Ubiquitous Data Capture for Cultural Heritage Research

Siddhartha Ghosh and Nick Ryan
Computing Laboratory
University of Kent,
Canterbury, CT2 7NF, UK
{sg55,n.s.ryan }@kent.ac.uk

ABSTRACT
In this paper, we outline the MobiComp context-aware support infrastructure and a testbed application FieldMap that provides access to previously collected information and supports data collection in the field.

Categories and Subject Descriptors
J.0 [**Computer Applications**]: General, field sciences; J.5 [**Arts and Humanities**]: Archaeology; J.4 [**Social and Behavioural Sciences**]; H.3.4 [**Systems and Software**]: *Distributed systems*.

General Terms
Measurement, Documentation, Design.

Keywords
Smart Environments, Ubiquitous Computing, Pervasive Computing, Context-aware, PDA, GPS, Data Collection, Archaeology, Cultural Heritage.

1. INTRODUCTION
Our interest in this area began with the Mobile Computing in a Fieldwork Environment project (1997-2000) in which we began investigating how mobile devices and context aware software might benefit the field activities carried out in a range of disciplines, including archaeology, anthropology and ecology [1] Since the completion of that project, we have developed a simple infrastructure, called MobiComp, which is intended to support the development of context aware and ubiquitous applications. One such application is FieldMap, essentially a simple handheld GIS and recording system based on our earlier experience of field survey requirements. FieldMap acts as an advanced testbed for MobiComp, and has been developed over several years of close cooperation and field trials with archaeologists from the Groningen Institute of Archaeology in the Netherlands. The MobiComp infrastructure and the FieldMap application, both written in Java, are outlined here and described in more detail elsewhere [5].

2. FIELD SURVEY REQUIREMENTS
Data collection in archaeological research may take many forms. Whilst much work is confined to relatively inexpensive desk-, library- or museum-based studies of secondary material, including literature and artefacts, the highest costs are incurred in the field-based collection of primary material through field survey and excavation. Mobile and ubiquitous technologies have much to offer in these areas, particularly if they can help to improve the speed, efficiency and accuracy and to reduce errors in data capture. An early example of such systems applied to excavation is presented in [2] and a more recent example is reported at [3].

Unlike excavation, which typically concentrates on small areas such as parts of a single settlement or human activity area, field survey may extend to cover a region of hundreds of square kilometres. In its more extensive form, individuals or small groups may roam over a large area, visiting previously known and other potential settlement areas, and recording landscape features and artefacts found on the ground surface. This approach is often used as a part of the preliminary investigation of an area prior to more intensive and systematic survey.

Before undertaking a preliminary survey, archaeologists would normally have spent much time studying documentary sources and museum material to gain an understanding of previous knowledge of the area. Much of this information would have been entered into a GIS and associated database. However, whilst the use of desktop information systems is now the norm in archaeology, access to much of this information is also desirable when in the field.

In the past, this requirement could only be met by carrying paper-based information, including annotated topographic maps. In a rugged landscape, or in windy or wet weather, such material can be difficult and inconvenient to use. One of the key functions of FieldMap (figure 1) is to provide a single handheld source of all information relevant to the surveyor's needs. The device and its associated GPS receiver can be carried in a pocket until needed. During preliminary survey, the user's current location can be superimposed on vector or raster map layers, thereby aiding the relocation of previously known sites, and information about these sites can be displayed by tapping on their map symbols.

The second key requirement is context-aware data collection. The aim here is to minimise the amount of information that must be entered manually by annotating collected notes with contextual information. As a minimum, this includes user identity, date, time and location. FieldMap allows existing notes to be edited and new ones to be created. These may be associated with a single point location, or attached to simple geometric shapes such as lines, circles and polygons. The shapes may be drawn manually on the displayed map, or

Smart Environments and Their Applications to Cultural Heritage, UbiComp'05, September 11–14, 2005, Tokyo, Japan.

collected automatically using the GPS data while the user walks over the area of interest.

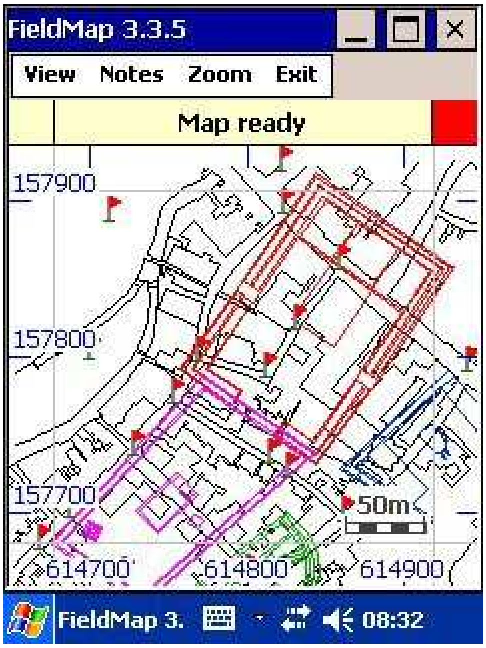

Figure 1: FieldMap map display showing multiple vector layers and clickable symbols representing previously recorded notes.

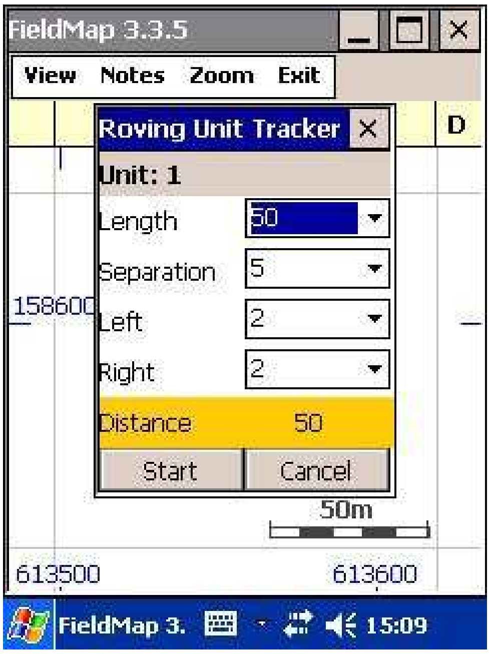

Figure 2: RovingUnitTracker dialog used to define the parameters for a survey unit.

Intensive survey methods typically involve a team of people systematically covering a landscape area and recording all artefacts found on the surface. The end result is often a set of map layers showing the density distribution of material of different types or periods. Approaches range from sampling to total coverage. A typical sampling approach is to use a team of, say, five people to collect material within 50x50m units. One member of the team paces out the size of the unit and marks its corners. The members of the team are then arranged 10m apart along one side of the unit. They then walk in straight lines to the opposite side, picking up all material found within one metre either side of their path. In this way, each member covers a two metre wide strip, with a resulting 20% sample of the covered area. At the end of each unit, the finds are gathered together, assigned an initial type and period and bagged for later study. Details of the unit location, size and all finds are recorded, usually on paper forms.

The process of pacing out the units before they are surveyed is time consuming, and the use of paper forms with subsequent transcription into a database is likely to introduce errors. Saving time and minimising such errors are the remaining key requirements addressed by FieldMap. Here, a RovingUnitTracker interface provides a mechanism for capturing all data needed to describe land units surveyed by the team. The geometry of the area covered is calculated from the number of walkers and their spacing which are entered into a simple dialog (figure 2). As the team moves forward, the dialog shows the distance covered and an alarm sounds once the required distance has been covered.

3. INFRASTRUCTURE SUPPORT

As described in [5], the core element of the MobiComp infrastructure is a context element store called the ContextService (figure 2). This represents a simple interface to a tuplespace, extended with event notification. It acts as a store for context elements and enables coordination between the components of context-aware applications. The approach here is similar to that employed in several other ubiquitous computing support infrastructures, for example the Stanford Event Heap [6]. Tracker components monitor sensors and other context sources and insert elements into the store. ContextListeners receive notification of put and remove events. Tracker and Listener interfaces may be combined to form context aggregators or to perform transformations.

In its minimal form the ContextService is simply a local cache for context elements allowing application modules running on an unconnected single device to share a common context store. More typically, the service is used in a connected or intermittently connected environment where ContextClient and ContextServer components handle communication between multiple devices behind the ContextService interface. Here, two modes of operation are possible. A more detailed outline of these different modes of operation is available at [7].

The first is intended to support simple sensor networks in a permanently connected LAN. One or more devices act as context stores. These receive context elements from simpler embedded devices, such as the Dallas Semiconductors TINI [8], and can respond to requests from other applications running in the network. The embedded devices run Tracker components to monitor sensors, and their local ContextServices include simple clients that, on joining the network, use multicast service discovery to locate one of the main services, and then establish a socket connection to pass their context elements to the service.

The second mode, more appropriate for mobile applications, uses an XML protocol over HTTP to pass context elements to one or more servers at known URLs. The protocol also allows applications to request contextual information about other devices from the servers. The servers act as central repositories and enable independent applications to share contextual information. When a mobile device contacts a server, its current network address and other capabilities are provided as part of its context. So, in an infrastructure-based or ad-hoc wireless network, individual devices may discover each other's addresses from any node acting as a central server. Mobile devices such as PDAs and laptops may include HTTP servers as part of their ContextService, so they are then able to directly exchange contextual information.

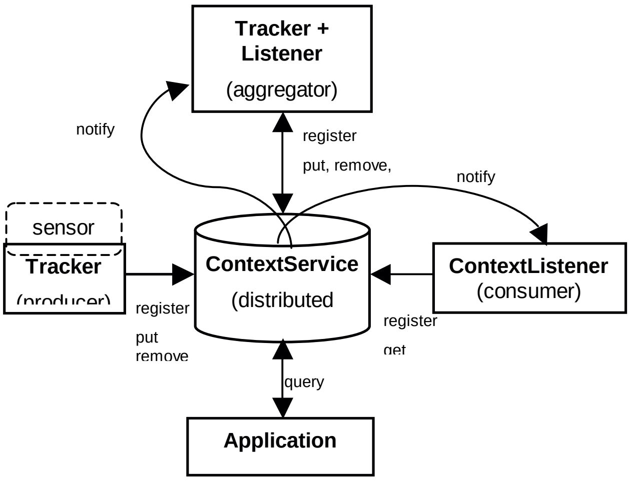

Figure. 3: The MobiComp infrastructure

However, whilst WiFi networks are suitable for applications intended for use within a limited area such as an urban centre, museum or visitor centre, their power drain on small devices limits their utility for field data collection. For the types of application considered here, WiFi is typically only enabled when the devices are within a small area such as a temporary field office.

Outside in the field, users may choose to run their devices in a disconnected mode, storing contextual information including all captured data locally. On return to the field office at the end of each day's work, WiFi is enabled and the devices attempts to connect to an HTTP ContextService running on one of the laptops in the office. Data collected during the day is then uploaded to the laptop server, and can be distributed to PDAs used by other field teams. In this way, each team starts the next day with a full complement of project data available on their PDA.

This approach works well for large projects in which several survey teams are active in the field over a period of several weeks. However, for smaller projects, and those where individual or small groups of specialists may visit the survey area at different times, an alternative, intermittently connected, mode may be more appropriate. Here, the PDA connects to the Internet at predetermined intervals via a mobile phone GPRS or UMTS link. Whenever a connection succeeds, recently acquired data is uploaded to a remote server, typically at the user's home institution.

This mode provides a near real-time tracking capability that would allow colleagues back at base to follow progress in the field. Although, this is rarely an important requirement of archaeological projects, it does provide an added benefit of increased reliability by providing secure backup of collected data and so protecting against possible PDA or other equipment failures in the field.

The HTTP server component is a Java servlet with equivalent capabilities to the *put*, *get* and *remove* operations of the ContextService. Unlike other implementations of the ContextService, it does not support the notification mechanism where devices register an interest in other devices, or specific context elements, and subsequently receive messages whenever these change.

When running on a small device, the infrastructure components include a small web server to support this, and other, servlets. HTTP request from the local or remote devices are then serviced by retrieving current context from the local store, typically the cache associated with the local ContextService. The cache is flushed to stable storage at regular intervals, overwriting previous values of context elements. Thus both cache and stable copy contain only current context elements. This same mode may be employed on laptops and desktops. However, for a field office server, or one back at the project base, it is more appropriate to use a DBMS to achieve persistent storage and maintain context history.

The current implementation of the persistent context store uses a PostgreSQL server, and takes advantage of the built-in rule mechanism to support historical queries. For example, it is possible to extract a client's location elements over an arbitrary period of time. An extension to the servlet XML protocol enables remote clients to use this capability. This store has proved effective over several years use with applications that employ a relatively small number of context elements with known semantics. However, the lack of a general query interface exposed through the servlet limits its potential for wider use.

However, as new applications are developed, new sensors become available, and an increasing number of people and devices are using the MobiComp services, the need to support much larger variety of context elements and tracker devices has become increasingly important. Our current aim is to allow new trackers to register new context elements together with information about their structure and semantics, and to allow clients to search for any form of relevant context.

The first steps towards this more generic handling of context is to enable trackers to register an XML schema fragment describing the structure and data types of the context elements that it may produce. To describe the semantics of the elements, trackers may also register OWL class definitions. Whilst these extensions are still at an early stage, we have also been working with an experimental context store that uses an eXist XML database [9] as the back-end to the HTTP servlet. Although, this makes the handling of historical context queries more difficult and, currently, rather inefficient when compared to the PostgreSQL database, it does allow us to easily add an XQuery interface to our servlet, with considerable potential for servicing arbitrary and potentially complex queries from clients.

4. EVALUATION

Since 2000, we have taken part in field survey projects undertaken by the Groningen (Netherlands) Institute of Archaeology at two field sites in Italy. This has formed part of a continuous program to develop and test FieldMap. Improvements and new functionality have been designed, implemented and tested in collaboration with the archaeologists who now consider FieldMap to be sufficiently robust and use it as a normal part of their research and student training.

Generally, the system works well and is sufficiently stable for production use by the teams of students participating in the Groningen survey campaigns. We are, however, always pushing the limits of PDA capabilities as new, experimental, functionality is added, and the scale of the field project data increases with successive seasons of work. Each field trial brings new problems, many of which can be traced to the limited memory of the devices, though we now have reservations about the Java runtime system.

We have been using the Jeode JVM, an implementation of the now outdated Personal Java, for the last few years. This runtime system is no longer widely available and has, for some time, been showing signs of age and lack of development. For the next period of field trials we hope to have completed the necessary changes to allow FieldMap to run under a J2ME CDC Personal Profile runtime environment and we will probably adopt the IBM J9 virtual machine [10] for this purpose.

Although the experimental wireless synchronisation is working, it is not being used yet as a part of normal field practice. Maintaining advanced databases and wireless networks is, we feel, too complex for many end-users to feel completely comfortable with the processes involved. In our next series of trials we will aim to provide automatic synchronisation between the PDAs and a master version of FieldMap running on a laptop, thereby hiding the complexities of the ad hoc network and database configuration from the users.

5. CONCLUSIONS

This paper began by setting the background to archaeological field survey methods and how these are addressed in the FieldMap application. FieldMap is a handheld GIS program designed to enable rapid data collection and information sharing in the field. It was built to serve two purposes; firstly to support experiments in the use of handheld devices in the field and, secondly, to provide an advanced testbed for the MobiComp infrastructure.

MobiComp, an experimental system for supporting distributed context-aware systems, was then briefly described. We also described current work on extensions to the MobiComp context store aimed at providing a more general and extensible ContextService.

Finally, we presented a brief evaluation of the system, pointing out some limitations and our plans to overcome them.

6. ACKNOWLEDGMENTS

We are particularly grateful to Martijn van Leusen and Peter Attema of the Groningen Institute of Archaeology. Since 2000, they have welcomed us as members of their survey projects in the Agro Pontino and Sibaritide [9] regions of Italy and, with many of their colleagues and students, have played an active role in the development and testing of FieldMap. Further development of MobiComp as part of an infrastructure to support smart environments across a range of cultural heritage applications is supported by the CIMAD project within EPOCH [10], the European Network of Excellence in Processing Open Cultural Heritage (IST-2002-507382).

7. REFERENCES

[1] Pascoe, J., Morse, DR., and Ryan, NS., Developing personal technology for the field, *Personal Technologies*, 2:28-36, August 1998.

[2] Ancona, M., Dodero, G., Mongiardino, M., and Traverso, A., Taking Digital Notes in the Field: the Archeo tool-set, in Barceló, J.A., Briz, I. and Vila, A. (eds.), *Proc. Computer Applications and Quantitative Methods in Archaeology*, BAR International Series 757, 117-121, Archaeopress, Oxford, 1999.

[3] IBM Wireless Technology penetrates prehistory, http://www-306.ibm.com/software/success/cssdb.nsf/CS/LBAL-5PWKPK?OpenDocument&Site=default

[4] Ryan, N.S., Pascoe, J. and Morse, D.R., FieldNote: extending a GIS into the filed, Barceló, J.A., Briz, I. and Vila, A. (eds.), *Proc. Computer Applications and Quantitative Methods in Archaeology*, BAR International Series 757, 127-131, Archaeopress, Oxford, 1999.

[5] Ryan, N.S., Smart Environments for Cultural Heritage, *24th International Research Symposium, International Research Center for Japanese Studies*, Kyoto, February 7-11, 2005. http://www.cs.kent.ac.uk/~nsr/papers/kyoto2005/SmartEnvCH.pdf

[6] Johanson, B. and Fox, A., *"The Event Heap: A Coordination Infrastructure for Interactive Workspaces,"* Proc. 4th IEEE Workshop on Mobile Computer Systems and Applications (WMCSA-2002), Callicoon, New York, USA, June, 2002.

[7] The MobiComp Infrastructure, August 4, 2003, http://www.cs.kent.ac.uk/projects/ubi/projects/infra/mobicomp/

[8] TINI – Tiny InterNet Interfaces, http://www.maxim-ic.com/TINIplatform.cfm

[9] eXist Open Source Native XML Database, http://exist.sourceforge.net/

[10] WebSphere Everyplace Micro Environment, IBM, http://www-306.ibm.com/software/wireless/weme/

[11] Attema, P., van Leusen, M., and Roncoroni, P., *Il Progetto archaeologico Raganello: rapporto preliminare 2002-2003*, Groningen Institute of Archaeology, Netherlands, 2004.

[12] EPOCH: the European Network of Excellence in Processing Open Cultural Heritage (IST-2002-507382) http://www.epoch-net.org