

# A Dynamic, Data-Driven, Decision Support System for Emergency Medical Services

Mark Gaynor<sup>1</sup>, Margo Seltzer<sup>2</sup>, Steve Moulton<sup>3</sup>, and Jim Freedman<sup>1</sup>

<sup>1</sup> Boston University, School of Management, 595 Commonwealth Avenue, Boston  
{mgaynor, jfreedma}@bu.edu

<sup>2</sup> Harvard University, Division of Engineering and Applied Sciences  
margo@eecs.harvard.edu

<sup>3</sup> Boston University School of Medicine, Boston MA  
smoulton@bu.edu

**Abstract.** In crisis, decisions must be made in human perceptual timeframes under pressure to respond to dynamic uncertain conditions. To be effective management must have access to real time environmental data in a form that can be immediately understood and acted upon. The emerging computing model of Dynamic Data-Driven Application Systems (DDDAS) fits well in crisis situations where rapid decision-making is essential. We explore the value of a DDDAS (iRevive) in support of emergency medical treatment decisions in response to a crisis. This complex multi-layered dynamic environment both feeds and responds to an ever-changing stream of real-time data that enables coordinated decision-making by heterogeneous personnel across a wide geography at the same time.

## 1 Introduction

Information technology has been associated with a shift in business strategy such as the shift in decision-making paradigms of the organization [1]. The emerging computing model of Dynamic Data-Driven Applications Systems (DDDAS) fits well in crisis situations where rapid decision-making [5] is essential. Effective decision-making in a crisis greatly depends upon immediate access and interpretation of local information within the context of the overall environment at any particular point in time. A key element of a crisis is its dynamic, turbulent, uncertain nature that creates both a sense of urgency and a need to reduce uncertainty in decision-making.

We view theories of network organizational decision-making [2][3] as the foundation for experimentation with emerging sensing networks and dynamic data-driven application technologies. The goal of these emerging technologies is to support decision-makers with relevant timely information that they can quickly interpret at the time and location that they must make a decision [4]. In this paper, we use emergency medical response to a mass casualty event as an illustrative example of how emerging technology supports a shift in decision-making during a crisis. In a

medical emergency situation, many split-second decisions are made in the process of locating, identifying, evaluating, treating and transporting a critically ill patient. When emergency medical personnel are dealing with multiple patients, all with various types of life or limb threatening injuries, critical decisions regarding who to treat, who not to treat, and in what order, must be made with split second accuracy to reduce the number of casualties as well as the number of patients with long term disabilities. Our emergency medical response application receives real time sensor and emergency resource operational data from the edge of the incident and presents this information to emergency medical personnel at the time and place that they need it to enable more effective patient management, including decisions that may effect human survival rates.

Mass casualty events (MCEs) are sudden, unexpected situations in which, over a short period of time, large numbers of casualties are generated and organized community support mechanisms are either crippled or destroyed [6]. A central problem in mass casualty events is the need to rapidly assess and respond to the situation by deploying multiple personnel and resources to monitor, triage, and track critical care patients. If the injured and severely injured cannot be triaged in a rapid and coordinated manner, their large numbers can quickly overwhelm emergency field personnel and hospital staff preventing them from providing quality trauma care [7]. A variety of problems may arise at large scale events, beginning with the inability to locate or gain access to patients in distress because of collapsed infrastructure, crowds, noise and other event specific circumstances (e.g., flooding or radiation). In addition, documenting care and monitoring patient status may become sporadic and disorganized due to multiple calls for assistance or many patients at one location within an event. All of these factors can lead to delays in recognizing when patients deteriorate, errors when administering medications, and incomplete capture of patient information, including histories of the present illness, physical exam findings, and response to treatment. Furthermore, available patient treatment and outcome data may be lost during the triage and transfer process, as patients are handed from one team to another on their way to more definitive care.

In order to address the problems faced by EMT workers, we have developed iRevive, which is a mobile pre-hospital database system that allows point-of-care data capture in an electronic format. It consists of a network of wireless, handheld computers running the iRevive application, wireless patient location and vital sign sensors, a multi-frequency ambulance base station, and a central command center where field information is collected, viewed and stored. Wireless VitalDust sensors automatically capture and integrate patient location (GPS) and vital sign data (including heart rate and blood oxygen saturation) into each patient's pre-hospital record (Figure 1). This collated information is wirelessly and securely pushed over the Verizon 3G Wireless cellular network to a server that stores and relays selected patient information to a command center and hence to the receiving hospital.

Patient location and vital sign data is locally accessible to on-scene personnel to assist them with triage and decision-making processes. This same data is available in near real-time at the central command center, where a global view of the field

environment can be obtained and a greater semblance of order established. A properly trained triage officer at the central command center is better able to assist field personnel in determining the best course of action for each patient when that officer has immediate access to global information about the field environment and medical resources that are available at the trauma center, including bed and Operating Room availability, on-site staffing and transportation resources. Triage in the field and triage at the hospital gates therefore becomes an interactive, coordinated process that better matches the out-of-hospital caseload with critical, hospital-based trauma facilities and resources. More appropriate levels of medical care can be provided, thus limiting the number of casualties and reducing the number of patients with long-term disabilities.

Our emergency medical application is a Multi-Layer Dynamic Data-Driven Decision Support System. It provides decision support at three important layers: at the site of the incident, at local command centers, and at a central point of coordination. These three points form a hierarchical layering with on-site care at the bottom and central coordination at the top. At each layer, data is aggregated from layers below (similar to the different time variant hierarchical structure of Chaturvedi [5]). Each layer presents data aggregated from a dynamic set of real-time sensors and mobile Emergency Medical Technicians (EMTs). By linking real-time sensor data, procedural data, and geographic data to specific points in time, iRevive enables rapid decision support that considers actual EMT intervention and patient outcomes, as they occur to determine optimal future action.

## 2 iRevive Application Description

Figure 1 illustrates the overall system architecture of the triage/Geographic Positioning System (GPS) application (iRevive) [8][9]. It describes the current system under development (supported by NSF, NIH, and US army). This system has several major components including: Web Services to aggregate and present information, handheld mobile Personal Digital Assistants (PDAs) to provide situational awareness to each EMT, local command site for field coordination, central command site for global resource management, cellular/Satellite wireless link for real time communication between local and remote sites, infrastructure for real-time data transport between motes and local PDAs, and Vitaldust (Developed by Matt Walsh at Harvard University) patient tag (pulse/ox sensor, GPS receiver, micro-processor, data storage & transmitter) attached to patient.

On each patient, data from the GPS and medical sensors is combined on the Vitaldust patient tag and forwarded to the ambulance base station via a proprietary protocol. The local EMTs use their PDAs to view current vital-sign information and to enter patient medical information as treatment is performed in the field. These end-user devices also have business/medical rules, embedded in applications resident on the PDA, that help triage a group of patients based on their vital signs. Our end-user software provides several views of resources: a local view of patients, to assist a particular EMT, or a global view of all patients being treated by a group of EMTs,

which permits a commander to coordinate resources in the immediate area of an event. The overall system goal is to provide secure, end-to-end, real-time (including medical, environmental, and geolocation) information to first responders that allow situational awareness for local decision support and global management of resources. Aggregation of sensor data is made available to a centralized, or group of distributed command and control centers, via web services. Our emergency medical services application, iRevive, provides an electronic pre-hospital patient care record that combines manual entry of procedures provided in the field with real-time sensor data. This link enables valuable event data that can facilitate later mining to help determine better medical field procedures.

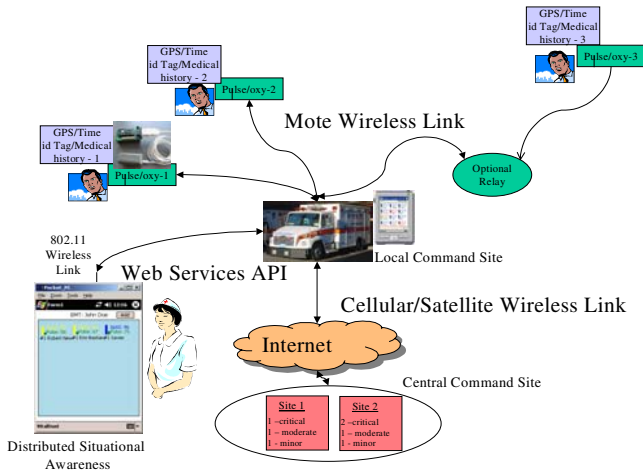


Fig. 1. iRevive System Architecture

The Vitaldust patient tag is based on the MICA 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 Kpbs radio operating in the 433 MHz or 916 MHz spectrum. These motes run the open source TinyOS operating system. The mote is interfaced to a pulse/oximeter and the Crossbow MTS420CA MICA2 Sensor board with a Leadtek 9546 GPS module. In addition to location and time, the MTS420CA also provides onboard dual-axis accelerometer, barometric pressure, humidity and temperature. The mote transports its data to a laptop computer in the onsite ambulance via a proprietary protocol (however, the industry is beginning to standardize around 802.15.4). These motes provide a powerful platform for experimentation with both digital and analog sensors.

Sensor data must be “application friendly” to facilitate widespread adoption of real-time sensor data within IT applications. Currently, the emerging web services standards are the dominant design for 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 utilizing 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 Internet connectivity. 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.

The triage aspect of iRevive fits well into the paradigm of a DDDAS because its current view of the emergency response situation is based on a continuous flow of real-time, vital-sign sensor data combined with information input by the local EMT. Figure 2 illustrates the dynamic aspects of our triage application – the left side is the stable view, the right is a shifting view caused by changes in patients' conditions, which are based on changing vital signs. In future editions of iRevive, the medical rules that determine triage order will vary based on feedback from the EMT. Triage is a multi-layer application – each EMT has triage information based on the patients for whom they are responsible and each local commander has triage information for all local patients. The triage function aids local EMTs in their decision making process. Central coordination of global resources is critical to manage large-scale (or multiple) events that span large geographical areas.

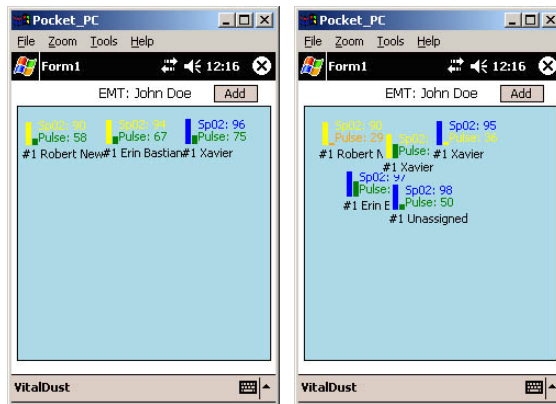


Fig. 2. Dynamic triage function in iRevive

## 2.1 Multi-layer Architecture

Figure 1 illustrates the three layers of decision-making that drive the iRevive decision support system: at the edge EMTs have situational awareness with sensor data and data from the local command center. The local command center receives both sensor data and procedural data from EMTs. At the highest layer the central command receives aggregated data from each local site and can also send real-time information to the local command centers.

The edge of an incident is where the EMTs and patients are physically located. Each EMT needs continuous access to real time sensor data of each patient to

determine the triage order of all patients assigned to them. This is accomplished by sending each EMT a continuous stream of vital sign data from each patient under the EMT's control. The primary concern in a crisis situation is that each EMT has situational awareness of their assigned patients based upon real time data. Information also flows from the local command center to each EMT, which might include data about a particular patient, treatment suggestions, or general instructions to the EMT. Providing real-time data to the EMT allows for situational awareness of their immediate time-critical responsibilities.

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

The top layer is the central command center that receives data aggregated from each local command site and includes real-time sensor as well as data input by each EMT about the care of each patient. This data enables resources to be managed across a broad region with many emergency events. The granularity of data required at the command center depends on the particular application. Systems that try to diagnose or suggest treatment algorithms need fine-grained patient-level data. However applications such as resource management across a distributed set of sites demand aggregated data. The overall architecture of iRevive enables data to flow between central and local command sites.

### **3 Limitations of Current Architecture**

As described so far, iRevive relies on a centralized architecture, with a single ultimate destination for all data (the centralized control). While this is feasible in the short run, it is not a long-term, scalable solution. In reality, a major metropolitan area has a number of ambulance companies, multiple hospitals, and no central dispatching system that coordinates among them. Thus, the architecture does not reflect the reality of the particular application.

In a more general sense, the architecture as described here has fundamental limitations in its flexibility and scalability. In particular, there are a number of issues that it does not address: the ability to balance data flows across potentially overloaded network links, the ability to direct data to underutilized nodes, intermittent or disconnected operation in the presence of extremely high-bandwidth data streams, support for in-network data processing (e.g., aggregation, filtering, triggering, etc.), and optimizing the placement of in-network services to minimize overall bandwidth utilization

In the next section, we present a general architecture that addresses these limitations and provides a fundamentally new abstraction for the delivery of streaming, real-time data.

## 4 Hourglass

Hourglass is the networking infrastructure that we are developing to support the iRevive application environment. It is a particular application and implementation of a new abstraction called a Stream-based Overlay (SBON). Stream-based overlays are designed to provide the networking infrastructure to support dynamic, data-driven applications. Applications, similar to iRevive, are characterized by long-lived streams of real-time data that may demand high bandwidth. A network that is subject to multiple dynamic data-driven applications will easily become overloaded, so the infrastructure must be able to adapt to a rapid influx of data and reconfigure itself appropriately so as to avoid network overload, make efficient use of nodes in the network, and provide acceptable quality of service for the delivery of data from their source to their, potentially multiple, destinations.

The fundamental unit of dataflow in Hourglass and an SBON is the circuit. A circuit is a logical connection between one or more data producers and a data consumer, including zero or more in-network services. A service might be a filter (e.g., ignore all “normal” pulse readings), a trigger (e.g., perform some action whenever you see a pulse rate above 180 beats per minute), an aggregator (e.g., what is the average time from first contact to ambulance access), or an application-specific computation (e.g., match patients with particular emergency rooms). At any point in time, a common infrastructure may support many such circuits. The technical challenge in Hourglass is to realize a physical implementation of a logical circuit. This process entails locating the data producers and consumer, instantiating the appropriate services on intermediary nodes, and then setting up the particular data flows. As the data flow through these circuits, the system must also adapt to changes in network load, node load, and flow rate. This adaptation usually involves moving services around, instantiating new services, or denying admission to circuits that cannot adequately be supported.

Most of these technical challenges can be approached as an optimization problem. That is, we strive to minimize total network load subject to the response time constraints of a particular circuit description. More formally, a circuit is composed of some number of services that fall into one of the following categories: pinned, unpinned-unplaced, unpinned-placed. A pinned service is one that must reside on a particular node in the network. Data producers, such as sensors, are examples of pinned services; the system has no control over where these services can be located. An unpinned-unplaced service is one that can run anywhere in the network and that has not yet been assigned to a particular node. An unpinned-placed service is one that is free to run anywhere, but has already been assigned to a particular location. Our challenge then, is to place all the unpinned-unplaced services in a manner that minimizes network load subject to the application constraints. These constraints

might be response-time constraints or they might be social constraints (e.g., only a system under the control of an ER can assign a patient to a bed). Also, in the face of changing load conditions, the system may need to change the placement of unpinned services to continue meeting application constraints.

In our prototype Hourglass implementation, we have developed a decentralized network-aware service placement algorithm called Relaxation placement [10]. The main idea behind Relaxation placement is to partition the placement problem into two phases. First, an unpinned service in a circuit is placed in a virtual metric space (using latency) using a spring relaxation technique. Then, we map the virtual placement back into a real physical placement in the network. Performing data path optimization in our latency space has the advantage of naturally capturing global knowledge about bandwidth and latency in the network topology without imposing a large overhead due to network probing. Using our distributed spring-relaxation placement algorithm allows us to do this without requiring any centralized or global knowledge.

As load conditions change (either in the network, from the data source, or on particular nodes), our optimizer will move services as necessary. We build dampening into these optimization movements to avoid continual thrashing and reconfiguration.

While Hourglass is a solution specific to data sensor networks, these ideas and SBONs are applicable to the entire class of dynamic, data-driven applications, where consumers and producers enter and leave the system and bandwidth availability changes in unpredictable ways.

## 5 Conclusion

This paper described the iRevive EMT application in the context of a DDDAS. The multi-layer architecture illustrates the flexibility of the web services approach in sharing data among heterogeneous data sources. The feedback of real-time data from sensors, and other data sources provides multiple layers of decision support through iRevive. The limitations of the traditional web services model is described, and a new, more robust and scalable distributed data collection infrastructure called Hourglass is discussed.

## Acknowledgments

NSF (ACI-0330244 and PFI-0227879), NIH (R41 RR018698-01A1), and US Army (Grant Contract No.:W81XWH-04-C-0113), and Matt Welsh and his research group.

## References

1. Huber, G. P. (1990). "A Theory of the Effects of Advanced Information Technologies on Organizational Design, Intelligence, and Decision Making." *Academy of Management Review* **15**(1): 47-71.



2. Baldwin, C. and K. Clark (1997). "Managing in an Age of Modularity." *Harvard Business Review*, 75(5): 84-93.
3. Alberts, D. S. and R. E. Hayes (2003). Power to the Edge.
4. Strassner, M. and T. Schoch (2002). Today's Impact of Ubiquitous Computing on Business Processes. *International Conference on Pervasive Computing*.
5. Chaturvedi, A, Synthetic Environments for Continuous Experimentation – Working Paper
6. American College of Surgeons Committee on Trauma 1998. In: Resources for Optimal Care of the Injured Patient, 1999. Chicago, IL: American College of Surgeons, 87-91.
7. Frykberg ER, and Tepas JJ III, 1988. Terrorist bombings: lessons learned from Belfast to Beirut. *Ann Surg* 208:569 576.
8. Gaynor, M., and Welsh, M, and Moulton, S, Integrating Wireless Sensor Networks with the Grid, *IEEE Internet Computing*, special issue on the wireless grid, July/Aug 2004
9. Tollefsen, W, and Pepe, M, and Myung, D, and Gaynor, M., and Welsh, M, and Moulton, S, iRevive, a Pre-hospital Mobile Database for Emergency Medical Services, *International Journal of Healthcare Technology Management (IJHTM)* , Summer 2004
10. Shneidman, J., Pietzuch, P., Welsh, M., Seltzer, M., Roussopoulos, M., "A Cost-Space Approach to Distributed Query Optimization in Stream-Based Overlays," Accepted, *The Proceedings of the 1<sup>st</sup> IEEE International Workshop on Networking Meets Databases (NetDB'05)*, Tokyo, Japan, April 2005.