# A Software Architecture and Data Model for Community-Based Healthcare Environments

Antti Sarela, Niranjan Bidargaddi and Mohan Karunanithi

*CSIRO ICT Centre (E-Health Research Centre),*
*P O Box 10842, Level 20, 300 Adelaide Street, Brisbane 4000, QLD, Australia*
*FirstName.LastName@csiro.au*

*Abstract*— **Major changes in healthcare delivery are needed to ease the pressures caused by global increase in ageing population and prevalence of chronic diseases. Recent care initiatives address these problems by delivering care in community-based settings, away from hospitals. The community care model requires that a patient's health condition is also monitored in the home environment to allow assessment and follow-up of the health status.**

**Our work presented in this paper aims to design a system architecture and data model that would provide efficient means to perform essential information management for ambulatory monitoring in a home health care setting. We describe a general purpose architecture and data model that is designed for collecting ambulatory data from various existing devices and systems, as well as for analyzing, storing and presenting clinically significant information to the care personnel.**

## I. INTRODUCTION

There is a global need to introduce major changes in healthcare delivery to ease the pressures caused by increase in ageing population and prevalence of chronic diseases. For example, Cardiovascular Disease is one of the leading health problems in Australia, USA, and other developed countries. It still remains the leading cause of mortality in Australia, contributing to 18% of the total burden in disability-adjusted life years (DALYs) [1].

Recent care initiatives address these problems by delivering care in community-based settings away from hospitals, which requires changes both in work practices and health information systems [2]. The community care model requires that a patient's health condition is monitored in the home environment to allow assessment and follow-up of their health status. The challenge is to optimize the technologies both on the patient's and the care worker's side. The unobtrusiveness of home care devices is essential, especially for the patients suffering from chronic diseases. The care workers, on the other hand, want to receive only the relevant information on their patients for fast and reliable assessment.

There are many measurement devices and systems developed for home care and some have already been commercialized and sold in modest volumes. Despite many studies done on telehealth applications [3] showing promising results such as decrease in readmittance [4], the uptake of telehealth in practical care processes has been slow. The lack of interoperability is claimed to be one of the key factors [5]. The standardization to address this is proceeding but

the vast majority of commercial systems have proprietary communication interfaces and data formats, which makes the integration to other information systems difficult. They also record huge amounts of sensor data, that the care workers do not have the time to interpret [6]. By providing large amounts of data, the systems often fail to present optimized and care specific measures that would help the care workers to manage their patients efficiently. These issues can be addressed by designing optimal health information systems.

The information systems are the main tools for collecting, storing and analyzing the data and, most importantly, for providing essential information to the health care professionals. The design principles and architectures of such systems have been described for example in [7] and [8]. It is important to design standardized external interfaces to address the communication problem. In a recent work [5], the authors proposed an architecture that is based on standardized HL7 clinical document architecture (CDA) and Arden Syntax to develop an interoperable information system. In addition to the external interfaces, the internal data structure and architecture of a software system is essential to enable efficient and flexible data management. Our work aims to develop an architecture and internal data model that would be practical to the current non-standardized environment and yet be flexible to accommodate any new development or emerging standard to ensure the interoperability with other systems.

## II. METHODS

This paper presents a system architecture and internal data model design that provide efficient means to collect, analyze, store and present ambulatory time series data from existing home monitoring devices and systems. In the first part of the Section III, we describe the main characteristics of the architecture and the software structure. In the second part, we present the main features of ambulatory measurement data and describe our data model, that captures these features.

In addition, we are conducting a clinical trial within a Cardiac Rehabilitation program to develop and validate data processing algorithms for clinically significant measures and to test the developed architecture and data model with real patient data. In the trial, we are using a waist mounted ambulatory device to collect physiological data from the patient's home environment. The device measures both movement
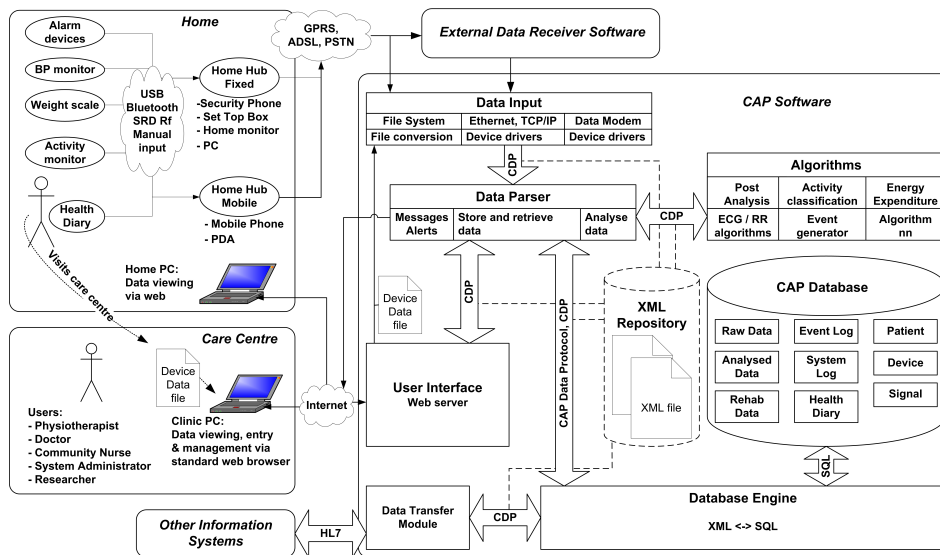
Fig. 1. The CAP software system architecture and external interfaces

activity and single lead ECG. The recorded data needs to be translated into clinically relevant measures which show the trends of patients condition and easily understood by health care professionals. An example of the modeling and presenting of real patient data is described at the end of Section III.

## III. RESULTS

### A. System Architecture

The Care Assessment Platform (CAP) Software system architecture follows a three-tier information system design model [8] and is presented in Figure 1. The architecture has the following main modules:

1) Interface Services Tier:
   - **Data Input** layer connects to external monitoring devices and software systems and converts data from a manufacturer specific format to internally used general XML structure (**CAP Data Protocol, CDP**). The converted data will be stored in a temporary **XML Repository**.
   - **User Interface** provides a web-browser based view on the measured data and other information as well as a means to manage and enter new data and patient information in the system. The device specific data files can be fed through the **User Interface** and will be passed to **Data Input** for conversion. All the other communication between **User Interface** and **Data Parser** is done by using the XML files.

2) Business Logics Tier:
   - **Data Parser** module acts as a relay station administering raw and analyzed data between **Data Input, Algorithms, User Interface**, and **Database Engines**.
   - **Algorithms** are activated by **Data Parser** and they read in the XML format data, calculate measures

from the data and return a new XML file containing the analysed results.

3) Data Services Tier:
   - **Database Engine** provides an interface between the **Data Parser** and the relational database. It reads and writes the internal XML files and transfers the data in and out the database by using SQL commands and queries.

The architecture provides a flexible platform that can receive data from any existing ambulatory measurement devices or systems, that have proprietary communication interfaces. A new device can be added to the system by writing a device driver module into the Input Layer that converts the proprietary data representation to the fixed internal XML format. New data types can be added by writing the required data definitions in the database. Pre- or post processing algorithms modules can also be flexibly included. The interface between the **Algorithms** and **Data Parser** is always using the internal XML format. Any algorithm simply reads the XML file passed by the **Data Parser** and writes the results to a new file. When the data is permanently stored in the database or not otherwise needed, the temporary XML files are deleted from the repository.

### B. Data Model

The main novelty of the architecture is the internal communication structure and data model, which uses an optimized XML format and a XML Repository to transfer all the data between the main modules and database. This enables high modularity, flexibility, low coupling, and high cohesion in the software. The same data structure is used in all the communication between the software modules and database to transfer measurement data and other information.

The measurement data that needs to be presented in XML format and stored in a database has the following characteristic features.
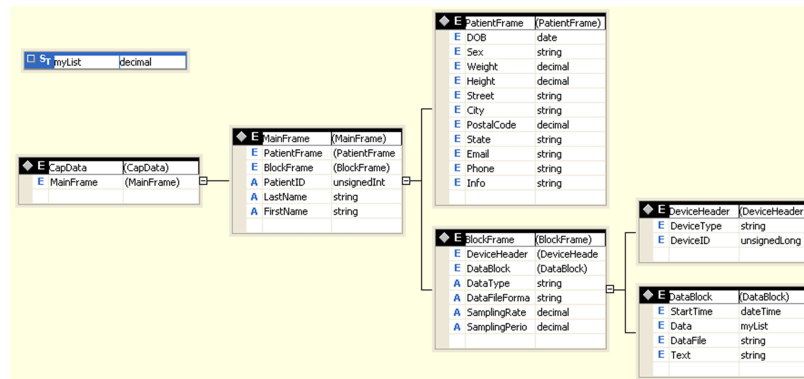
Fig. 2.   XML structure for storing the measurement and patient data

1) The data has a wide range of sampling rates, which can vary from more than 1kHz for ECG signal to one sample a day or less (for example body mass) depending on the measured physiological parameter. The analyzed data has often very low sample rates because the clinically interesting feature might be the monthly or weekly average of a measured signal.
2) The data does not necessarily have a defined sampling rate such as heart rate interval, which may be coded as beat-to-beat timing values.
3) The collected data is often highly discontinuous. The patient might use the activity monitor continuously for few days but have several days between the measurement sessions.
4) The devices might produce very large amounts of raw data or only few values occasionally.
5) The data might describe an event such as a fall. An event is defined with an event code and a timestamp and can be associated with other explanatory data such as event duration.
6) A diary entry which includes textual data, a timestamp and may also have numerical data is a special case of an event and can be described with a similar structure.

The XML format used to capture the data is presented in Figure 2. All the patient data is stored within a **MainFrame** structure. The **MainFrame** attributes always define the patient id and name, which is a minimum amount of patient identification associated with the patient data in one XML document. All the other elements in the document are optional. For example a XML document containing only **MainFrames** with patient names and IDs without any other elements can be used to communicate the list of patients in the database to the **User Interface**. The **PatientFrame** element in the **MainFrame** contains detailed patient demographics. The model can be expanded to include other types of information by appending elements parallel to **PatientFrame** and **BlockFrame**.

The **BlockFrame** and **DataBlock** elements are the key structures that hold the different types of time series and event data described above. A **BlockFrame** contains always at least one **DataBlock** and optionally a **DeviceHeader**.

The **DataBlock** carries the start time (**StartTime**) of the continuous sequence of data samples that are stored in the **Data** element. The numerical data is stored directly in the XML document if the amount of samples in one **DataBlock** is small. If there are large number of samples, for example in case of raw ECG data, it is more efficient to store the samples in a binary file and reference the file in the **DataFile** element. The essential information to read the data is stored in the **BlockFrame** attributes: **DataType**, **DataFileFormat**, **SamplingRate** for fast signals, and **SamplingPeriod** for slow signals. The **DataType** attribute is used to identify the data in the **BlockFrame** and to determine the required data operations. If the sampling rate is not defined, the **Data** element of this data type either includes the timing information such as beat-to-beat interval relative to the **StartTime** or event data, which only has the absolute time stamp defined in the **StartTime**. The event codes or diary entries are stored in the **Text** element and the possible associated data in the **Data** element.

Figure 3 illustrates how patient data is stored in the XML document. Every continuous sequence of data forms one **DataBlock** element. The **StartTime** and **SamplingRate** elements fully define the timing of each **DataBlock**.

The created XML document is a temporary data storage. The data is eventually written to a relational database, which has a similar model to store all the data. The Entity Relationship Diagram (ERD) of the database section that stores the data is presented in Figure 4. The structure is very similar to the XML format, which makes it straightforward for the **Database Engine** to convert data to and from the database. Each **DataBlock** in the XML document is written to a table defined in **DataType**. The **DataBlockHeader** holds the timing information of each data block and relates the data to the information in the **Patient**, **Signal**, and optionally the **Device** tables.

The described structure that is based on **BlockFrames** and **DataBlocks** can capture the characteristics of all the various signal types and features that may be collected in a ambulatory setting. Because the mapping between different data types and related algorithms is also stored in a database table, it is straightforward to append completely new physiological time series signals to the model.
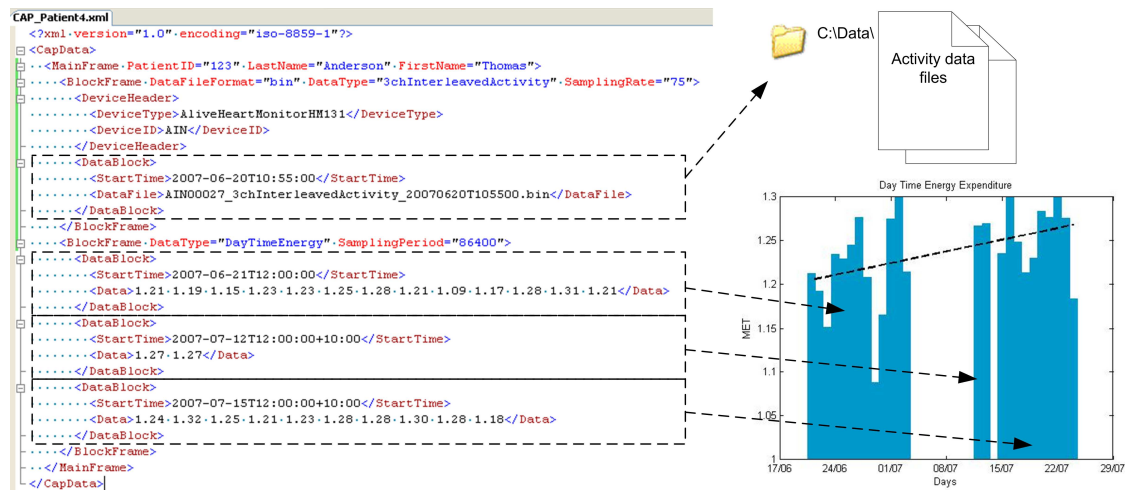
Fig. 3. Sample XML file that stores patient's activity information and average day time energy expenditure values calculated by an energy expenditure algorithm. The framed DataBlocks illustrate how discontinuous data is stored
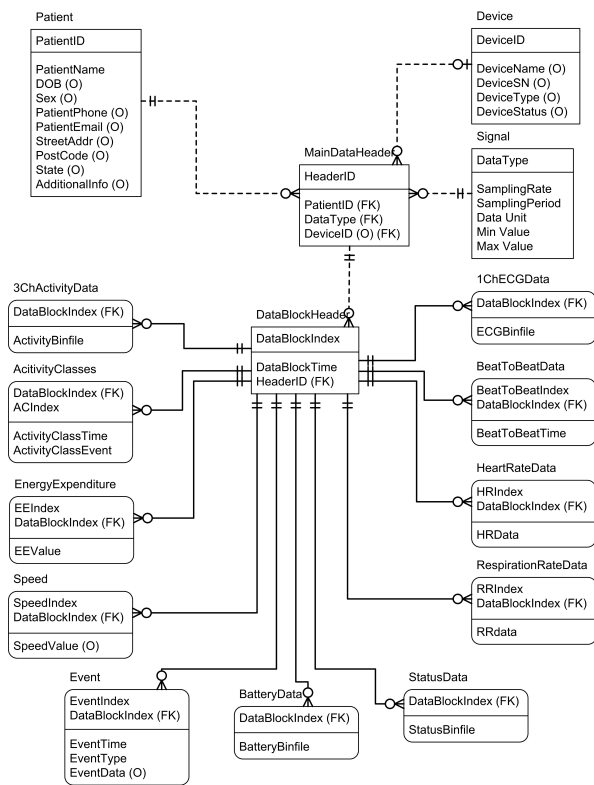


Fig. 4. The database ERD model

## IV. CONCLUSION

We developed a novel software architecture for community care setting. It uses an XML-based data model and communication structure and a relational database to efficiently handle typical features of physiological measurement data and analysis results. The architecture and model is flexible to accommodate variety of ambulatory devices, handle data and utilize a number of data processing algorithms.

Our future work will focus on building a web server based user interface that will present the processed and analyzed clinically significant information to the care workers. We are planning to evaluate the complete prototype system in a community-based cardiac rehabilitation setting. We will also continue to integrate new devices and data processing algorithms to the system.

## V. ACKNOWLEDGMENT

## REFERENCES

[1] Theo Vos and Stephen Begg, "The burden of cardiovascular disease in australia for the year 2003", Report prepared for the National Heart Foundation of Australia, July 2007.

[2] Reinhold Haux, "Health information systems - past, present, future.", *Int J Med Inform*, vol. 75, no. 3-4, pp. 268–281, 2006.

[3] Guy Par, Mirou Jaana, and Claude Sicotte, "Systematic review of home telemonitoring for chronic diseases: the evidence base.", *J Am Med Inform Assoc*, vol. 14, no. 3, pp. 269–277, 2007.

[4] P. deToledo, S. Jimeanez, F. delPozo, J. Roca, A. Alonso, and C. Hernandez, "Telemedicine experience for chronic care in COPD", *IEEE Transactions on Information Technology in Biomedicine*, vol. 10, no. 3, pp. 567–573, July 2006.

[5] Oliver-J. Bott Mirko Geisler Maik Plischke Wolfram Ludwig Andreas Hornberger Reinhold Haux Michael Marschollek, Klaus-H. Wolf, "Sustainable ubiquitous home health care architectural considerations and first practical experiences", in *MEDINFO 2007*, 2007.

[6] Labb F-Gagnon MP Lamothe L, Fortin JP and Messikh D, "Impacts of telehomecare on patients, providers, and organizations.", *Telemed J E Health.*, vol. Jun, pp. 12(3):363–9, 2006.

[7] Wolf KH Bott OJ, Marschollek M and Haux R., "Towards new scopes: sensor-enhanced regional health information systems - part 1: architectural challenges.", *Methods Inf Med.*, vol. 46(4), pp. 476–83, 2007.

[8] Stephen Chu and Branko Cesnik, "A three-tier clinical information systems design model", *International Journal of Medical Informatics*, vol. 57, pp. 91107, 2000.