/Understanding-Blood-Pressure-Readings_UCM_301764_Article.jsp. Last accessed 26/05/2015.

[7] American Heart Association, "American heart association recommendations for physical activity in adults". [Online].

Available from:
http://www.heart.org/HEARTORG/GettingHealthy/Physica
lActivity/FitnessBasics/American-Heart-Association-
Recommendations-for-Physical-Activity-in-
Adults_UCM_307976_Article.jsp. Last accessed
26/05/2015.

[8] Mobihealthnews, "Fitbit changes the way it tracks active
minutes". [Online]. Available from:
http://mobihealthnews.com/42241/fitbit-extends-minimum-
time-frame-for-active-minutes/. Last accessed 26/05/2015.