A Distributed e-Healthcare System Based on the Service Oriented Architecture

Firat Kart, Gengxin Miao, L. E. Moser, P. M. Melliar-Smith
Department of Electrical and Computer Engineering
University of California, Santa Barbara, CA 93106
{fkart, miao, moser, pmms}@ece.ucsb.edu
Web site: http://phoenix.ece.ucsb.edu/ehealthcare

Abstract

Healthgrid Review

Large-scale distributed systems, such as e-healthcare systems, are difficult to develop due to their complex and decentralized nature. The Service Oriented Architecture facilitates the development of such systems by supporting modular design, application integration and interoperation, and software reuse. With open standards, such as XML, SOAP, WSDL and UDDI, the Service Oriented Architecture supports interoperability between services operating on different platforms and between applications implemented in different programming languages. In this paper we describe a distributed e-healthcare system that uses the Service Oriented Architecture as a basis for designing, implementing, deploying, invoking and managing healthcare services. The e-healthcare system that we have developed provides support for physicians, nurses, pharmacists and other healthcare professionals, as well as for patients and medical devices used to monitor patients. Multi-media input and output, with text, images and speech, make the system more user friendly than existing e-healthcare systems.

1 Introduction

Healthcare is a field in which accurate record keeping and communication are critical and yet in which the use of computing and networking technology lags behind other fields. Healthcare professionals and patients are often uncomfortable with computers, and feel that computers are not central to their healthcare mission, even though they agree that accurate record keeping and communication are essential to good healthcare.

In current healthcare, information is conveyed from one healthcare professional to another through paper notes or personal communication. For example, in the United States, electronic communication between physicians and pharmacists is not typically employed but, rather, the physician writes a prescription on paper and gives it to the patient. The patient carries the prescription to the pharmacy, waits in line to give it to a pharmacist, and waits for the pharmacist to fill the prescription. To improve this process, the prescriptions could be communicated electronically from the physician to the pharmacist, and the human computer interfaces for the physicians, nurses, pharmacists and other healthcare professionals could be voice enabled.

According to Carmen Catizone of the National Association of Boards of Pharmacy [15], there are as many as 7,000 deaths from incorrect prescriptions in the United States each year. A Washington Post article [36] indicates that as many as 5% of the 3 billion prescriptions filled each year are incorrect. These numbers indicate that there is an urgent need to reduce the errors in healthcare. The report, To Err is Human: Building a Safer Health System [34], discusses human errors in the workplace:

> "Human beings, in all lines of work, make errors. Errors can be prevented by designing systems that make it hard for people to do the wrong thing and easy for people to do the right thing."

The Healthgrid Review [19] concludes that large healthcare systems have difficulties in managing personal data, standardizing the data, extracting content-based knowledge and federating databases. These problems indicate the need to improve the quality of healthcare systems, ease the access to healthcare and healthcare information, and reduce the cost of delivery of healthcare [35].

In this paper we describe a distributed e-healthcare system that we have developed. The system is intended for use by physicians, nurses, pharmacists and other healthcare professionals, as well as by patients and medical devices used to monitor patients. First we describe our e-healthcare Service Oriented Architecture, and then we describe the design of the Clinic and Pharmacy modules of the system. Next we discuss our implementation of the e-healthcare system, and then we present a simple performance evaluation. Finally, we discuss related work, and then we present conclusions and future work.

2007 IEEE International Conference on Services Computing (SCC 2007)
0-7695-2925-9/07 $25.00 © 2007 **IEEE**

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 17, 2009 at 00:06 from IEEE Xplore. Restrictions apply.

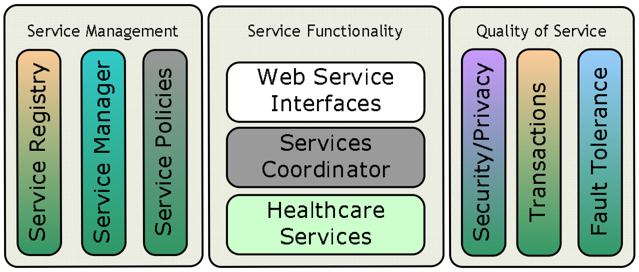

Figure 1. e-Healthcare SOA.

2 e-Healthcare SOA

The e-healthcare system that we have developed is based on the Service Oriented Architecture (SOA) [28] and uses Web Services [37] and Atom/RSS [11].

The Service Oriented Architecture reinforces basic software architecture principles such as abstraction, encapsulation, modularization and software reuse. It provides well-defined interfaces for client applications and separates the interfaces from their implementations. It allows service capabilities and interfaces to be implemented as a collection of processes. Each process itself provides a service, one that offers a particular capability. Because each process is exposed through a standard interface, the underlying implementation of the individual service is free to change without affecting how the service is consumed. The Service Oriented Architecture not only encompasses the services from a technology perspective, but also includes the policies and practices by which the services are provided and consumed.

For these reasons, the Service Oriented Architecture is an appropriate model for developing a distributed e-healthcare system. The Service Oriented Architecture for our e-healthcare system consists of three layers as shown in Figure 1. The top layer provides the Web Services interfaces. The bottom layer contains the healthcare services, which are described in Section 3. The Services Coordinator within the middle layer controls the flow of messages in the system from the Web Services interfaces to the healthcare services and vice versa. The other components of the architecture are application neutral, and are not specific to healthcare.

Security and privacy are particularly important issues for healthcare. Personal health information is confidential, so access to such information must be restricted to authenticated and authorized users. Secure transmission of such information must be complemented with secure storage of the data [4]. The use of the Service Oriented Architecture is critical for enforcing such policies.

Our e-healthcare system is designed with these security and privacy issues in mind. Users of the system are authenticated, and session information is kept with logging of service calls. Resources in the system are attached to the resource creator, and privileged users can view/modify the data in the system. For applications deployed on devices like PDAs, authentication and session management are strictly enforced.

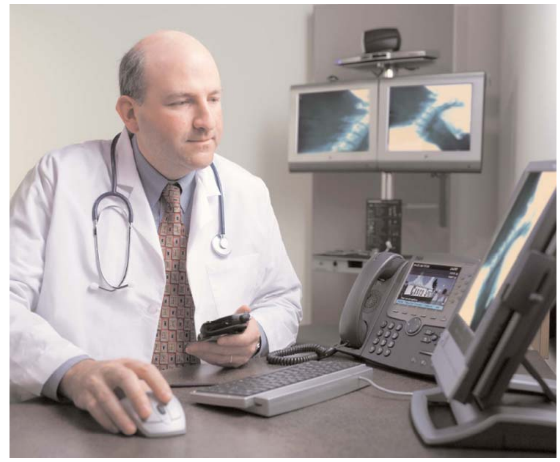

Figure 2. Physician using PDA and PC.

3 Healthcare Services

The healthcare services of our e-healthcare system are provided by a Clinic module and a Pharmacy module. The system also provides user interfaces for patients, physicians, nurses, and pharmacists. The devices accessing these modules can be desktop or server computers and PDAs or smart cell phones, as shown in Figure 2. They can also be medical monitoring devices, such as blood pressure monitors.

3.1 Clinic Module

The Clinic module exposes two interfaces, a Web Server and a Web Service, for the clinic staff, the patients and the medical monitoring devices, as shown in Figure 3. The Web Server interface is intended for users who prefer to use a Web browser to access the healthcare services. The Web Service interface can be used by humans or devices to communicate with the e-healthcare system. The Web Server uses the Web Services to access the data.

The Clinic module provides support for routine activities of the physician. It maintains information, such as the physician's appointments for a specific day/week as shown in Figure 4, the patients that s/he has examined, notes related to the patients, etc. Access to a patient's private information is restricted and secured, as discussed previously. The Clinic module sends prescriptions from the physician to the desired pharmacies over the wide-area Internet using the Web Service provided. It uses the Yahoo! LocalSearch [38] Web Service to locate the pharmacy closest to the patient's home or the physician's office.

2007 IEEE International Conference on Services Computing (SCC 2007)
0-7695-2925-9/07 $25.00 © 2007 IEEE

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 17, 2009 at 00:06 from IEEE Xplore. Restrictions apply.

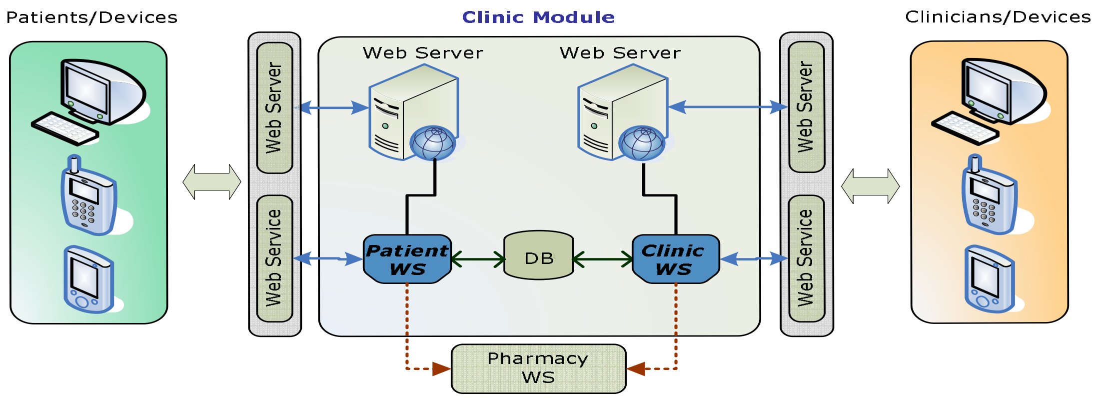

Figure 3. Clinic module.

The physician can use the Web Server interface to access the e-healthcare system using a browser from a desktop computer or a PDA. The physician can use the PDA to enter/retrieve information about the patient during/after an appointment and access this information any time, any where. The use of a PDA with a small keyboard makes it difficult for the physician to input information about the patient. Consequently, in addition to the graphical interface, we have enabled the PDA with speech recognition software that allows the physician to enter/retrieve information by speaking. We also provide feedback to the physician by means of speech synthesis software. These speech technologies ease the task of the physician in completing tasks and encourage the use of the PDA.

Medical monitoring devices deployed with wired or wireless network support can also transmit information to the Clinic Web Service. Our blood pressure monitoring device is bluetooth-enabled, which allows information to be transmitted from the patient to the e-healthcare system for examination by the physician.

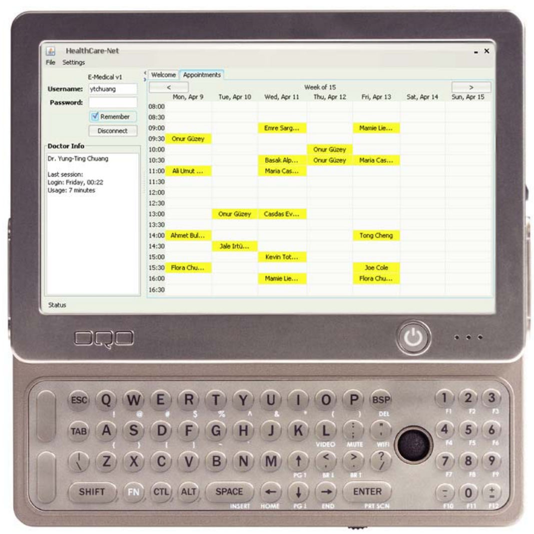

Figure 4. Graphical interface showing a weekly calendar on physician's OQO PDA.

3.2 Pharmacy Module

The Pharmacy module exposes Web Server and Web Service interfaces, as shown in Figure 5. The Web Server interface allows the users to access the e-healthcare system at the pharmacy using a browser. The Web Service interface provides access for applications deployed at the pharmacy and can also be used by humans and devices.

The Pharmacy module provides services to the pharmacist, patients and devices used at the pharmacy. The Pharmacy module keeps a record of the patient's prescriptions for the pharmacist's and the patient's reference.

When the physician submits a new prescription to the pharmacy, the Clinic module at the physician's office communicates with the Pharmacy module at the pharmacy over the wide-area Internet. Removing human intervention from the communication between the physician and pharmacist, and maintaining the information electronically, reduces the possibility of human errors.

The pharmacist can view the outstanding prescriptions for the patients, as they are received from the physicians. The Pharmacy module updates the status of the prescriptions as the pharmacist fills them. The patient can determine, via the Web Server or Web Service, whether a prescription has been filled and is ready for pick up or delivery.

According to the National Community Pharmacists Association, a pharmacy in the United States dispenses an average of 204 prescriptions per day [1]. Most of these prescriptions are renewals of existing prescriptions. Therefore, the patient interface also has access to services that provide renewal of existing prescriptions, custom alerts, etc.

2007 IEEE International Conference on Services Computing (SCC 2007)
0-7695-2925-9/07 $25.00 © 2007 IEEE

IEEE COMPUTER SOCIETY

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 17, 2009 at 00:06 from IEEE Xplore. Restrictions apply.

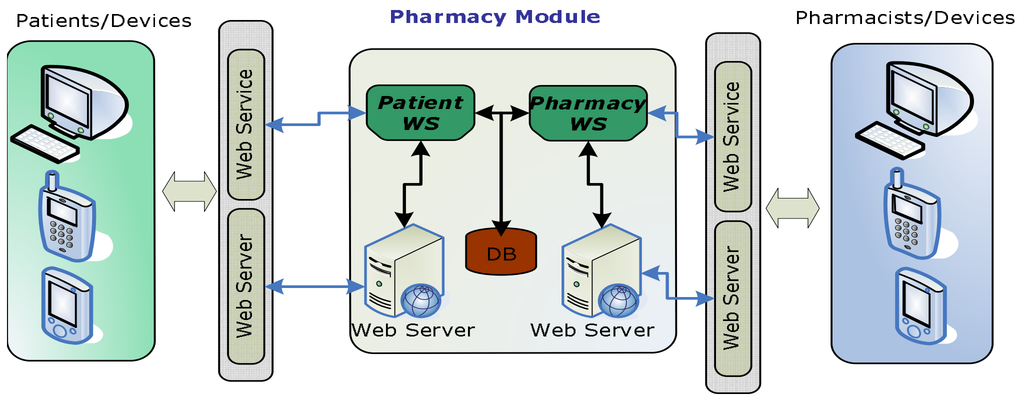

Figure 5. Pharmacy module.

4 Implementation

In our implemention we deployed the Web Services on 3 GHz computers with 2 GB memory. We deployed the client applications on 2 GHz computers with 2 GB memory. The client applications communicate with the Web Server or Web Services over the wide-area Internet.

For the PDA we used the OQO device, shown in Figure 4, which is a full-featured 3" x 5" personal computer. The OQO device is powered by a 1 GHz Transmeta Crusoe processor, which is powerful enough to run both an embedded speech recognition engine and our physician application. The device features a 800 x 480 resolution screen that is capable of providing the user with detailed graphical information. The PDA can communicate with a desktop or server computer using wireless (WiFi) communication.

In the subsections below we discuss the software technologies that we employed in our e-healthcare system and the considerations that we used in making these choices.

4.1 Web Services

Although it is possible to use any programming or scripting language to implement a Service Oriented Architecture or a distributed e-healthcare system, we decided to use the Java™ language for its ability to be deployed on small wireless devices as well as on powerful server computers.

Our Web Services use the Apache Axis2 Framework, which is the core engine for Web Services built on Apache Axiom, and the Apache Tomcat server [5]. We decided to use Axis2 for the reasons that it provides:

- XML client API including WSDL and policy support
- Support for any message exchange pattern
- Synchronous and asynchronous function calls
- WS-Policy driven code generation
- Flexible service lifecycle model
- Support for SOAP, WSDL, WS-Reliable Messaging, WS-Security, WS-Addressing and SAAJ.

In addition, Axis2 provides data binding that enables application developers to generate SOAP messages without having to worry about constructing or parsing them. Example SOAP messages for our e-healthcare system are shown in Figure 6 (a).

To ease the development and debugging of our e-healthcare system, we used Plain Old Java Objects (POJO), based on the Spring Framework [26].

4.2 Speech Software

In our e-healthcare system we use SRI's DynaSpeak speech recognition engine [27]. DynaSpeak supports multiple languages, adapts to different accents, and does not require training prior to use. It incorporates a Hidden Markov Model (HMM) for separating speech from interfering signals with different statistical characteristics. DynaSpeak is ideal for embedded platforms, because of its small footprint (less than 2 MB of memory) and its low computing requirements (66 MHz Intel x86 or 200 MHz Strong Arm processor). DynaSpeak can be used with either a finite-state grammar or a free-form grammar. We chose to use a finite-state grammar, because it offers greater control over parsed phrases than the free-form grammar. The prescription grammar for our e-healthcare system is shown in Figure 7, and an example usage scenario is shown in Figure 8.

We also used AT&T's Natural Voices speech synthesis engine [6]. Natural Voices provides a simple and efficient way to produce natural sounding device-to-human voice interaction. It can accurately and naturally pronounce words, and speak in sentences that are clear and easy-to-understand, without the feeling that a computer is talking to the human. Natural Voices supports many languages, male and female voices, and the SAPI, VoiceXML and JSAPI interface standards. Using Natural Voices, we created text-to-speech software for the prototype device, that runs in the background and accepts messages in VoiceXML format.

2007 IEEE International Conference on Services Computing (SCC 2007)
0-7695-2925-9/07 $25.00 © 2007 IEEE

IEEE COMPUTER SOCIETY

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 17, 2009 at 00:06 from IEEE Xplore. Restrictions apply.

```
<!-- Request to get patient information -->
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope
xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
<soapenv:Header/><soapenv:Body>
  <ns2:getPatient xmlns:ns2="http://ws.doctor/xsd">
     <ns2:param0>fkart</ns2:param0>
  </ns2:getPatient>
</soapenv:Body></soapenv:Envelope>

<!-- Response to getPatient operation -->
<?xml version='1.0' encoding='UTF-8'?>
<soapenv:Envelope
xmlns:soapenv="http://www.w3.org/2003/05/soap-envelope">
<soapenv:Header /> <soapenv:Body>
  <ns:getPatientResponse xmlns:ns="http://ws.doctor/xsd"><ns:return>
    <firstname xmlns="http://data.doctor/xsd">Firat</firstname>
    <lastname xmlns="http://data.doctor/xsd">Kart</lastname>
    <ssn xmlns="http://data.doctor/xsd">111-11-1111</ssn>
    <birthdate xmlns="http://data.doctor/xsd">1981-03-30</birthdate>
    <address xmlns="http://data.doctor/xsd">
       <line1>6616 Abrego Road</line1> <city>Goleta</city>
       <state>CA</state><zipCode>93117</zipCode>
    </address>
  </ns:return> </ns:getPatientResponse></soapenv:Body>
</soapenv:Envelope>
```

(a) SOAP request/response messages
to retrieve patient information

```
<?xml version="1.0" encoding="UTF-8"?>
<feed xmlns="http://www.w3.org/2005/Atom"
     xmlns:dc="http://purl.org/dc/elements/1.1/" >
 <title>PDA Data</title>
 <id>http://phoenix.ece.ucsb.edu/PDAInfo.xml</id>
 <entry>
   <title>PDA Data</title>
   <link rel="alternate" href="http://phoenix.ece.ucsb.edu" />
   <author><name>Firat Kart</name></author>
   <id>tag:phoenix.ece.ucsb.edu,2007:note-1</id>
   <updated>2007-04-06T01:38:32Z</updated>
   <published>2007-04-06T01:38:32Z</published>
   <content type="xhtml">
     <CDR xmlns="" Database="doctorws" Table="note" ID="1"
     UID="cd86e54c-a1cc-497e-9967-29dd28c0bdc7" Type="INSERT"
     Created="1175822908000">
       <AppointmentID>7</AppointmentID>
       <ID>3</ID>
       <Type>Height</Type>
       <CreatedOn>2007-04-05 18:28:28</CreatedOn>
       <Content>5'5</Content>
     </CDR>
   </content>
   <summary type="text">INSERT operation on table note</summary>
   <dc:creator>Firat Kart</dc:creator>
   <dc:date>2007-04-06T01:38:32Z</dc:date>
</entry></feed>
```

(b) Atom feed to synchronize information on PDA
and desktop/server computer

Figure 6. (a) Example SOAP messages, and (b) example Atom feed.

```
# Physician Prescription Grammar
<Start1>        = start prescription;
<Start2>        = {prescription for <Patient>
     | patient is <Patient>} <Confirmation>];
<Pharm>         = pharmacy is <Pharmacy> [<Confirmation>];
<Med>      = {[first | next] {medication
     | prescription} is <Medication> | remove <Medicationinlist>
     | change <Medicationinlist> | cancel} [<Confirmation>];
<Dosage>        = {dose is <Quantity> | number <Number>
     | per day <Number> | <Number> per day
     | with food | <Timeofday>} [<Confirmation>];
<Show>          = show [all | whole | complete]  [prescription[s]];
<End>           = {issue|cancel} [prescription];
<Confirmation>    = yes | no | OK | right | cancel;
<Timeofday>       = with meals | before breakfast
     | with breakfast | with lunch | with dinner
     | after dinner | at bed time;
# <Patient>, generated by Patient Contact Application
# <Pharmacy>, generated by Pharmacy Contact Application
# <Medication>, generated by Formulary Application
# <Medicationinlist>,generated dynamically by this Application
# Primary Prescription Grammar
[<Start1><Start2> | <Start2>]
{{<Pharm> {{<Med> [<Dosage>*]}*<Show>}*} |
{{{<Med> [<Dosage>*]}* <Show>}* <Pharm>}} <End>;
```

Figure 7. Prescription grammar.

4.3 Atom/RSS

Enabling physicians, nurses and pharmacists to use PDAs or smart cell phones requires those devices to be able to communicate with the Clinic or Pharmacy modules on their desktop or server computers using a wireless or wired network. It is possible that at certain times or places no network communication is available. For example, in an emergency, such as an earthquake, a physician cannot expect to have a network connection to his/her desktop or server computer. Physicians, nurses and pharmacists must be able to access, and modify, healthcare information offline.

Atom/RSS [11] are syndication technologies, based on XML, that enable the sharing and communication of information between heterogeneous platforms by making the information self-describing. They allow a publisher to make information available to consumers on the Web, which retrieve that information subsequently. The information is delivered from the publisher to the consumer as an XML file, called an Atom/RSS feed.

We have developed a Consistent Data Replication (CDR) and Reliable Data Distribution (RDD) infrastructure [20] that replicates information from one computer to another using Atom/RSS feeds. We can use this infrastructure to synchronize information on the physician's desktop or server computer with that on his/her PDA, as shown in Figure 9, allowing the physician to view that information when it is offline. At the start of the day, our software on the PDA retrieves the necessary updates from the Clinic Web Service on the desktop or server computer via a wired or wireless network. Any modifications to the information on the PDA are stored locally on the PDA. At the end of the day, our software on the PDA generates an update feed for the Clinic Web Service on the desktop or server computer to read. An example Atom feed for our e-healthcare system is shown in Figure 6 (b).

2007 IEEE International Conference on Services Computing (SCC 2007)
0-7695-2925-9/07 $25.00 © 2007 IEEE

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 17, 2009 at 00:06 from IEEE Xplore. Restrictions apply.

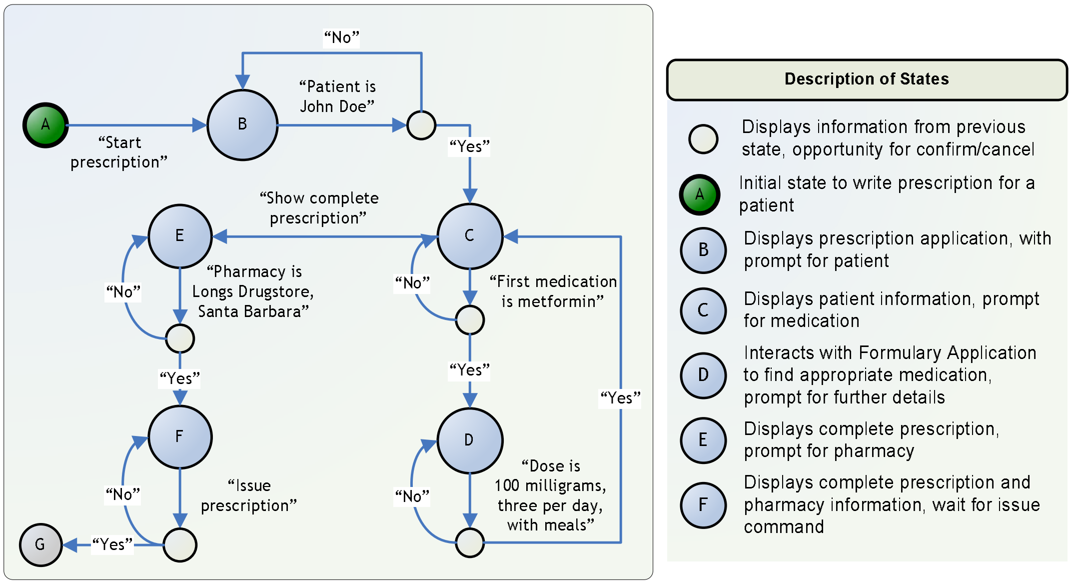

Figure 8. Example usage scenario of prescription grammar.

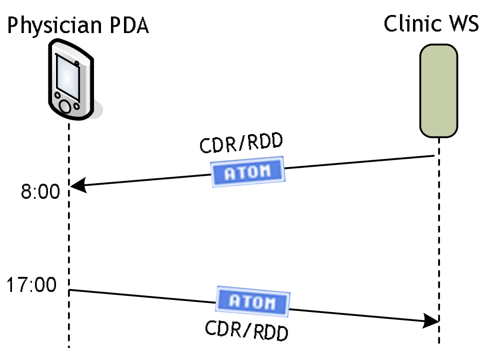

Figure 9. Synchronization of information on physician PDA and desktop/server computer.

5 Performance Evaluation

For the performance evaluation we measured the latency of the Pharmacy Web Service, *i.e.*, the delay between the physician's sending a prescription to the pharmacist and the physician application receiving an acknowledgement that the pharmacy application received the prescription.

In the experiments we used different rates of sending prescriptions (0.1, 0.2, 0.5, 1, 2 and 3 prescriptions per second) and different numbers of medicines in a prescription (1, 2, 3, 4 and 5 medicines).

Figure 10 shows the experimental results, namely, the mean latency for a prescription containing a certain number of medicines, taken over 300 seconds. As expected, increasing the rate of sending prescriptions increases the latency. The increased latency is due to the increased usage of the Pharmacy Web Service. Likewise, for a particular rate of sending prescriptions, increasing the number of medicines in a prescription increases the latency, because the message size is larger and the processing time for the XML is greater.

6 Related Work

Extensive work has been undertaken on the development of e-healthcare systems. Beyer, *et al.* [8] discuss the limitations and challenges of developing a flexible, process-oriented architecture for an integrated healthcare network. As key issues, they identify flexibility, adaptability, robustness, integration of existing systems and standards, semantic compatibility, security and process orientation. Song, *et al.* [25] provide a survey of the topic of computer-aided healthcare workflow. They define workflow properties for common healthcare practices and present a summary of workflow properties and requirements. Ardissono, *et al.* [2] discusses implementing healthcare workflow as a BPEL process, which we plan to investigate further.

Much of the work on e-healthcare systems has focused on record keeping and databases [10, 30, 31, 33]. Work has also focused on access and security [4], as well as on social implications of recording and communicating healthcare information [9]. Less work has been done on human computer interfaces and usability by healthcare professionals and patients. Our e-healthcare system aims to reduce human errors by exploiting electronic communication and record keeping, and by providing user friendly input and output capabilities.

2007 IEEE International Conference on Services Computing (SCC 2007)
0-7695-2925-9/07 $25.00 © 2007 **IEEE**

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 17, 2009 at 00:06 from IEEE Xplore. Restrictions apply.

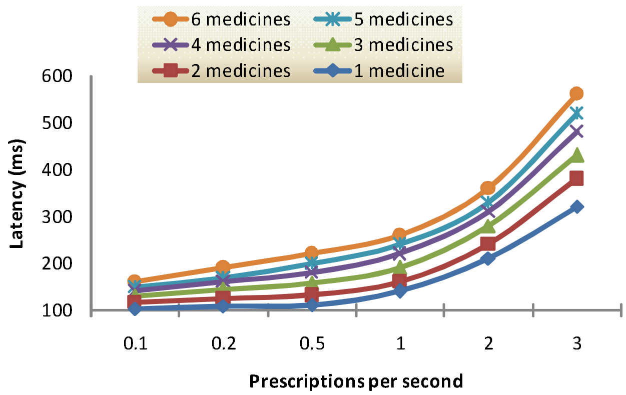

Figure 10. Latency of Pharmacy Web Service.

Governmental and private organizations, such as the U.S. Department of Health & Human Services [32] and the California HealthCare Foundation [13], have promoted the use of electronic technology for healthcare, but these organizations incline towards highly centralized or centrally administered systems [3, 16]. However, the fragmented nature of healthcare in the United States, and the increasingly international nature of healthcare services and patients, mandates more distributed and interoperable solutions based on open international standards [18].

Commercial enterprises, such as Aurora Healthcare Systems [7], Medseek [21] and Palm [24], provide e-healthcare Web portals, systems and handsets, but none of them offers all of the functionality that our e-healthcare system provides. We use modern technology to expose the functionality of our e-healthcare system as Web Services based on the Service Oriented Architecture, so that both humans and applications can use the services provided.

Other university researchers have developed e-healthcare systems based on the Service Oriented Architecture. In particular, Omar and Taleb-Bendiab [23] have used the Service Oriented Architecture, in conjunction with grid computing technology, for a sensor and actuator framework that monitors the health status of a patient and supplies feedback. Our e-healthcare system provides services that involve patients, physicians, nurses and pharmacists as well as medical monitoring devices, whereas their framework focuses specifically on the use of medical monitoring devices.

Taylor, *et al.* [30] describe a Service Oriented Architecture for a health research data network that provides control over access and use of information records. Care2x [14] is an open-source, Web-based university project that implements a modern hospital information system, for training health care engineering students and medical students. It includes a central data server and a health exchange protocol, and is implemented using the Apache Web server, the PHP scripting language and the mySQL database system. Our e-healthcare system focuses on the interactions between patients, physicians, nurses, pharmacists and medical monitoring devices outside the hospital setting, rather than on a health research data network or health care training for the hospital environment.

Related to our use of Atom/RSS, Subramanian, *et al.* [29] have developed a model for patient-centered healthcare services using a mobile device to push/pull data to/from a data analysis engine based on the Service Oriented Architecture. Budgen, *et al.* [12] have developed a data integration broker for healthcare systems, based on the software service model, that collects and integrates data from autonomous healthcare agencies. Our Consistent Data Replication and Reliable Data Distribution infrastructure can also be used for that purpose. An interesting open research topic is how to solve the semantic interoperability problems that arise with the use of heterogeneous medical information systems that employ different standards to represent the same information.

7 Conclusions and Future Work

In this paper we have presented a distributed e-healthcare system that uses the Service Oriented Architecture as a basis for designing, implementing, deploying, managing and invoking e-healthcare services. Healthcare requires modern solutions, designed and implemented with modern technologies, that encourage healthcare professionals and patients to adopt new procedures that can improve the presentation and delivery of healthcare. Multimedia input and output, particularly graphics and speech, makes the system seem less computer-like and more attractive to users who are not computer-oriented.

Our e-healthcare system currently focuses on the relationships between patients, physicians, nurses and pharmacists. We plan to extend the system to other healthcare facilities and professionals, such as laboratory technicians who perform and report tests and analyses requested by physicians. We also plan to investigate whether our Clinic and Pharmacy modules can be interfaced to applications supplied by pharmaceutical companies that provide information on medications and dosages and warn of interactions between medications, such as Epocrates Rx [17]. In addition, we plan to investigate drug delivery devices, such as e-pillboxes [22], that prompt and monitor the regular and timely consumption of medications.

We are currently talking with vendors of such healthcare devices about the benefits of providing open interfaces for incorporating their devices into an e-healthcare system based on the Service Oriented Architecture. We are also working with healthcare professionals to ensure that our e-healthcare system meets their needs, and we are improving our system based on their feedback.

0-7695-2925-9/07 $25.00 © 2007 **IEEE**
IEEE COMPUTER SOCIETY

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 17, 2009 at 00:06 from IEEE Xplore. Restrictions apply.

References

[1] V. Alexandria, "Number of independent pharmacies on the rise, Impact of Medicare Part D looms on the horizon," April 28, 2006, http://www.ncpanet.org/media/releases/2006/number_of_independent_pharmacies_on_the_04-28-2006.php

[2] L. Ardissono, A. Di Leva, G. Petrone, M. Segnan and M. Sonnessa, "Adaptive medical workflow management for a context-dependent home healthcare assistance service," *Electronic Notes in Theoretical Computer Science*, Elsevier, 2005, http://www.di.unito.it/

[3] R. Andersen, T. H. Rice and G. F. Kominski, *Changing the U.S. Health Care System: Key Issues in Health Services, Policy and Management*, Josswey-Bass, 2001.

[4] R. J. Anderson, *Security in Clinical Information Systems*, British Medical Association, 1996.

[5] Apache, Axiom and Axis2, http://ws.apache.org/

[6] AT&T, Natural Voices, http://www.research.att.com/ttsweb/tts/

[7] Aurora Health Care, http://www.aurorahealthcare.org

[8] M. Beyer, K. A. Kuhn, C. Meiler, S. Jablonski and R. Lenz, "Towards a flexible, process-oriented IT architecture for an integrated healthcare network," *Proceedings of the 2004 ACM Symposium on Applied Computing*, Nicosia, Cyprus, March 2004, 264-271.

[9] B. P. Bloomfield, "The role of information systems in the UK National Health Service: Action at a distance and the fetish of calculation," *Social Studies of Science* 21, 1991, 701-734.

[10] M. K. Bourke, *Strategy and Architecture of Health Care Information Systems*, Springer, 1994.

[11] T. Bray, "RSS 2.0 and Atom 1.0 compared," July 2005, http://www.tbray.org/atom/RSS-and-Atom

[12] D. Budgen, M. Rigby, P. Brereton and M. Turner, "A data integration broker for healthcare systems," *IEEE Computer* 40, 4, April 2007, 34-41.

[13] California HealthCare Foundation, "E-prescribing," November 2001, www.chcf.org

[14] Care2x, http://www.care2x.org

[15] C. Catizone, Colorado State Board of Pharmacy News, http://www.nabp.net

[16] D. E. Detmer, "Building the national health information infrastructure for personal health, health care services, public health and research," *BMC Medical Informatics and Decision Making* 3, 2003, 1-40.

[17] Epocrates, Epocrates Rx, http://www.epocrates.com

[18] J. Grimson, W. Grimson and W. Hasselbring, "The SI challenge in health care," *Communications of the ACM* 43, 6, June 2000, 48-55.

[19] HealthGrid Association & Cisco Systems, Healthgrid whitepaper, http://whitepaper.healthgrid.org/

[20] F. Kart, L. E. Moser and P. M. Melliar-Smith, "Reliable data distribution and consistent data replication using the Atom syndication technology," *Proceedings of the WORLDCOMP International Conference on Internet Computing*, Las Vegas, NV, June 2007.

[21] Medseek, http://www.medseek.com

[22] MedSignals, http://www.medsignals.com

[23] W. M. Omar and A. Taleb-Bendiab, "Service oriented architecture for e-health support services based on grid computing," *Proceedings of the IEEE International Conference on Services Oriented Computing*, Chicago, IL, September 2006, 135-142.

[24] Palm, "ePrescribing," http://www.palm.com

[25] X. Song, B. Hwong, G. Matos, A. Rudorfer, C. Nelson, M. Han and A. Girenkov, "Understanding requirements for computer-aided healthcare workflows: Experiences and challenges," *Proceedings of the 28th International Conference on Software Engineering*, Shanghai, China, May 2006, 930-934.

[26] Spring Framework, http://www.springframework.org/

[27] SRI, DynaSpeak, http://www.sri.com/

[28] L. Srinivasan and J. Treadwell, "An overview of Service Oriented Architecture, Web Services and Grid Computing," http://h71028.www7.hp.com/ERC/downloads/SOA-Grid-HP-WhitePaper.pdf

[29] M. Subramanian, A.S. Ali, O. Rana, A. Hardisty and E. C. Conley, "Healthcare@Home: Research models for patient-centered healthcare services," *Proceedings of the 2006 International Symposium on Modern Computing*, October 2006, 107-113.

[30] K. L. Taylor, C. M. Colton, R. Baxter, R. Sparks, U. Srinivasen, M. A. Cameron and L. Lefort, "A Service Oriented Architecture for a health research data network," *Proceedings of the International Conference on Scientific and Statistical Database Management*, Santorini, Greece, June 2004, 443-444.

[31] M. Tsiknakis, C. E. Chronaki, S. Kapidakis, C. Nikolaou and S. C. Orphanoudakis, "An integrated architecture for the provision of health telematic services based on digital library technologies," *International Journal of Digital Libraries* 1, 3, December 1997, 257-277.

[32] U.S. Department of Health & Human Services, "Electronic prescribing," http://healthit.ahrq.gov

[33] U.S. Institute of Medicine, R. S. Dick, E. B. Steen and D. E. Detmer (eds.), *The Computer-Based Patient Record, an Essential Technology for Health Care*, 1997, National Academy Press.

[34] U.S. Institute of Medicine, L. T. Kohn, J. M. Corrigan and M. Donaldson (eds.), *To Err is Human: Building a Safer Health System*, 2000, National Academy Press.

[35] U.S. Institute of Medicine, *Crossing the Quality Chasm: A New Health System for the 21st Century*, 2000, National Academy Press.

[36] R. Weiss, "Medical errors blamed for many deaths: As many as 98,000 a year in U.S. linked to mistakes," *The Washington Post*, November 30, 1999, A01.

[37] World Wide Web Consortium, "Web Services Architecture," 2004, http://www.w3.org/TR/ws-arch

[38] Yahoo!, "LocalSearch APIs," http://developer.yahoo.com/search/local/

2007 IEEE International Conference on Services Computing (SCC 2007)
0-7695-2925-9/07 $25.00 © 2007 **IEEE**

IEEE COMPUTER SOCIETY

Authorized licensed use limited to: IEEE Xplore. Downloaded on January 17, 2009 at 00:06 from IEEE Xplore. Restrictions apply.