Profile Based Sensor Data Acquisition in a Ubiquitous Medical Environment

John O'Donoghue, John Herbert
Department of Computer Science
University College Cork
{j.odonoghue, j.herbert}@cs.ucc.ie

Abstract

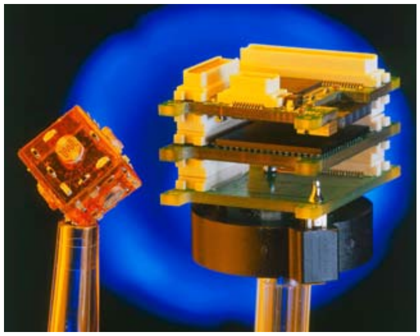
The timely delivery of relevant, accurate data is essential within a ubiquitous medical environment. At patient point-of-care, relevant sensor real-time vital sign readings may be required for diagnosis. These vital sign data sets may be supplied through a wireless sensor network (WSN) comprised of wireless patient sensor nodes. The Data Management System User Profile (DMS-UP) is designed to effectively support data distribution techniques within a WSN. A practitioner's DMS-UP determines the type and form of real-time patient readings to be uploaded onto their mobile device at the point-of-care. Details are given on the DMS-UP prototype and how it is integrated into the existing DMS (Data Management System) architecture. The current implementation includes sensors that take readings of the electrocardiogram, blood pressure and pulse rate.

1. Introduction

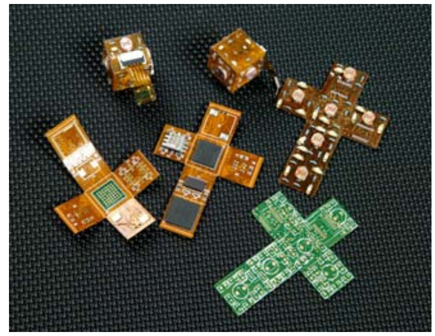
The Data Management System (DMS) provides a context aware data Quality of Service (QoS) within a WSN medical environment [1]. This paper presents a Data Management System User Profile (DMS-UP) which is integrated into the DMS architecture. DMS in conjunction with DMS-UP has the potential to enhance the accuracy of patient diagnosis. The DMS-UP facilitates practitioners in receiving relevant real-time patient readings at the point-of-care. Each patient module generates large amounts of data; data being considered at present include blood pressure, pulse rate, and electrocardiogram readings. At the point-of-care medical staff may wish to view just a sample or a summary of the patient vital sign readings. To ensure that appropriate information is delivered to the practitioner's mobile device, each member of the medical staff is assigned a user profile. User profiles have been shown to play a significant role in ubiquitous environments [2], [3]. [2] uses the term data recharging for "a service that aims to recharge device-resident data in an analogous manner to the way a battery's power may be recharged". Data recharging updates a user's mobile device with context relevant information. The DMS-UP applies a similar approach while also emphasising that "data coming from a variety of sources, including various sensors, PDAs and medical servers must be merged and correlated based on well-defined semantics" [1]. By combining a user's profile [4] with a well-defined semantic structure, relevant context-based information may be delivered to the end-user's mobile device. This has the added effect of reducing information overload and bandwidth usage. More importantly it provides medical assistance during diagnosis.

The DMS architecture aims to increase the accuracy of patient data delivery and is designed to work within a WSN medical environment. A WSN is built using Tyndall25 [5] nodes. The Tyndall25 hardware platform is a 25mm x 25mm stackable layer structure that provides communication, processing, sensing and power supply (cf. figure 1). The communication layer is comprised of a microcontroller, RF transceiver and embedded antenna. Various application specific sensors as well as a generic sensor interface/communications layer have been developed to communicate with the DMS Architecture. The next generation of the Tyndall module is the Tyndall10 (10mm x 10mm) [6]. The Tyndall10 is built on a flexible circuitry structure, and should result in little or no disturbance to a patient when attached.

The patient module samples a patient's vital signs. These real-time readings are then transmitted back to the DMS server. Requests to the DMS for information may come from practitioners, for example from their PDA or high-spec mobile phone during their daily rounds. The DMS processes live sensor readings and analyses them with respect to the existing information. Relationships are defined between static (e.g. patient record, medical database) and dynamic (live sensor readings) patient data sets. These relationships help to



Tyndall 10mm and Tyndall 25mm modules



Tyndall 10mm module in various forms

Figure 1. Tyndall 25mm and 10mm modules.

identify which information is relevant or important and may indicate possible directions during diagnosis to the medical staff. The current DMS implementation operates with the Protégé ontology tool [7]. Generated results may then be forwarded to the practitioner's display device, such as a PDA.

DMS itself is built on top of a Jade agent platform [8]. JADE (Java Agent Development framework) is a software framework that supports the development of agent applications [9]. Software agents can successfully implement solutions that involve distributed computation or communication between components.

2. Profile based sensor data acquisition

When uploading patient information to a mobile device two techniques may be employed:

1. Push based (e.g. upload patient data records onto a practitioner's mobile device before their scheduled appointment [10])
2. Event-response based (e.g. upload real-time patient readings when requested at the point-of-care).

The DMS architecture supports both techniques. As real-time patient sensor readings need to be transmitted at the point-of-care DMS-UP employs an event-response based approach. The real-time patient readings may be managed in two ways 'transmitted and stored' or 'stored'. The rate at which a patient module samples a reading is set according to the current context. The current DMS-UP prototype deals with two main contextual elements: practitioner profile; and environmental context (e.g. time, location).

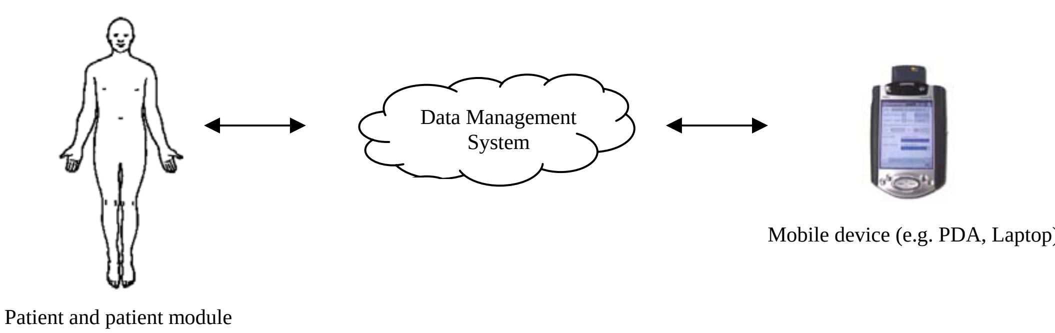


Figure 2. A logical overview of the interaction between a patient (wireless patient module) and practitioner (wireless PDA or laptop).

A user profile may contain many variables. Two types of variables exist:

1. Static (e.g. name, gender, date of birth, job title)
2. Dynamic (e.g. current location, patient list etc).

Dynamic context variables play an important role in deciding which data set a mobile user receives. Thus, it ensures that medical staff will receive a specific patient data set if that patient is on their profile (i.e. patient list).

An event-response based strategy in uploading real-time patient readings to a handheld device is built upon a rule based expert system. It is designed to function alongside the decision-making processes (i.e. software agents) within the DMS server. Jess, a rule based expert system and scripting language, is employed within the DMS architecture [11]. Jess can interpret and evaluate the contextual elements of a user's profile to recommend data management operations. The contextual elements required to enable efficient data download include the location of the handheld device, the state of the patient and the profile of the practitioner. A logical overview of the interaction between a practitioner's mobile device and a patient at the point-of-care is presented in figure 2.

Patient medical data carries a higher degree of significance compared to data from non-critical environments (e.g. supermarkets, parking facilities). Data generated by the wireless patient module (and other peripheral input devices) is sent to the DMS server for context-based analysis before it is passed on to a member or group of the medical staff. This ensures that patient data readings are analysed correctly, and may even inform practitioners of possible outcomes. Dynamic variables on a user's profile constantly change. To save on communication overhead DMS-UP's practitioner profiles are located within the DMS server.

A goal of the DMS is to ensure that patient data have significance beyond just numbers. Thus all sensor readings have meaning and are constantly compared against an ontological model for further analysis, thus improving the diagnosis process.

3. DMS-UP architecture

Real-time patient readings are received from the wireless patient module and transmitted back to the DMS server (cf. figure 3). Requests from practitioners may come via their mobile device (e.g. PDA, laptop). Static data, such as patient records, are stored in a

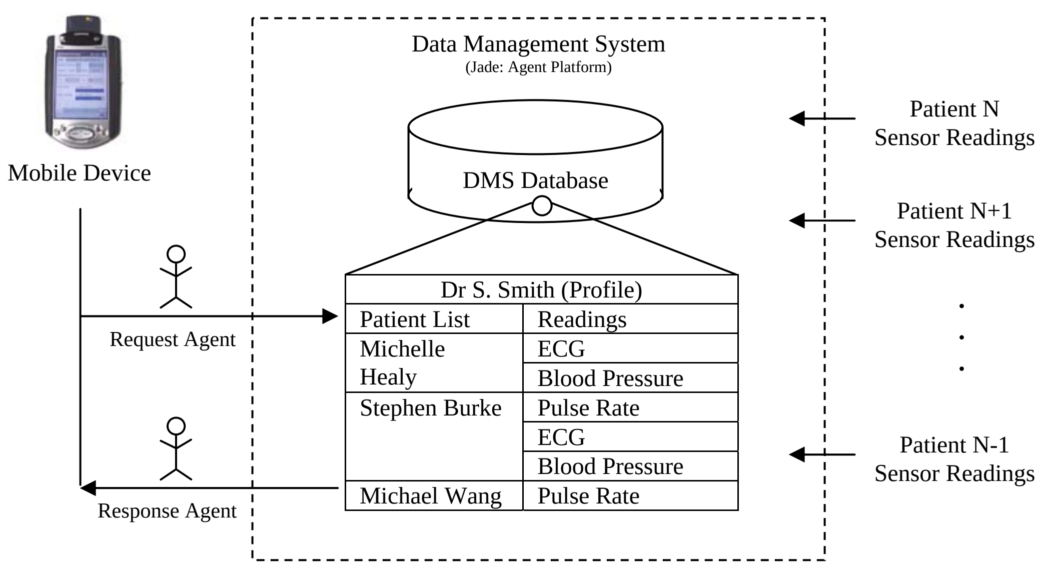


Figure 3. Mobile device interaction with a DMS server through an agent platform.

medical database. Data communication is facilitated through agents over a Wi-Fi network. Interaction between the DMS through DMS-UP and a medical practitioner is outlined.

Scenario: A Medical Consultant requests real-time vital sign readings and a summary of a patient at the point of care.

Stage 1 (cf. figure 4)
A consultant attending to a patient within a hospital ward requests information on the patient's vital signs.

Stage 2
A 'request agent' is created on the consultant's PDA. This communicates the new task to the DMS server.

Stage 3
A DMS 'data management agent' receives the consultant's 'request agent'. The 'data management agent' examines the consultant's profile and reacts to the 'request agent'. As the consultant in question is a cardiovascular expert the 'data management agent' identifies that electrocardiogram (ECG), blood pressure and pulse rate readings are required in the summary report.

Stage 4
A 'data collection agent' scans the DMS database for the patient's ECG, blood pressure and pulse rate readings. It also makes a request for a real-time reading to be taken.

Stage 5
Some specific real-time sensor readings are taken if required.

Stage 6
The 'data management agent' now contains a summary of the patient's cardiovascular state with the latest sensor readings.

Stage 7
The 'data management agent' communicates this information onto the 'response agent'.

Stage 8
The 'response agent' transfers the patient sensor readings onto the mobile device where it will assist the medical consultant in making a well informed diagnosis.

Implementation Details
DMS is built on a Jade agent platform. Jade, a FIPA compliant agent framework, executes on the DMS server while Jade-LEAP executes on the mobile device (i.e. PDA). This provides an effective client-server platform. A predefined collection of Jess rules are built within the DMS server. Each Jess rule works alongside the DMS Protégé model, which is based on data relationships between patient readings and the simple DMS cardiovascular ontology model.

4. Conclusions and future work

This paper presents the DMS User Profile (DMS-UP). An essential characteristic of a ubiquitous medical environment is timely delivery of accurate, relevant data. A network of wireless patient sensors within a medical environment may generate large amounts of data, resulting in data overload. The introduction of user profiles adds a new dimension to

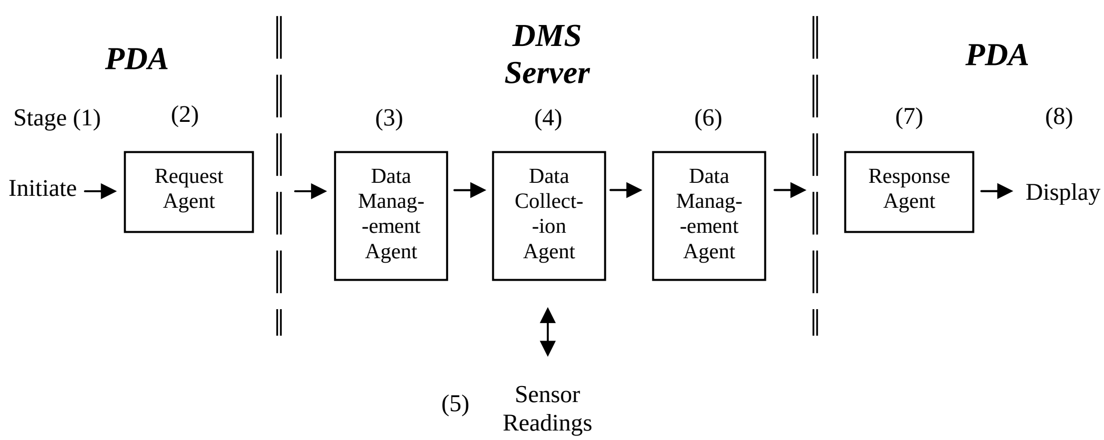


Figure 4. Sequence of events between a practitioner's PDA and the DMS server.

patient data management within a wireless sensor network. These are used to effectively manage the delivery of relevant information to a practitioner's mobile device. Architectures such as DMS require an established middleware framework to effectively manage its patient data sets. An agent based middleware is used in DMS-UP to manage data in an efficient, flexible manner. By making use of patient situation and practitioner profiles in conjunction with an explicit data model, a comprehensive analysis of archived and real-time patient readings may be provided, thus giving practitioners a valuable tool to assist in patient diagnosis. DMS is currently built to work with ECG, blood pressure and pulse rate sensors.

Acknowledgment

This work is funded by the Boole Centre for Research in Informatics and is supported by the Tyndall National Institute through the SFI-funded National Access Programme (NAP).

References

[1] J. O'Donoghue, J. Herbert, "Data Management System: A Context Aware Architecture For Pervasive Patient Monitoring" in Proceedings of the 3rd International Conference on Smart Homes and Health Telematics (ICOST 2005), pp. 159-166.

[2] M. Cherniack, M. J. Franklin, and S. Zdonik, "Expressing user profiles for data recharging", IEEE Personal Communications: Special Issue on Pervasive Computing, 2001, pp. 32--38.

[3] D. Poo, B. Chng, and J. Goh, "A Hybrid Approach for User Profiling", in proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS'03), pp. 103.

[4] A. Sinner, T. Kleemann, and A. Hessling, "Semantic User Profiles and their Applications in a Mobile Environment", In Artificial Intelligence in Mobile Systems 2004 (AIMS 2004).

[5] B. O'Flynn, S. Bellis, K. Delaney, and C. O'Mathuna, "A 3-D Miniaturised Programmable Transceiver", XXVIII International Conference of IMAPS, 2004.

[6] B. O'Flynn, A. Barroso, S. Bellis, J. Benson, U. Roedig, K. Delaney, J. Barton, C. Sreenan, and C. O'Mathuna "The Development of a Novel Miniaturized Modular Platform for Wireless Sensor Networks", Proceedings of the IPSN Track on Sensor Platform, Tools and Design Methods for Networked Embedded Systems (IPSN2005/SPOTS2005).

[7] J. H. Gennari, M. A. Musen, R. W. Fergerson, W. E. Grosso, M. Crubézy, H. Eriksson, N. F. Noy, and S. W. Tu, "The Evolution of Protégé: An Environment for Knowledge-Based Systems Development" http://smi-web.stanford.edu/auslese/smi-web/reports/SMI-2002-0943.pdf (Technical report)

[8] F. Bellifemine, A. Poggi, G. Rimassa, "JADE – A FIPA-compliant agent framework" In Proceedings of PAAM, 1999.

[9] F. Bellifemine, A. Poggi, and G. Rimassa, "JADE: A FIPA 2000 Compliant Agent Development Environment," Proceedings of the International Conference on Autonomous Agent, ACM, 2001.

[10] T. O' Sullivan, J. O' Donoghue, J. Herbert, and R. Studdert, "CAMMD: Context Aware Mobile Medical Devices. International Journal of Universal Computer Science, Special Issue on Pervasive Health Management: New Challenges for Health Informatics, 2006.

[11] E. Friedman-Hill, "Jess in Action, Rule-Based Systems in Java", Manning Publications, Greenwich, 2003.