Data Management System: A Context Aware Architecture For Pervasive Patient Monitoring

John O'DONOGHUE and John HERBERT
Department of Computer Science
University College Cork, Ireland

Abstract. Good data management infrastructures within a medical environment help improve productivity levels for medical practitioners. It also has the potential for enhanced accuracy in patient diagnosis. An essential characteristic of a ubiquitous medical environment is timely delivery of accurate data. This trait is not always found in technology rich environments. The DMS (Data Management System) architecture aims to optimise accurate data delivery and is designed to work within a WSN (Wireless Sensor Network) medical environment. An overview is presented outlining the advantages of realising a DMS using agents within a WSN. Agents can effectively encapsulate, extract and interpret real-world context aware information ensuring physicians get the 'correct' data on time every time. Details are given in relation to the hardware and software implementation of the DMS prototype under development.

Keywords. Software agents, patient monitoring, wireless sensor networks.

1. Introduction

This paper presents an architecture called DMS (Data Management System) which provides a context aware service within a WSN medical environment. Timely acquisition of relevant information is of great importance. DMS collects real-time patient data using a software agent platform on top of a WSN. The advantage of agent technologies stem from their ability to be proactive, capacity to adapt (e.g. to deal with intermittent network connections) and their efficient bandwidth usage. These characteristics are ideal for developing architectural solutions for context and situation aware applications. The software agents work in three logical layers: 1) data collection; 2) data correlation and 3) data presentation (refer to figure 1). The agents within each layer operate according to the DMS rules and context aware parameters. This ensures that greater attention is given to higher priority data sets, and that critical data is given the appropriate amount of system resources.

2. Research Motivation

Research has been conducted in a number of related areas including contextual retrieval of medical data [1] and agent based medical environments [2], both of which highlight the role that agents can play in a medical environment. Analysis into the cause of errors within a medical environment over the period 1998-2005 was presented at the EAN [3]. It was found that due to the lack of sufficient technological aids basic administration tasks were jeopardised. Standard procedures such as identifying patient prescriptions were found to be incorrectly managed resulting in an inappropriate drug or dosage being administered.

The level of intercommunication between medical staff, other members of staff, and patients is high. Without appropriate tools, ill-informed decisions are made resulting in poor medical attention. One solution to alleviate some of these issues is the collaboration between medical staff and software agents [4]. With the integration of software agents, medical practitioners have at their disposal a context and situation aware support tool to assist in making well informed decisions.

Research into real-world context aware medical technologies was conducted using medical eGadgets [5]. By introducing medical eGadgets a patient's environment becomes partly context aware and physicians can view a limited number of patient vitals on request within a wireless network. The DMS will improve on this approach by integrating agents into the environment. It places greater emphases on mobility, location and the explicit meaning of the data within the medical environment. This approach will enhance the medical environment's context awareness capabilities. In developing the DMS, other issues need to be taken into account including mobile computing and database constraints [6]. The manner in which patient records are stored determines the level of efficiency for the DMS's logging and retrieving of real-time patient data.

The integration of software agents conforms to the idea expressed in [7] "that mobile agents capable of discovering, extracting, interpreting and validating context will make significant contribution to increasing efficiency, flexibility and feasibility of pervasive computing systems". This paradigm can be very effective within a pervasive medical environment. The attributes of software agents make them ideal for handling external stimuli such as patient sensors. As data within medical environments carry high levels of responsibility DMS has built in QoS agents to monitor system resources. This ensures that data is processed as long as there are sufficient system resources to execute the task [8].

3. DMS Architecture

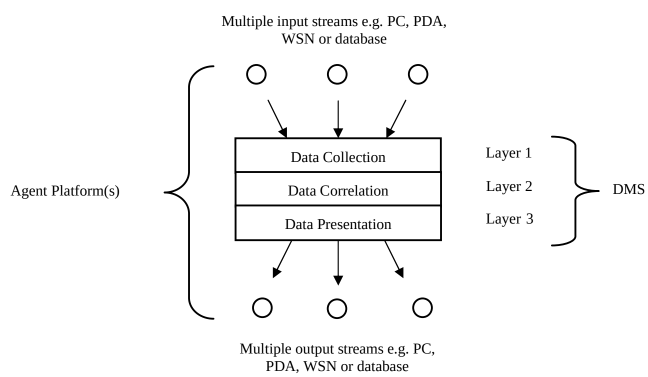

Figure 1. A Logical Overview of the DMS Architecture.

As ubiquitous tools function in data rich environments, DMS is designed to contextually respond to the end users run time needs. DMS is comprised of three layers each of which execute on top of an agent platform. Their primary task is to handle and present data in the required format while ensuring that all context and situation aware data are taken into account. DMS consumes data from a number of input streams (e.g. PDA, patient module) it correlates this data to ensure that explicit relationships are reviewed for further analysis. This approach is vital in a context aware environment, as the DMS can inform the end user if a future event breaks one of the system rules. The core DMS agents operate in the following areas:

- Data Collection [Layer 1]

DMS may collect data from wireless patient modules and medical staff PDAs. Such data is transmitted to the DMS layer (refer to figure 2). The frequency of data transmission from the patient module to the DMS layer is determined by the practitioner's requirement for that particular data set.

- Data Correlation [Layer 2]

Multiple data records relating to the same real-world element may exist within the DMS. To ensure that vital information is not ignored, an explicit relationship between these data sets is defined giving the physician a complete and accurate overview of the patient's current status.

- Data Presentation [Layer 3]

Finally data presentation is determined by the context of the mobile environment in association with patient and medical staff profiles. This will ensure that medical staff are not over loaded with irrelevant information.

4. Experimental Prototype

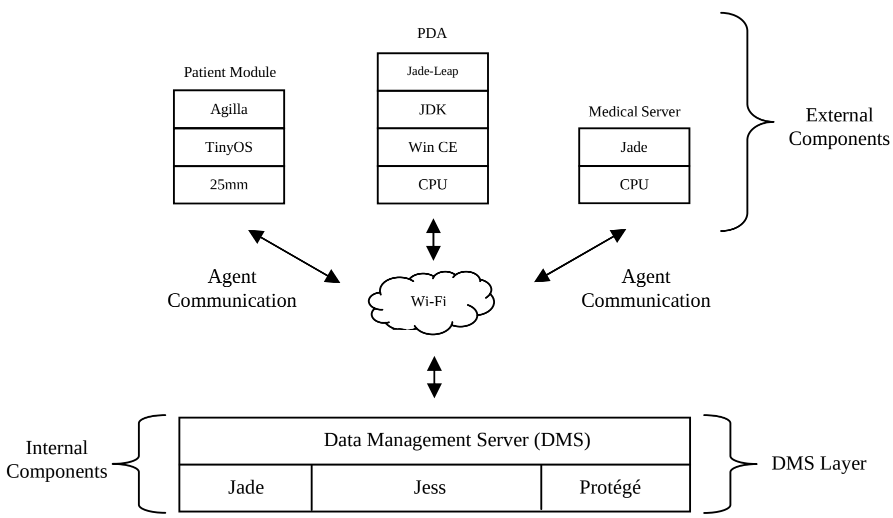

Figure 2: A Logical Overview of the DMS Implementation.

The DMS architecture is built on a number of components (refer to figure 2). Decision making processes are made within the DMS layer. The DMS layer comprises of three main components and these are Jade (agent platform), Jess (expert decision system) and protégé (explicit medical ontology).

Real-time patient readings are received from the wireless patient module and transmitted back to the DMS layer. Requests from practitioners may come from their PDAs. Static data, such as patient records, are stored in a medical database. Data communication is facilitated through agents over a Wi-Fi network. The three main external components are as follows:

- A Tyndall 25mm cube (Patient module) [9] which may be attached to a patient. Patient vital readings are returned to the DMS where automatic analysis may begin based on the patient's current and previous states. Generated results will be forward on to the physician's PDAs and/or PC.
- A PDA, enabling medical staff to monitor their patient's state.
- Finally a medical server which stores the archived data of the patients. This static information pool can play an important role when analysing a patient's current state. Relationships defined in the ontological model of the DMS layer between static (medical database) and dynamic (live sensor readings) patient records may indicate possible solutions to assist the medical staff during diagnosis.

4.1 DMS Scenarios in a Medical Environment

The DMS is built on the explicit relationship between patient data and the current context of the external environment. Presented are two scenarios highlighting DMS capabilities.

Scenario 1 [Timing and Mobility]
Andrew a new patient at his local hospital had been complaining of small chest pains. After initial tests doctors could not find anything in particular but agreed to keep him under observation for the next few days. Wireless monitoring sensors were attached to Andrew. This enabled him to walk around the hospital grounds while doctors were able to monitor his condition on a constant basis. A few hours later the wireless sensors indicated that Andrew's heart rate had reached a serious level and that he needed instant attention. The DMS was configured to react as follows:

- Instantly notify relevant practitioners of Andrew's state and location.
- Selected staff members PDAs were instantly uploaded with Andrew's real-time and archived data.

Scenario 2 [Informed Decision]
Michelle the first resident doctor to attend Andrew could see that Andrew was just after suffering a heart attack. After examining Andrews's condition she decided to administer a thrombolytic drug. Before carrying out the procedure the DMS reported that Andrew was allergic to such drugs and identified another course of action.

With DMS's timely delivery of accurate data built on an explicit relationship between real-time data (Andrew's vital signs) and archived data (structured medical database) the number of medical errors may be reduced.

4.2 DMS Components

Communication between the DMS layer and its external components is achieved by deploying a software agent platform to handle the dynamic real-world requirements. The three layers of the DMS architecture (refer to figure 1) are built as follows:

- Data Collection
  Sensors: Patient module. Operating System (OS), TinyOS [10]. Agent platform, Agilla [12]. Sensor agents combine the advantages of intelligent agents and smart sensors. The current DMS prototype operates with ECG, Blood pressure and Pulse rate sensors.

  PDA: Compaq Pocket PC. OS, WinCE. Agent platform, Jade-Leap. Combines the capabilities of a PC based operating system with an agent technology.
- Data Correlation
  Jade: A FIPA compliant software agent architecture [11]. Dynamically manages and organises incoming and outgoing medical data in a context and situation aware manner.

  Protégé: helps create an ontology which defines the data and the explicit relationships between the different data, static and dynamic, as used in the DMS. This provides the basis for the dynamic diagnosis in the system as illustrated in scenario 2.

  Jess: A rule based expert system [13]. Enables developers to define a set of context and situation aware rules based on the DMS's ontology.
- Data Presentation
  Data filtering: depending on the current context aware variables, only relevant data is sent to the end user.

4.3 Patient Modules and Agent Technologies

A DMS patient module consists of two logical elements 1) Tyndall 25mm cube, 2) patient sensor(s) (refer to figure 3). A Tyndall module may contain the following sub modules 1) FPGA 2) Sensor (with RS232 Serial Port Interface) 3) Transceiver (with Atmel microcontroller ATmega128L) and 4) Coin cell (power).

Wireless sensor applications are limited in that they must be downloaded onto the sensor node prior to deployment. During run-time only small modifications can be made to the application, severely limiting the patient module's capabilities. With the emergence of mobile agents within sensor network environments [12], context aware attributes can now be associated with a patient module.

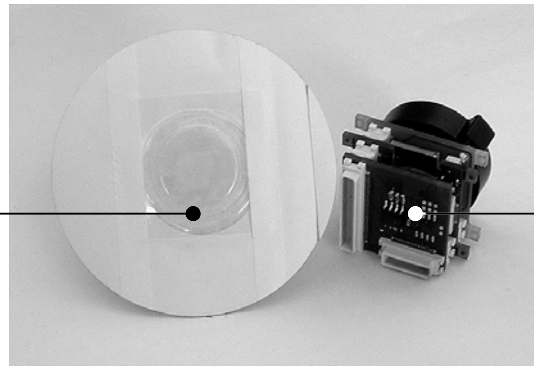

Figure 3: ECG (electrochardogram) Patch with Patient Module.

Multiple patient monitoring sensors may be attached to the Tyndall 25mm module at any given time. Illustrated in figure 3 is one ECG patch alongside a patient module. Analysis of patient vital signs can be processed locally. When a DMS rule is triggered (refer to scenario 2) communication from the patient module to the DMS is stimulated.

5. The Role of Ontologies in Pervasive Medical Environments

The emphasis that the DMS places on 'correct' data leads immediately to the need for careful attention to the meaning of data. Data coming from a variety of sources, including various sensors, PDAs and medical servers must be merged and correlated based on well-defined semantics. In addition, one must consider that information will be input to the DMS from other medical information systems and that information generated within the DMS will be output to other systems. Thus we require that the semantics of data is well defined both internally within the DMS and externally for import/export of information.

The most complete and well-defined semantics of data is provided by medical ontologies. Two of the most comprehensive medical ontologies are HL7-RIM (Health Level 7 Reference Information Model) [14] and UMLS (Unified Medical Language System) [15]. HL7-RIM comes from HL7 which is an ANSI-accredited standards development organisation dealing with "standards for the exchange, management and integration of data that support clinical patient care and the management, delivery and evaluation of healthcare services" [14]. The UMLS is based on 60 medical thesauri and classifications and has the UMLS semantic network that provides a medical ontology containing 134 semantic types linked by 54 types of semantic connection. Such comprehensive medical ontologies become very large - the UMLS is built from a corpus of almost 500,000 medical concepts.

The approach taken in the DMS is to base the internal semantic data on a subset of the medical ontology that is relevant to the targeted area or cardiac/blood applications, thus avoiding the overhead of the full medical ontology. While the initial implementation uses an efficient internal representation of appropriate information, this will be replaced by a small ontology built with the Protégé toolkit [16]. Protégé is a Java-based ontology development environment and so fits well with the other Java-based parts of the implementation. For external import/export of information a medical ontology server is used. The demands of compatibility with external information servers mean that this will be based on a standard such as HL7-RIM. A good initial candidate might be the (incomplete) version of HL7-RIM implemented as a Protégé ontology [17]

There are also a number of efforts at standardising medical information to support interchange of data. Health record standards is one of the concerns of ISO TC 215, a technical committee of the international standards organisation, ISO [18]. CEN TC251 is part of CEN [19] (the European Standards body) dealing with medical informatics. It has defined 10 full standards and over 40 pre-standards. OpenClinical [20] is an international organisation promoting methods and information technologies capable of supporting knowledge management in healthcare. Compatibility with such international efforts will also be a priority for the final implementation.

6. Summary and Future Research

This paper presents the DMS architecture. The usage of WSNs in a medical environment adds a new dimension to patient data management. An essential characteristic of a ubiquitous medical environment is timely delivery of accurate data. By integrating an agent platform to function alongside an explicit data model, a comprehensive analysis of archived and real-time patient readings is given, giving physicians a valuable tool to assess a patient's current condition. Semantic and ontology tools can assist greatly in developing explicit data models. Compatibility with international clinical records standards will also be a priority for the final implementation. DMS is currently built to work with ECG, Blood pressure and Pulse rate sensors. These three sensors will be utilised to monitor a patient's cardiovascular state within a medical environment. Subsequent DMS versions will be developed to monitor a patient's cardiovascular state within the comfort of their own home.

References

[1] Marcela Rodriguez and Alfredo Preciado, "An Agent Based System for the Contextual Retrieval of Medical Information", AWIC 2004. pp 64-73.

[2] Marcela Rodriguez, Jesus Favela, Victor M. Gonzalez and Miguel A. Munoz. "Agent Based Mobile Collaboration and Information Access in a Healthcare Environment" in Proceedings of Workshop of E-Health: Applications of Computing Science in Medicine and Health Care 2003.

[3] "EAN Medical Conference" http://www.ean.ie/health.shtml?sid=3&ssid=7

[4] James Allen, Nate Blaylock and George Ferguson "A Problem Solving Model for Collaborative Agents", Proceedings of the first international joint conference on Autonomous agents and multiagent systems, 2002, pp 774 – 781.

[5] Stephen Bellis, David Murphy et al, "Medical eGadgets" in Ubicomp 2004.

[6] Daniel Barbara, "Mobile Computing and Databases – A Survey", in IEEE Transactions on Knowledge and Data Engineering, Vol. 11, pp 108-117. 1999.

[7] Arkady Zaslavsky, "Mobile Agents: Can they Assist with Context Awareness?", in MDM. 2004, pp 304

[8] Luis Moura Silva, Victor Batista and Joao Gabriel Silva, "Fault-Tolerence Execution and Mobile Agents", in International Conference on Dependable Systems and Networks. 2000, pp 135-143

[9] Brendan O'Flynn, S. Bellis, et al, "A 3-D Miniaturised Programmable Transceiver", XXVIII International Conference of IMAPS Poland Chapter, 2004.

[10] Philip Levis, Sam Madden, et al, "TinyOS: An Operating System for Wireless Sensor Networks", Aarts (Eds.), Ambient Intelligence, Springer-Verlag, 2004.

[11] Fabio Bellifemine, Agostino Poggi and Giovanni Rimassa "JADE: a FIPA2000 compliant agent development environment", Proceedings of the fifth international conference on Autonomous agents 2001, pp 216-217.

[12] Fok, C.-L., Roman, G.-C. and Lu, C., "Rapid Development and Flexible Deployment of Adaptive Wireless Sensor Network Applications" Technical Report WUCSE-04-59 Washington University, Department of Computer Science and Engineering, St. Louis.

[13] Jess, the Java Expert System Shell: http://herzberg.ca.sandia.gov/jess/

[14] Jenders RA, Sujansky W, Broverman C, Chadwick M. "Towards Improved Knowledge Sharing: Assessment of the HL7 Reference Information Model to Support Medical Logic Module Queries", Proc AMIA Annu Fall Symp, 1997, pp 308-312.

[15] UMLS, the Unified Medical Language System: http://www.nlm.nih.gov/research/umls/

[16] N. F. Noy, R. W. Fergerson, and M. A. Musen. The knowledge model of Protege2000: Combining interoperability and flexibility. Proceedings of the 2th International Conference on Knowledge Engineering and Knowledge Management (EKAW'2000), Juan-les-Pins, France, 2000.

[17] Bhavna Orgun, "interoperability in heterogeneous medical information systems using smart mobile agents and HL7 (EMAGS)" M.Sc. thesis, Dpartment of Cmmunication Siences, Macquarie University, Australia. 2003.

[18] ISO, http://www.iso.org

[19] CEN, http://www.cenorm.be

[20] OpenClinical, http://www.openclinical.org/