A Sensor-based, Web Service-enabled, Emergency Medical Response System

Nada Hashmi
10Blade, inc., nada@10blade.com
Dan Myung
10Blade, inc., dan@10blade.com
Mark Gaynor
Boston University, mgaynor@bu.edu
Steve Moulton
Boston University, smoulton@acs.bu.edu

Abstract:

Recent advances in low power wireless sensor networks are enabling new and exciting applications for wireless devices. At the same time, the power and flexibility of web services is expanding the power of the internet. Herein, we describe a scalable emergency medical response system that couples the efficient data collection of sensor networks with the flexibility and interoperability of a web services architecture

1. Introduction

Sensor networks consist of small, low-power, low-cost devices with limited computational and wireless communication capabilities. They represent the next step in wireless communication's miniaturization, and their power and size make it feasible to embed them into wearable vital sign monitors, location-tracking tags in buildings, and first responder uniform gear [1, 2].

Emergency Medical Services (EMS) systems need to communicate with hospitals from the field and exchange information about patient condition, expected time of patient arrival, and occasionally inquire about the ability to accept more patients [3]. An ideal EMS system should therefore provide real-time information and tracking of patients, staff and emergency vehicles. A web services architecture can meet these needs by addressing the problems associated with systematic application-to-application interactions over the web. This is because the key components of web services are a focus on interoperability and support for efficient application integration [4].

In this paper, we present a sensor-based, web service-enabled emergency medical response system. The paper begins with a brief overview of EMS systems in the United States. This is followed by a presentation of our system's architecture and data flow. Next, we review our preliminary experience with the system and lastly, discuss the future of this research.

2. Background

Emergency Medical Services (EMS) respond to sudden, unexpected events in oftentimes unfamiliar surroundings—where important information may be lacking or unknown until they arrive at the scene. The information gathered by Emergency Medical Technicians (EMTs) and paramedics is sometimes radioed ahead of the patient, depending on the severity of the patient's condition and the expected transport time. In all cases, the patient care report is summarized verbally during hand-off of the patient and a paper-based or electronic report later completed and added to the patient's hospital record. Within this process, tracking of the emergency response vehicle from the time of dispatch to scene arrival time, and finally to the health care facility is ad-hoc and poorly monitored [5].

In mass casualty situations, the current system often fails because EMTs and hospitals cannot effectively triage the injured and severely injured in a rapid, coordinated manner. Large numbers of victims can quickly overwhelm the triage process and prevent emergency field personnel and hospital staff from providing quality trauma care. Rapidly identifying and stratifying the most severely injured patients from those less severely injured poses a unique set of challenges, as does efficiently monitoring and transporting victims to an appropriate and awaiting trauma center.

The triage process has traditionally occurred at the hospital gate (typically at the entrance to the

Emergency Department). There, an emergency physician or experienced trauma surgeon stratifies patients with one or more triage tools based on mechanism of injury, physiologic criteria, injury site and severity, preexisting disease, age, and survival expectation. Over-triage, a common problem, can tie up valuable resources that could otherwise be made available for more severely injured patients. Emergency personnel must therefore decide, as early as possible, which patients will benefit most from transport to a dedicated trauma center and which patients require less immediate care [3, 5]. Clearly, there is a need for improved communication, documentation, and exchange of information between the pre-hospital and hospital phases of emergency care.

3. System Architecture

Figure 1 illustrates the overall system architecture of our emergency services application. The system has several major components including:

- A web services architecture to process, interpret, aggregate and present information
- A local command site for field coordination
- A central command site for global resource management
- Cellular/Satellite wireless links for real time communication between local and remote sites
- A wireless infrastructure for real-time data transport between motes and local PDAs and tablet PCs
- Patient sensors (a pulse oximetry sensor integrated with a GPS receiver, micro-processor, data storage & transmitter) for patient vital sign and location monitoring
- Wireless PDAs and tablet PCs for use by EMS field personnel

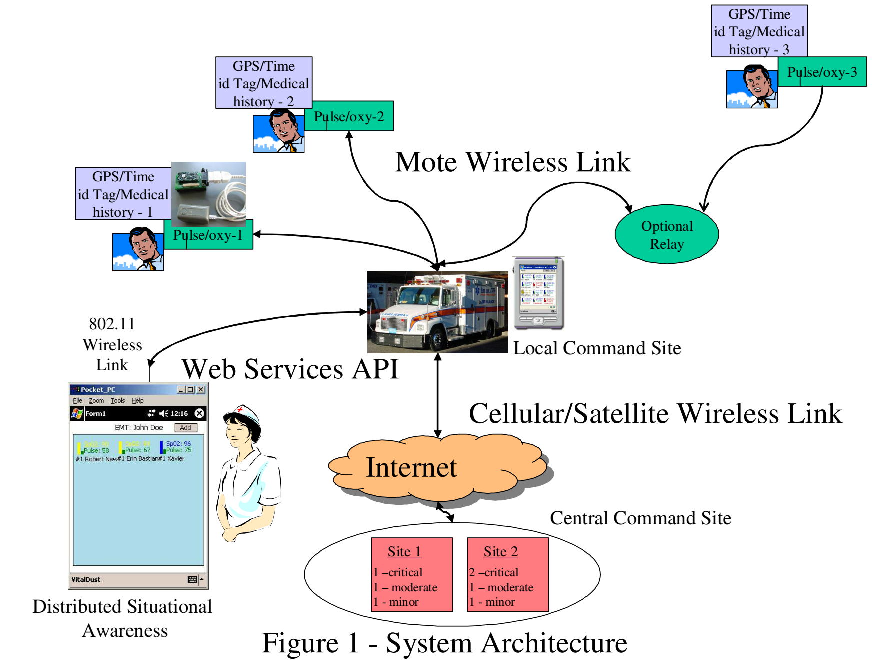

Figure 1 - System Architecture

Data from the patient sensors is combined on the mote and forwarded to the ambulance base station via a proprietary protocol. The local EMTs and paramedics use PDAs or tablet PCs to enter patient information as patients are evaluated and treatment is provided in the field; this information is linked to a real-time sensor data timeline by the web service application. Each patient's medical history is available in the field (even when disconnected from the internet), and at the central server (if connected to the internet). All data exchange (except between the sensors and base station) is based on emerging web services standards, encouraging interoperability by supporting the straightforward integration of real-time sensor data into applications and the easy exchange of sensor data between applications. The overall system goal is to provide secure, end-to-end real-time (including medical, environmental, and geo-location) information to first responders, who can then provide situational awareness for local decision support and the global management of resources.

Individual wireless, patient sensors are an important component of our system, for they provide real-time vital sign and patient location data both locally and centrally. When a patient is moved to an ambulance or temporary command tent, the vehicle or command center GPS overrides the GPS on the patient, thereby providing location data for all patients at that site. At the same time, however, each patient's individual vital sign sensor continues to transmit unique, real-time, vital sign data.

The patient sensor, shown in Figure 2, is based on the MICAz mote, developed at UC Berkeley in collaboration with Intel Research. This device consists of an 8-bit Atmel ATMEGA 128L micro controller, 132K of memory, 512K of nonvolatile flash memory, and a 19.2 Kbps radio operating in the 2.6 GHz spectrum. These motes run the open source TinyOS operating system. The mote is interfaced to a pulse oximeter (BCI, Inc.) and the Crossbow MTS420CA sensor board, which has a Leadtek 9546 GPS module. In addition to a GPS receiver, the MTS420CA has an onboard dual-axis accelerometer, a barometric pressure sensor, humidity sensor, and temperature sensor. The mote transports its data to a laptop computer (interfaced to a mote) in the onsite ambulance via 802.15.4. These motes provide a powerful platform for experimentation with both digital and analog sensors.

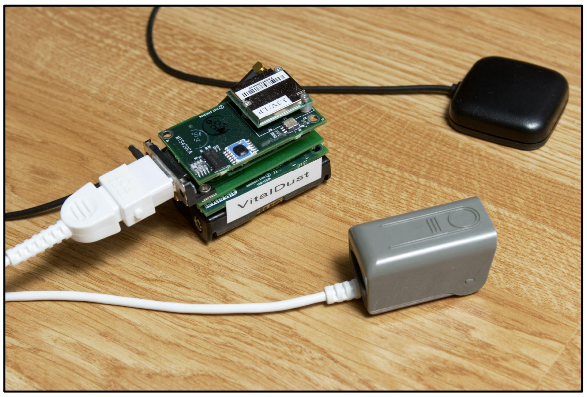

Figure 2: Patient mote with pulse oximeter and GPS sensors.

Sensor data must be "application friendly" to facilitate wide spread adoption of real-time sensor data within IT applications. Emerging standards for web service are a dominant design for the distributed exchange of data between applications. Our application has adopted these standards for both local and wide area exchange of real-time sensor data. Local personnel outfitted with applications on mobile devices have situational awareness with real-time access to sensor data via a web service. This local connectivity does not depend on a connection to the traditional internet. When connected via a cellular or satellite link, the real-time sensor data is available as a web service to centralized or distributed command centers. Compliance with emerging open standards (such as web services) enables a flexible architecture in the context of exchanging data between heterogeneous systems in both the local and wide area.

First responders provide situational awareness of the local, dynamic environment. Our system provides this with local PDAs and tablet PCs, which display data from local/remote sensors in real-time. These end devices have, in addition, embedded rules to help triage local patients based on their vital signs. Our end-user software provides several views of resources: a local view of patients being cared for a by single EMT, or a global view of all patients being treated by a group of EMTs. This allows a commander to "see" and respond to the needs of the EMTs and monitored patients, based on the available resources.

The central coordination of global resources is critical in the management of large scale (or multiple) events that span large geographical areas. In our system, aggregated sensor data is made available to a centralized, or group of distributed command and control centers, via web services. This allows for the development of command and control software in both Microsoft and Unix development environments.

Our emergency medical services application generates individual, electronic patient care records. These pre-hospital records are composed of manually entered patient information (history, physical exam findings, procedural information, and response to treatment) and automated sensor data. The manually entered patient information is captured in a SQL database and converted into an XML (HL-7) data format. The geo-location and pulse oximetry data (heart rate and blood oxygen saturation) generated by the motes is transmitted in XML and automatically integrated into each patient's electronic patient care record. These combined XML datasets can then be shared via web-service XML RPC calls. The resultant XML data is compatible and shareable with similarly structured web services. This means that the contextual healthcare data that is captured by our application could be combined with hospital medical record, laboratory, radiological, nursing and enterprise-wide demographic data. In the not too distant future, these large datasets will be mined for research purposes, provided the data is de-identified in accordance with current Health Insurance Privacy and Accountability Act (HIPAA) regulations.

4. Data Flow

Figure 3 illustrates the 3-layers of feedback and 5 categories of real-time dynamic data that drive the application's decision support system: at the edge, EMTs and paramedics have situational awareness with sensor data (1) and data from the local command center (2). The local command center receives patient specific healthcare and sensor data (3) from EMTs. At the highest layer the central command center receives aggregated data from each local site (5).

4.1 Edge Computing

The edge of an incident is where the EMTs and patients are physically located. Each EMT needs continuous access to real time sensor data from each patient to determine the triage order of all patients assigned to him or her. This data feed is represented as (1) in figure 3. It depicts the continuous stream of vital sign data from each patient under a specific EMT's control. There is also feedback from the EMT to the sensors (1). The sensor attached to each patient carries individual patient information. The primary concern in a crisis situation is that each EMT has situational awareness of their assigned patients based upon real time data (1). There is also a flow of data from the local command center to each EMT (2), which might be information about a noted change in a particular patient's vital signs, treatment suggestions, and/or instructions on where to transport the patient. Providing real-time data to the EMT allows for situational awareness of immediate time-critical responsibilities.

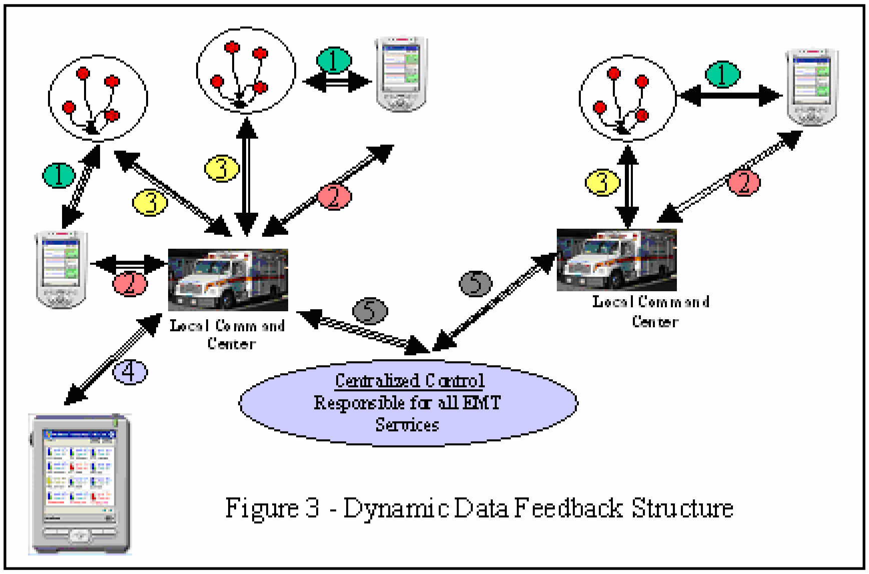

Figure 3 - Dynamic Data Feedback Structure

4.2 Local Command Center

Data flows from each sensor network (3) and is aggregated at the local command site along with data from each EMT (2). The local command center provides a view of all monitored patients in the sensor network, thereby allowing the effective management of local resources. One primary application at this layer utilizes a PDA to receive data from the local command center (4) to provide the EMT Commander a view of all local patients. Each local command center also receives data from the central command center for the effective coordination of global resources (5). The local command site enables a view of all local resources, along with data from the central command center.

4.3 Centralized

The top layer is the central command center. It acts as a web service client and receives data from each local site through the internet. Hence, there are no physical location limitations. This data is aggregated from each local command site (5) and includes real-time sensor as well as data input by local EMTs regarding the care of each patient. This data enables resources to be managed across a broad range of emergency events. The granularity of data required at the command center depends on the particular application. Systems that try to diagnosis or suggest treatment algorithms need fine grained data at the patient level. However applications for resource management across a distributed set of sites demands aggregated data in aggregated form. The overall architecture of our application enables data to flow between central and local command sites.

5. Evaluation

For proof of concept, an ambulance with two monitored "patients" was driven around our city. This urban setting, with its many high-rise buildings, was felt to be a challenging but realistic environment in which to test our wireless, sensor network infrastructure. We were especially interested in the accuracy and timeliness of the information relayed—both GPS location and vital sign sensor information. The two patients, each with a sensor mote, were loaded into the ambulance and GPS as well as vital sign (pulse oximetry and heart rate) data were relayed via the ambulance base station to a central command center. Time delays between the patient location (patient sensor) and actual location (ambulance GPS) were noted.

GPS satellites intentionally add a small error to civilian applications. While the accuracy is said to be about 100 meters, we were able to achieve accuracy to about 30 meters; the system used by the U.S. military has accuracy of 16 meters or less [6]. Pulse oximetry and heart rate were relayed in near real-time to the command center. There were, however, notable time delays of up to one full minute in relaying the GPS information. These time delays could be due to the implementation architecture of sensors or the use of web services over a wireless internet. Both Java and Microsoft .NET web service architectures were implemented.

While web services provide powerful and flexible service oriented architectures, they also introduce overheads such as the extraction of the SOAP envelope and parsing of the contained XML information. SOAP also requires typing information into every SOAP message, which creates bottle necks in synchronous applications. Binary data gets expanded (by an average of 5-fold) when included in XML, and also requires encoding/decoding of this data. These are the issues known over a wired internet. It is possible that these problems increase exponentially over a wireless internet, where there are bandwidth and connectivity issues. We are now in the process of conducting quantitative empirical studies to test web services over a wireless internet. The latency and through-put will be tested while the vehicle is standing still and at varying speeds. The data types and lengths will also be varied.

6. Future Work

Harvard University has an ongoing project called Hourglass, which is an internet-based infrastructure for connecting a wide range of sensors, services, and applications in a robust fashion [7]. In the coming year, we plan to integrate our system with Hourglass to take advantage of its scalable, robust architecture. Furthermore, the empirical experiments will provide guidance on optimizing our application to cater to various types of unexpected situations. Other sensors to collect other data such as EKG information will also be integrated. In addition, a recently formed partnership with our regional aero-medical group will serve as a real-life test bed for the evaluation of our system.

7. Conclusion:

Sensor networks combined with web services offers a unique system for EMS and other pre-hospital patient care systems. Our preliminary results suggest that the architecture described herein is sufficiently robust and scalable to meet the needs of our changing world.

References:

1. J. Hill et al., "System Architecture Directions for Networked Sensors," *Proc. 9th Int'l Conf .architectural Support for Programming Languages and Operating Systems* (ASPLOS 2000), ACM Press, 2000, pp. 93–104.
2. T.R.F. Fulford-Jones, G. Wei, and M. Welsh, "A Portable, Low-Power, Wireless Two-Lead EKG System," to be published in Proc. 26th IEEE EMBS Ann. Int'l Conf., 2004.
3. E.R. Frykberg and J.J. Tepas III, "Terrorist Bombings: Lessons Learned from Belfast to Beirut," Annals of Surgery, vol. 208, no. 5, 1988, pp. 569–576.
4. F. Curbera, W.A. Nagy, and S. Weerawarana. Web services: Why and how. In OOPSLA 2001 Workshop on Object-Oriented Web Services. ACM, 2001. pp. 34
5. Mann, Glenn Eric. Data Capture in the United States Healthcare System- the Pre-Hospital Phase of Patient Care. Masters Thesis. Boston University School of Medicine-Division of Graduate Medical Sciences. 2002
6. Highly, Sam. GPS use expanding by 2nd Lt. http://gislounge.com/freisin/gps.shtml
7. J. Shneidman, P. Pietzuch, J. Ledlie, M. Roussopoulos, M. Seltzer, M. Welsh. Hourglass: An Infrastructure for Connecting Sensor Networks and Applications Harvard Technical Report TR-21-04