

# ELMER: Lightweight Mobile Health Records

Arvind Kumar, Amey Purandare, Arthur Meacham, Lakshminarayanan Subramanian  
Courant Institute of Mathematical Sciences  
New York University  
{arvind.kumar,amey,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 (ELMER) 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. The design of ELMER employs a number of techniques including semantic compression of messages, a reduced query set, a user-based intermittent database consistency model and the batching of queries to reduce bandwidth and per-packet costs. Additionally, ELMER includes an SMS reliability layer to cope with poor wireless service, and a lightweight privacy model to prevent identity spoofing and theft of sensitive data. We have implemented ELMER 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. Currently, we are witnessing an explosive growth in the

availability and adoption of low-cost mobile devices in the rural developing world. Cell phones have enormous potential for enhancing rural healthcare, given their ability to act 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. The open-source movement in mobile software has opened opportunities for developing new mobile healthcare applications.

We aim to build a lightweight mobile record system that can enable health workers and patients in rural areas to fetch and update their patient records and also allow doctors to remotely track the health record information of patients. Existing mobile telemedicine efforts such as OpenRosa[4], OpenMRS[3], Voxiva[5] and Partners in Health have made significant advances in this space. However, all of the existing systems rely 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 economically viable only in urban settings; in most rural settings voice and Short Messaging Service (SMS) capabilities (140 byte packets) are available. Third, existing implementations may be heavyweight for low-end cell phones.

To address these limitations, we have designed and implemented Efficient Lightweight Mobile Records (ELMER), a generic record system that enables health workers and patients to access and update health records in extremely bandwidth constrained 140 byte SMS channel. ELMER is a client-server model, where the actual database sits resides at the server side and client provides an interface to the user to access and update health records over SMS. ELMER employs a suite of optimizations to support aggregate fetch and update operations in a single SMS message. ELMER uses an append-only database model and data inconsistencies are explicitly handled by the server administrator and users. ELMER also provides a lightweight reliability protocol to handle SMS message losses and provides a simple authentication mechanism to protect data privacy. In this demonstration, we show how ELMER can be applied in the context of an HIV application we have developed for use by African health workers.

Section 2 describes the Application Scenario. Section 3 describes the System Overview and Section 4 describes the Demonstration Content.

## 2. APPLICATION SCENARIO

Our research group directly works with several big 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. Hence, 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 general 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. Hence, 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. In our application, the doctor or the patient does not need to type a health record as an SMS message but the healthcare application specifies an interface to capture the user's query; the application automatically encodes the query and processes the SMS response. In the extreme case, we need text-free user interfaces to make the system usable for illiterate rural patients; in our prior study, we found that illiterate users are able to comprehend pictographic representations. 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 ELMER that enable the system to fetch and update several records in a single SMS message. ELMER is client-driven where the client specifies a 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[6] for sensor networks.

**Restricted Set of Operations:** ELMER is primarily designed for medical health records. Therefore, instead of providing a full-

fledged SQL query interface, ELMER supports a restricted set of operations. In addition, ELMER supports a specific types of compact sub-schemas (reduced number of fields) which are the primary responses for any allowed operation. Every operation (fetch, search, create, update) and every sub-schema is associated with a unique pre-coded identifier. ELMER employs the technique of semantic compression in which the identifiers corresponding to each field, sub-schema, operations, operation parameters is known in advance to the mobile device (in practice, this information is seeded in the cellphone when the patient first visits a doctor). In a typical use case, whenever a client issues a query to the server it sends the appropriate schema ID, operation ID and associated information to the server in the form of a small message. An update operation is configurable to update specific fields.

**Aggregation of Queries and Updates:** Cell phone usage rates in rural Africa are fairly expensive ranging from 10-25 cents/min; hence, it is essential to aggregate queries and updates to save operational costs. Using semantic compression, ELMER can compress several queries, updates and query responses into a single SMS message. For example, if a patient consumes 7 medications per day, the consumption history over a month can be compressed within 140 bytes thereby making the usage costs of ELMER negligible.

**User driven intermittent consistency:** ELMER operates in intermittent environments where updates and access requests are intermittent. However, in typical health scenarios, simultaneous update and access requests to specific records are rare. Hence, ELMER adopts a user driven consistency model where any conflicts across updates are resolved by the user. In ELMER, fields within the databases are either append only or non-modifiable which simplifies the consistency requirement. Each update is time stamped along with the user's identity. A fetch operation outputs the append history along with the identities and conflicts are resolved by the end-user. In case of scenarios where explicit consistency guarantees are required, it can be easily incorporated in our design in which each client request comes with a time stamp and conflicting updates can be resolved in favor of the latest update.

**Lightweight SMS reliability:** SMS messages can be lost which can affect application semantics. ELMER 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. ELMER 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 ELMER can use a bit-vector based cumulative acknowledgement vector within the response and this simplifies reliability semantics. Unlike TCP, ELMER explicitly does not perfect reliability to prevent continuous retransmissions of lost messages (primarily to reduce the cost of expensive SMS messages). Instead, ELMER restrict the number of retransmissions to one; if even the retransmitted message is lost, ELMER reports the error to the user as "Unable to complete request".

**Lightweight Privacy:** ELMER provides lightweight data privacy using a collection of simple access control rules and symmetric key cryptographic operations. Every mobile device shares a different secret key with the server established offline at the schema setup time. Every message is associated with a 4-byte

message authentication code signed by the mobile device to authenticate to the server and vice versa. In addition, the server implements access control rules both at the database and at the record level based on the identity of the mobile device. Each DB is associated with a set of global identities with specific access policies for different fields and records. In addition, each record is associated with a access control list of specific identities.

**Cached Updates:** In ELMER, we cache the updates/append operations performed by Client and send the request to the Server after enough requests have been cached. This greatly reduces the number of SMSs transferred. Further optimization could be to send the difference between the previous version and the new version of the record to be updated.

#### 4. DEMONSTRATION CONTENT

We have implemented ELMER in J2ME using the Nokia Series 40 Version 3 SDK.

The main schema divided into the following sub-schemas: a) Demographic Information, b) Symptoms, c) Medical History, d) Medication Consumption, e) Diagnosis Information. The server uses Record Store Manager (RSM) of J2ME to store different sub-schemas.

There are 3 types of users of the systems: a) Health Workers, b) Patients and c) Doctors. Each user has different levels of accessibility to the different parts of the main schema.



Figure 1 Fetch operation

Figure 1 demonstrates the user experience while doing a fetch operation on the database.



Figure 2: Aggregate Fetch Operation

In addition to simple fetch operations (such as in Fig. 1), the user can also do an aggregate fetch operation, where she specifies a collection of patient ids and constraints. The constraint could be a single number 'n' (which means fetch last 'n' records) or a <field id, comparison type, value> tuple. Figure 2 demonstrates the user experience while doing an aggregate fetch operation.



Figure 3: Search Operation

Figure 3 demonstrates the search functionality. The user can specify a collection of search criteria in terms of values for different fields of the database along with field ids and comparison type. In addition to multiple search criteria, the user can also specify whether he or she wants to AND or OR various criteria. Due to the restricted GUI available on cell phone, each search criteria should be a simple tuple of comma separated values: field id, comparison type (</>=), the value. For example (5,=,50) means field 5 should be equal to 50.

Thus we can use semantic compression to enable a restricted set of operations over SMS in mobile distributed database systems in order to implement a cost-effective solution for availing the technology in developing countries.

#### 6. 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] Samuel Madden et al. "TinyDB: An Acquisitional Query Processing System for Sensor Networks" <http://tinydb.org>