ELMR: Lightweight Mobile Health Records

Arvind Kumar, Amey Purandare, Jay Chen, Arthur Meacham, Lakshminarayanan Subramanian
Courant Institute of Mathematical Sciences
New York University
{arvind.kumar, amey, jchen, meacham, lakshmi}@cs.nyu.edu

ABSTRACT

Cell phones are increasingly being used as common clients for a wide suite of distributed, database-centric healthcare applications in developing regions. This is particularly true for rural developing regions where the bulk of the healthcare is handled by health workers due to lack of doctors; the widespread availability of cellular services have made mobile devices as an important computing platform for enabling healthcare applications for these health workers. Unfortunately, the current SQL model for distributed client/server systems is far too heavy-weight for these applications, particularly in light of the high communications cost and extremely limited data transmission capacity available in these environments.

In this demonstration, we describe the Efficient Lightweight Mobile Records (ELMR) system that provides a practical and lightweight database access protocol for accessing and updating records remotely from mobile devices under an extremely bandwidth and cost-constrained Short Messaging Service (SMS) channel comprising of 140 byte packets. We have implemented ELMR using the RMS functionality in J2ME, and integrated it into an HIV treatment application we are developing for use by African health workers.

Categories and Subject Descriptors

C.2.4 [Distributed Systems] Distributed Applications

General Terms

Design, Economics, Reliability, Experimentation, Security, Human Factors.

Keywords

Cell phones, user interface, rural healthcare

1. INTRODUCTION

Several rural regions around the world, especially in under-developed areas, do not have access to basic healthcare services and much of the burden of healthcare delivery falls on local health workers who have limited skills and expertise. Recently, cell phones have demonstrated their enormous potential for enhancing rural healthcare by acting as a low-cost computing platform for distributed applications. Several recent research and developmental efforts [1,3,4,5] around the world have explored the use of cell phones as a potential tool for improving rural healthcare. Existing mobile telemedicine efforts such as OpenRosa [4], OpenMRS [3], Voxiva [5] and Partners in Health [8] have made significant advances in this space. However, all of the existing systems are not directly scalable and sustainable in developing country settings due to a variety of factors. First, all existing approaches rely on high-end smart phones or PDAs which may not be economically viable for health workers and patients. Second, these systems rely on standard database implementations on standard TCP/IP networking stacks. These applications rely on GPRS network connectivity which is available and affordable only in urban settings; in most rural regions only voice and Short Messaging Service (SMS) capabilities (140 byte packets) are available. Third, existing implementations may be heavyweight for low-end cell phones.

On the opposite end of the spectrum, systems such as FrontlineSMS [6] offer communications platforms which are affordable and highly available, but lack the complete functionality and consistency semantics required for database record systems. To address these limitations, we have designed and implemented Efficient Lightweight Mobile Records (ELMR), a generic record system that enables health workers and patients to access and update health records in extremely bandwidth constrained 140 byte SMS channel. In this demonstration, we show how ELMR can be applied in the context of an HIV application we have developed for use by African health workers.

2. APPLICATION SCENARIO

Our research group directly works with several large hospitals and AIDS care centers in Ghana and South Africa and our application scenario is motivated by the healthcare delivery setting in these regions. The system has three types of users: a) Health Workers, b) Patients and c) Doctors. Patients primarily reside in rural areas and nearly 30-75% of rural patients possess cell phones as we determined in a recent user study of AIDS patients in rural Ghana. Health workers and doctors may be either stationed in specific locations or conduct mobile health camps across different rural areas on a periodic basis. A given patient may be under the surveillance of one primary doctor/health worker but will regularly be diagnosed and tested by other doctors and health workers.

The usage scenario consists of a doctor/health worker diagnosing a patient in a mobile camp or a rural care center. Under this situation, the doctor uses a mobile phone to fetch and update the patient's medical records using SMS as the primary transport mechanism. In most rural areas, GPRS connectivity is scarce and most health workers use low-end cell phones which have basic Java programmability embedded in them. Any communication should either be voice-based or SMS-based; we choose the latter since the former requires speech recognition tools which is a function of the local language. We describe three different scenarios that occur on the field:

Scenario 1: A health worker can conduct a CD4 count mobile camp in a village where the health worker measures the CD4 count of each patient and updates the health record of each patient. The health worker or a remote doctor may be able to fetch the CD4 history of a patient to track health progress.

Scenario 2: Patients can use cell phones to report their medication consumption along with the corresponding tag information. In the SmartTrack [2] project, we tag each drug with a unique code (barcode or RFID) and can use the cell phone to track the flow of drugs and can serve as a low-cost mechanism for bringing better accountability into the medicine distribution system.

Scenario 3: A health worker diagnoses a patient with a collection of symptoms and is unsure of the diagnosis. The doctor may use the system to search for health records of other patients who have observed the same set of symptoms.

3. SYSTEM DESIGN

We discuss the key points in the design of ELMR that enable the system to fetch and update several records in a single SMS message. ELMR is client-driven where the client specifies search query using a single SMS message and obtains a response in one or more SMS messages. This problem is similar in spirit to the design of TinyDB [7] for sensor networks except the primary constraint is cost of bandwidth rather than efficient usage of battery power.

Restricted Set of Operations: ELMR's motivating problem is medical record systems, and it targets developing regions where the only data connectivity is via SMS. While there are innumerable possible SQL queries that may be issued in order to access the database in the traditional SQL model, if we look at domain-specific operations required for medical record systems (e.g. **create, fetch, update, and search**), only a few specific queries are useful, and we design ELMR around these. This allows the client application to be simple and lightweight. We support only the following operations: create, append, update, destroy, search, fetch, and aggregate fetch. In a given command message, the first 3 bits contain the operation ID and the subsequent 8 bits contain the schema ID. Following these are various bits reserved for control information and the payload containing the variable-length data sent to perform the operation specified by the operation ID.

Semantic Compression: ELMR employs semantic compression which reduces the number of messages exchanged between the client and server to complete an operation by reducing payload space required for each command's requisite data. Each schema, each field of each schema, and each operation is assigned an identifier using a small number of bits, concatenated appropriately to form a small fixed-width record. To better utilize the space in each SMS message we limit the types of fields in a schema. These simple data types encompass the fields typical in medical forms:

1. Date
2. Integer
3. String
4. Boolean
5. Multiple Choice

We optimize by automatically assigning the minimum number of bits to each datatype (e.g. 1 bit to booleans) during schema encoding and decoding. In addition, each integer and date field carries a precision modifier and each multiple choice contains a finite number of options, which aids in correctly assigning the minimum number of bits to make the most of the 140-byte SMS payload. Variable-length strings have a 1-byte length value prepended to them.

Lightweight SMS Reliability: SMS messages can be lost which can affect application semantics. ELMR separates each client interaction with the server into sessions. Each session is associated with a set of requests and a set of responses bounded by a maximum of 16 SMS messages in either direction. ELMR provides reliability guarantees within a particular session by using bulk acknowledgements which are piggybacked in the response SMS messages. Bounding the number of messages in a session is important since ELMR can use a bit-vector based cumulative acknowledgement vector within the response and this simplifies reliability semantics. Unlike TCP, ELMR explicitly does not perfect reliability to prevent continuous retransmissions of lost messages (primarily to reduce the cost of expensive SMS messages). Instead, ELMR restricts the number of retransmissions to one; if even the retransmitted message is lost, ELMR reports the error to the user as "Unable to complete request".

User Driven Consistency: Consistency requirements vary depending on the environment and application of the system. In scenarios where explicit consistency guarantees are required, timestamps and conflict resolution mechanisms can easily be incorporated into our design to resolve conflicting updates in favor of the latest update. In addition, inconsistencies can either be handled by the system administrators or the users themselves. However, in other scenarios where perfect consistency is not required ELMR allows reduced consistency at a lower cost. This is useful for a data collection system where only a single CHW has ownership over certain records and immediate updates are not necessary. To facilitate these common types of intermittency requirements we have designed ELMR to operate with both immediate and lazy acknowledgements.

Immediate Acknowledgements:

In an immediate acknowledgement model the server sends an acknowledgement immediately after each session initiated by the client. The client also sends an acknowledgement back to the server in order to notify the server that the client has received the response from the server.

Lazy Acknowledgements:

In the lazy acknowledgement model the client does not expect an immediate acknowledgement from the server. Instead, the client caches transaction for which no acknowledgements are received and marks them as not committed, or 'dirty'. The server is then responsible to acknowledge these transactions in aggregate during future interactions with the client. The client deletes these cached transactions only when it receives an acknowledgment from the server.

Lightweight Privacy: Privacy and secrecy are of paramount importance in any system dealing with medical records. ELMR allows any client to send a request, but the server decides whether a request can be executed or not. In order to restrict access to the database, in ELMR, we implement security based on symmetric key stream encryption. The server maintains a list of phone numbers, their access types, and their secret keys. Each client is given a secret key at the time when client application is deployed on client's cell phone. Periodically, the client generates an initialization vector meant to create an updated version of its symmetric key. The periodicity of this behavior is tunable based on cost/privacy constraints as the two additional messages do incur some economic cost. Additionally, a 16-bit CRC is added to each message. Together, these prevent replay attacks and substitution attacks.

ELMR implements simple privacy and authentication models. The privacy model of ELMR addresses two basic questions: first, who can access the parts of the database and second, to what extent can an authorized user access each record? The authentication model ensures that the sender is actually who he says he is. In ELMR these requirements are satisfied by the server maintaining in a table the authorized users and their permissions. Upon receiving the properly decryptable message, the server verifies that the client under consideration has sufficient privileges to perform the operation which is being requested. If the client does not possess these privileges, the server aborts the request and returns an access violation message back to the client.

Aggregation of Queries and Cached Updates:

ELMR is designed to be a cost-effective solution. Therefore, minimizing the number of SMSs exchanged is essential. There are a number of simple optimizations that are incorporated into ELMR that are helpful and implemented to various degrees. We have not yet evaluated these in detail as they are minor compared to the other aspects of our system design.

Batched Updates:

In ELMR the client caches updates and appends transactions in its local memory and sends them to the server only when client has sufficient data to construct a nearly complete SMS. This greatly reduces the number of SMSs transferred.

Lazy Updates:

We optionally queue updates as above, and send them opportunistically appended to real-time fetch requests, in order to save payload space.

Further Optimizations:

Depending on the consistency level required for the application, a further optimization could be to send the delta between the cached version and the updated version of the record. In some cases this would be smaller than simply sending the entire set of operations.

4. DEMONSTRATION

We have implemented ELMR in J2ME using the Nokia Series 40 Version 3 SDK. For this demonstration we setup a small demonstration database with pre-filled entries along with the relevant schemas for several useful operations. The user is given a mobile phone with ELMR installed to access the remote database and acts as a CHW or doctor gathering medical data or retrieving records in the field. The forms have been specified and pre-loaded into ELMR and derived from paper forms for data typically collected by health workers in the field.

As the health care agent, the user is able to perform simple fetch operations using ELMR. All communications are performed across an SMS channel whose loss rates may be parameterized as a part of the demonstration. Several forms are available to the user and easily accessible via a set of simple user interface forms. Figure 1 shows the user interface while doing a fetch operation. In addition to simple fetch operations, the user is able to fill in forms to create patients and add them to the database or update records for patients already in the system. Finally, the user can also perform an aggregate fetch operation where a collection of patient records are returned by the database server.

Figure 1 User interface of ELMR during fetch operation

The demonstration will allow the user to select either of the consistency models and aggregation models we have described in this paper as a part of the applicaiton settings. We demonstrate that ELMR is cost-effective, capable of providing complete functionality despite the low-cost mobile phone device, and appropriate consistency despite the unreliable communication medium.

5. References

[1] Brian DeRenzi, et al., e-IMCI: Improving Pediatric Health Care in Low-Income Countries, ACM Conference on Computer-Human Interaction (CHI), April 7-10, 2008, Florence, Italy

[2] L. Subramanian et al. The Case for SmartTrack. NYU Technical Report, 2008

[3] "OpenMRS," http://www.openmrs.org/

[4] "Openrosa" http://www.openrosa.org/

[5] "Voxiva" http://mobileactive.org/directory/vendors/voxiva

[6] "FrontlineSMS," http://www.frontlinesms.com/

[7] Samuel Madden et al. "TinyDB: An Acquisitional Query Processing System for Sensor Networks" http://tinydb.org/

[8] "Partners in Health," http://www.pih.org/