ORIGINAL ARTICLE

A Secure and Expandable Electronic Patient Record System Using Web-based Technology

Michael Howie[1], Carlos del Río[1,2], Furrukh Khan[1,2,3], Luis Lopez[1], Roger Dzwonczyk*[1], Sergio Bergese[1,4]

[1]Department of Anesthesiology

[2]Department of Electrical and Computer Engineering

[3]Department of Computer Science and Engineering

[4]Department of Neurosurgery

*Corresponding author: Roger Dzwonczyk Email: dzwonczyk.1@osu.edu
Published: 01 December 2009
Ibnosina Journal of Medicine and Biomedical Sciences 2009, 1(3):73-79
Received: 10 August 2009
Accepted: 26 September 2009
This article is available from: http://www.ijmbs.org

Abstract

Introduction: In its 2000 report regarding medical errors, the Quality Interagency Coordination Task Force highlighted the unacceptably high rate of errors in health care. The report recognized that information technology (IT) offers a tremendous opportunity to reduce errors and recommended the adoption of electronic patient records (EPRs). However, providing EPR-enabled solutions is challenging due to the constraints of security and confidentiality imposed by the healthcare domain. Adding to the complexity is the need for healthcare organizations to exchange confidential EPRs safely across the security policy boundaries of healthcare enterprises and its partners (e.g. insurance companies).

Methods: We developed an EPR system that utilizes Microsoft Web Services architecture.

Results: The current system has three components: OR-Eye, OR-Med and OR-Track. The system was built using Web-based communication technology; it offers the security, stability and expendability required for a hospital EPR system. Our most developed component of the EPR system, OR-Eye, allows authorized users to view real-time and achieve intraoperative patient data wirelessly in a graphical form similar to an intraoperative anesthesia record on a variety of available devices. A unique feature of OR-Eye is a time line graphic that displays the sequence of patient and hospital actions that can be viewed in detail and strung together to form various hospital records.

Conclusion: Our EPR system is secure, stable and expandable. It interfaces with our existing hospital wireless network. It has the potential to improve patient care, patient safety and hospital efficiency as well as enhance medical research and medical education.

Implications Statement

We built our electronic patient record system using Web-based architecture. The system is secure, stable and expandable. The system has the potential to improve patient safety, patient care and hospital efficiency as well as enhance medical research and medical education.

Introduction

The Quality Interagency Coordination Task Force (QuIC), in its 2000 report regarding medical errors (1), highlighted the unacceptably high rate of errors in health care. Among its recommendations was the introduction of electronic patient records (EPRs). However, providing an EPR system is challenging due to the constraints imposed by the healthcare domain on security, confidentiality, expandability and dependability. Exponentially compounding the complexity and cost are the additional constraints imposed by the needs of healthcare organizations to exchange confidential EPRs safely across the security and policy boundaries of healthcare enterprises. Shortliffe (2) highlighted lack of standardization and security as well as difficulty in data entry and integration as the major factors that have traditionally limited the use of EPRs. These systems are also limited by lack of standardized infrastructures that would enable exchanging confidential data across the policy (e.g. security policy) boundaries of healthcare enterprises, as well as other organizations such as insurance companies.

These limiting factors are strongly interrelated. A cohesive approach, based on standards and robust technologies, to this problem is required. Substantial progress has been made in the industry recently in the form of new standards and enabling technologies to take a holistic view of the problem and offer a promising new approach to building EPR-enabled infrastructures for health care. Some promising key standards, structured on Web-based services, have already been available for the past few years.

The idea of EPRs is not new, particularly in the intraoperative setting. Duke University developed the DAMES and microDAMES systems in the 1970s (3). During this same time period, The Ohio State University Medical Center (OSUMC) Department of Anesthesiology also developed an EPR system when the microcomputer industry was in its infancy. At the time, there were virtually no high level system resources available that could be used as building blocks for an integrated patient monitoring system. The patient monitoring operating systems and graphical user interfaces were created line by line using scientific and machine languages. Clearly, this was a coding-intensive undertaking. Several commercial products came to market that evolved from this early research and development work (4). However, these devices were inflexible from both the hardware and software standpoint and did not establish a lasting clinical foothold.

In this paper, we describe an EPR system under development at OSUMC that utilizes Web-based programming standards that meet the crucial stringent needs of an EPR data communications system.

High-Level Requirements of an EPR System

A successful EPR system must meet the following requirements:

Expandability: The system must be loosely coupled so that future data sources can be added easily. The solution has to pull together data from and facilitate communication between many isolated data sources and functionalities, much in the same way that the various functions in a hospital each have their own systems and processes yet still communicate to coordinate activities and share data. In addition, the solution must serve other essentially unaffiliated entities that have academic, clinical and/or business relationships with the hospital.

Security: To meet healthcare industry requirements for confidentiality of patient data (Health Information Portability and Accountability Act requirements), all information passed between the distributed services needs to be encrypted. It would be preferable to utilize available security mechanisms to avoid the daunting task of coding these mechanisms from scratch.

Stability: The solution also needs to be robust and self-healing so that when a monitor is unplugged or another system is disconnected from the infrastructure that system performance or stability would not be disrupted.

With these requirements in mind, we realized that a Web-based approach to the EPR design would ideally meet the design objectives. The internet Web as we know it today is nearly infinitely expandable. Computers in all forms are connected to and disconnected from the Web constantly with essentially no effect on the system as a whole. Great strides have been made in the security of information sharing, particularly by the banking and financial industries. Many new standards, policies and procedures have been developed in the process which have made the development of high-level information sharing systems secure, stable and dependable.

Description of the OSUMC EPR System - Actions and Records

The current system has three components or subsystems, OR-Eye, OR-Med and OR-Track. Each of these subsystems feeds and retrieves data from a generalized master database in an operational manner similar to the way we send and access data over the Web. The data are stored on a secure remote server developed with Microsoft SQL Server 2000 database technology. Web Services Enhancements (WSE) 2.0 for Microsoft .NET, an add-on to the Microsoft

.NET Framework that implements Web service protocol specifications including WS-Security, WS-Trust, WS-Policy, and WS-SecureConversation, are used to secure confidential patient data as it passes between distributed solution components.

The data are stored as actions on the Web service-oriented architecture that supports OR-Eye, OR-Med and OR-Track. For example, there are vital signs actions, medication actions, patient location actions and procedure actions, each of which is generated by a different Web service that sends information to the common database for storage. All actions are time stamped. The system's open architecture will allow for the flexible addition of new actions in the future, such as radiology actions or pharmacy actions. Since the actions are time stamped, they form a continuous timeline (Figure 1).

A record is a sequence of actions strung together. For example, a complete anesthesia record for a patient can be constructed by extracting all vital signs actions, medication actions and procedure actions during the intraoperative time interval for that patient from the common database (Figure 1). Other types of records can be generated in the same manner. For instance, a complete patient record can

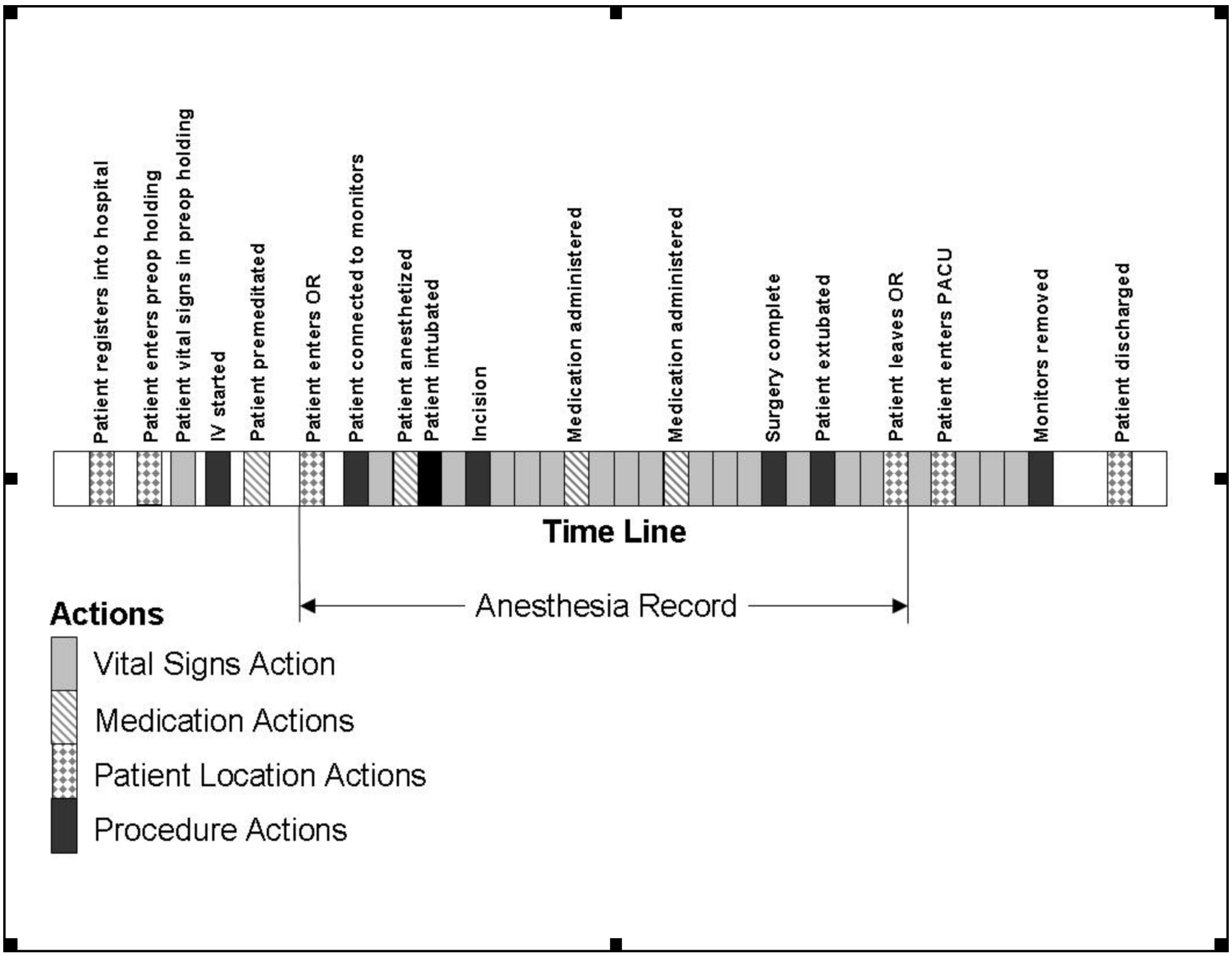

Figure 1. The concept of actions and records. Actions are repeatable events. This simulated timeline defines four distinct actions; however, the number of possible unique patient or hospital actions is essentially unlimited. Stringing together particular actions stored in the master database produces data records. For instance all actions occurring during the intraoperative period, from the time the patient enters the operating room to the time he/she leaves the operating room, produce an intraoperative anesthesia record.

be generated by extracting all actions for a patient from the database. Similarly, an operating room record that shows all activity for a given operating room can be assembled by querying the database for all actions associated with that room, or a monitor record can be generated by querying the data store for all actions associated with a monitor.

This model maps extremely well to a loosely coupled, service-oriented architecture based on Web services. In the case of OR-Eye, one Web service collects information from the proprietary monitor network and sends that information to another Web service that services the centralized data store. The OR-Eye smart client interfaces with the system through a third Web service, without having to be concerned about how that Web service interacts with the other Web services in the federation to provide the requested data.

Additional Web services serve a similar function for OR-Track and OR-Med, with each Web service capturing a single type of action and sending to the database. Altogether, the service-oriented architecture includes some 13 Web services, which all communicate with each other and with the smart clients that they support using advanced security mechanisms to help protect confidential patient data in a manner similar to devices on the internet.

Description of the Current Subsystem - OR-Eye

OR-Eye allows authorized users to securely and remotely monitor and replay actions that are generated in the operating rooms and intensive care units. OR-Eye consists of two parts: Service-oriented architecture that extracts vital signs data from the proprietary patient monitor network and securely and efficiently exposes information for access by the rest of OSUMC. Implemented as a federation of Microsoft .NET Web services (discrete application components that can be programmatically accessed using industry standard Web protocols), the loosely coupled, service-oriented architecture allows for easy addition of new data sources and functionalities and provides access to data and functionalities by new and existing interface devices.

Smart client applications use secure web services to communicate with the service-oriented architecture. OR-Eye provides a graphical user interface that allows the authorized user to intuitively view and replay actions from patients in the operating room or intensive care unit on smart clients such as desktop personal computers (PCs), laptops, Tablet PCs, iPODS, iPhones and automobile global positioning system screens. We have also developed a mobile version of OR-Eye that runs on Pocket PCs. The authorized user can access data from anywhere on OSUMC's wireless network or from remote locations using the OSUMC's virtual private network (VPN).

Figure 2 shows the main OR-Eye smart client application screen. The screen has three distinct areas. The patient selection area on the left side of the screen allows the user to choose which patient in the hospital to display on the screen. Graphical and digital vital signs data are displayed in an area that resembles both an intraoperative anesthesia paper chart and the patient monitors that are used in the operating rooms or at the bedside. The timeline shown in Figure 1 is provided on the screen and makes it easy for the user to rapidly move forward and backward when viewing data. The player controls in the upper left hand corner of the screen work in conjunction with the timeline to navigate through a period of archived patient data. The privacy button, when activated, hides any patient-identifying data for confidentiality. The system provides two buttons, Live and Achieve, to shift between an ongoing case in the operating room or intensive care unit (live) or review data from a past case.

Description of Other Planned Subsystems - OR-Med and OR-Track

OR-Med, the second subsystem that plugs into the OR-Eye service-oriented architecture, has undergone pilot phase development. Through the OR-Med smart client, anesthesiologists will be able to enter the medications that they give to a patient intraoperatively, with information stored in the master database. That information can then be viewed with the OR-Eye smart client, with medication actions appearing on the OR-Eye graphical timeline (Figure 1). This capability allows doctors, students and researchers to visually correlate the administration of a medication with the effect that it has on a patient's vital signs.

In the future, the service-oriented architecture will extend OR-Med to integrate with OSUMC's pharmacy system. At that point, an anesthesiologist in the operating room will be able to enter a medication and intended dosage, and have that data immediately sent to the pharmacy where it can be checked for drug interactions, patient allergies, dosing errors and other potential patient-specific contraindications. Results will be displayed on the OR-Med smart client graphics screen, providing the anesthesiologist with additional real-time information that can be used to help ensure patient safety and reduce medication errors.

In building the OR-Med smart client, we are taking advantage of forms created with the Microsoft Office InfoPath 2003 information-gathering program, providing a prebuilt interface with which users can view and enter data. InfoPath 2003 helped us provide rich, intuitive forms that are pre-populated with patient information, which therefore minimizes the time required for data entry and, again, reduces entry errors.

OR-Track is the planned third subsystem. This far-reaching subsystem will use Imote2 technology (Intel Corporation, Santa Clara, CA USA) to track patients' physical locations in the hospital. Imote2 is an advanced high performance wireless sensor network that utilizes the IEEE 802.15.4 communications standard. The technology is less expensive than the more familiar radio frequency identification (RFID) system. Imote2 devices will communicate with OSUMC's current wireless network.

Like OR-Eye and OR-Med, OR-Track plugs into the service-oriented architecture and uses its centralized database. We will have sensors strategically placed throughout the hospital. As a patient moves from location to location, the patient's position will be displayed graphically on a flat panel TV screen mounted near the operating room desk or on any authorized client display (a pocket PC logged into the system, for instance). Changes in a patient's location will cause repopulated InfoPath forms to appear on PCs in the appropriate areas for nurses to complete or modify.

OR-Track will not only track the location of the patient but will also track the current procedures that the patient is undergoing. For example, in the operating room, a form can be presented with a button to click when anesthesia is administered, when the surgeon approaches a patient, when the first incision is made, when the surgery is completed, when the operating room is ready for the next patient, and so on, with real-time visibility of the information provided throughout OSUMC. Like medication actions, patient actions and status changes captured by OR-Track can be recorded, replayed, and viewed on the OR-Eye timeline along with medication and vital signs data, enabling users to easily see the movement of a patient and correlate that

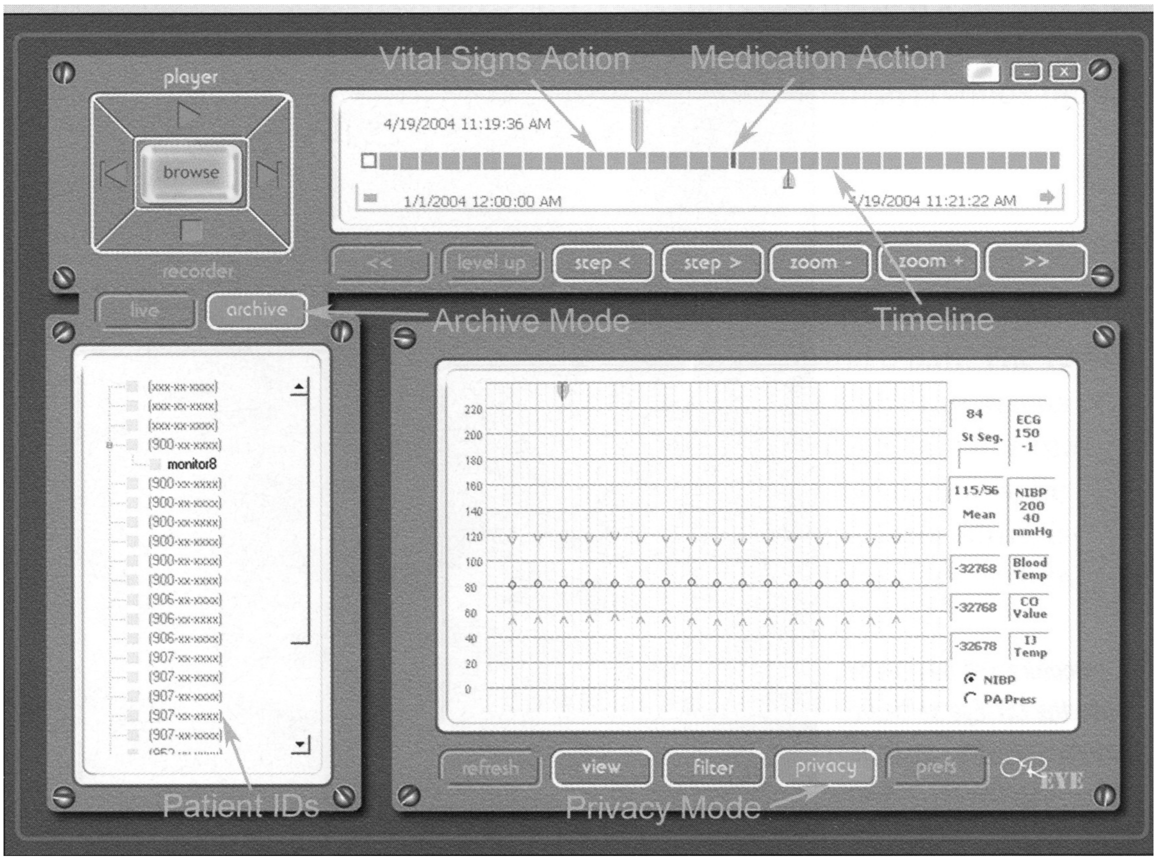

Figure 2. The OR-Eye main operating screen. OR-Eye allows the authorized user to access real-time or achieved data from the master database for a selected patient. Included on the main operating screen is the timeline (Figure 1) that displays the sequence of actions. The graphical interface allows an authorized expert to consult in a patient's care wirelessly and remotely, without being by the patient's side.

.

information with other data collected by the system.

The information collected in the common database assembled and used by OR-Eye, OR-Med and OR-Track will have immense value in all aspects of the operation of the hospital. From the clinical perspective, physiologic and medication data can be accessed to review particularly challenging cases and provide information on how to improve patient care in a program of continuing medical education. OR-Med, when fully implemented, will provide an extra measure of patient safety with respect to dosing levels and contraindication alerts. The data from all three subsystems will provide automatic hospital billing information to allow for accurate accounting and billing of all hospital medical services in much less time. Virtually everything can be tracked by querying the common database, including compliance issues.

Security

All subsystems are structured using the Microsoft Visual Studio .NET 2003 development system and run on the Microsoft .NET Framework, an integral component of the Windows operating system that provides a programming model and runtime for Web services, Web applications, and smart client applications. In building the system, we had to protect confidential patient data as the loosely-coupled system components communicated with each other.

We achieved our security goals by using Web Services Enhancements 2.0 (WSE 2.0) for Microsoft .NET, a supported add-on for Visual Studio .NET and the .NET Framework that gave us prebuilt support for evolving Web service protocol specifications. Features of WSE 2.0 that we used included support for:

WS-Security, which describes standards for how message headers can carry or refers to various types of security tokens used to help ensure message integrity and confidentiality through electronic signatures and encryption.

WS-Trust, which describes a standard model for establishing trust relationships, including standards for requesting and exchanging security tokens with a trust services.

WS-SecureConversation, which makes use of WS-Security and WS-Trust, and describes standards for establishing mutually-authenticated security contexts that enable long-running, secure and efficient conversations between message senders and receivers.

WS-Policy, which describes standards for how message senders and receivers can declaratively specify their policy requirements. WS-SecurityPolicy, part of the WS-Policy specification, was used to specify security requirements for solution components in a way that helped to separate business logic from security and avoid coding those security mechanisms by hand.

Microsoft offered an implementation of the Web service protocol specifications required to make the project a success. By using WSE 2.0, we were able to focus on the solution's business logic instead of writing security code. WS-Policy allowed us to simply install digital certificates, extend prebuilt classes to handle passwords and write a few hundred lines of configuration and policy code that describes how the Web services are to use them. Another big enabler was WS-SecureConversation, which gave us the security that was required without sacrificing performance.

The Pocket PC smart client runs on the Microsoft .NET Compact Framework, for which no WSE 2.0 equivalent is currently available. To overcome this, we developed our own security implementation that works with WS-Security and WS-SecureConversation, which we did by wrapping existing Microsoft Windows Mobile 2003 cryptographic libraries with managed C# code that runs under the .NET Compact Framework on the mobile device. These application components communicate with a software bridge that is part of the service-oriented architecture, which in turn uses WSE 2.0 to communicate with the targeted services. In those targets, Web originates from a WSE-enabled system Discussion.

A unique feature of our EPR system is its Web-based architecture. This architecture makes the system secure, expandable and stable. New data sources - additional patient monitors, for instance - can be easily added to the system. Using Web-based standards, the system shares confidential patient information securely among the various Web services and authorized users. The system is stable; authorized users and other Web services can be added to or removed with no effect on the system as a whole.

As is the nature of any EPR system, the data gathered from various Web services, such as vital signs data from patient monitors, is granular and time stamped. The system is not limited to recording every five minutes as the anesthesiologist does manually during a case. The need for this level of timed granularity becomes apparent when reviewing a particular case to analyze the effects of various procedures and medications. It allows the anesthesiologist to correlate administered medications with vital signs changes. It is also vital in clinical research trials in which the data-gathering process often requires the full attention of one or more research coordinators. It has been thought that one underlining concern of anesthesiologists about EPR systems is that the EPR system inherently collects too much data, particularly vital signs data, which could have a medical-legal backlash. However, Feldman demonstrated in a survey study of anesthesia departments that employed EPR systems that the automatic record did not hinder the defense process in malpractice cases. Moreover, the respondents in his survey study viewed the technology as essential for anesthesia risk management (5).

Another important feature of our EPR system is the concept of the timeline in relation to our database structure. Data are stored as a sequence of timed actions. These actions can be strung together automatically to form commonly used hospital records, such as the intraoperative anesthesia record. The timeline in conjunction with our wireless smart client data access allows the anesthesiologist to remotely view cases and effectively consult with colleagues without being present in the operating room or at the bedside.

Our EPR system has the potential to enhance the important missions of an academic medical institution, such as OSUMC, by wirelessly bringing in authorized experts to the patient's bedside to consult on treatment options. The types of data that our system gathers are invaluable in research practice. The data are essential for educational and review purposes. Finally, the data may provide a quantifiable means to improve the use of a hospital's valued equipment, infrastructural and personnel resources.

Conclusion

We developed a secure, expandable and stable EPR system. The unique feature of the system is its Web-based architecture using Microsoft .NET Web Services utilities that meets the stringent requirements of a modern medical center. If fully implemented, the system has the potential to improve the overall efficiency of patient treatment and hospital operation as well as to enhance medical research and medical education. The system may enhance revenue and improve the use of the scarce and expensive hospital resources. The system architecture allows for easy integration of other hospital subsystems (laboratory and radiology, for instance) with a minimum of programming effort. The system represents a state-of-the-art improvement in EPR technology.

References

1. Quality Interagency Coordination Task Force. Doing What Counts for Patient Safety - Federal Action to Reduce Medical Errors and Their Impact. February 2000.
2. Shortliffe EH. The evolution of electronic medical records. Acad Med 1999 Apr;74(4):414-419.
3. Block FE, Burton LN, Rafal MD, et al. Two computer-based anesthetic monitors: The Duke Automated Monitoring Equipment (DAME) system and the MicroDAME. J Clin Monit 1985;1:30-51.
4. Block FE, Reynolds KM, McDonald JS. The Diatek Arkive "Organizer" patient information management system: Experience at a university hospital. J Clinical Monitoring and Computing 1998;14:89-94.
5. Feldman JM. Do anesthesia information systems increase malpractice exposure? Results of a survey. Anesth Analg 2004;99:840-843.