UCGE Reports
Number 20165

Department of Geomatics Engineering

Mobile GIS as if Field Users Mattered: Small is Ubiquitous but can Speech be Recognized?

(URL: http://www.geomatics.ucalgary.ca/links/GradTheses.html)

by

Andrew J. S. Hunter

October 2002

UNIVERSITY OF CALGARY

Mobile GIS as if Field Users Mattered: Small is Ubiquitous but can Speech be Recognized?

by

Andrew James Simpson Hunter

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF GEOMATICS ENGINEERING

CALGARY, ALBERTA

JULY, 2002

© Andrew James Simpson Hunter 2002

A b s t r a c t

The research asked whether Mobile GIS incorporating speech recognition was a viable tool for locating defects in the streetscape. The Geography Markup Language for encoding spatial information was used to implement an application schema for street condition surveys. Speech accuracy exceeded 95% in environments that were quiet or constantly loud. However, for tests where the noise level varied, recognition accuracy plummeted to 58%. Accuracy of captured defects was determined while “standing”, “walking”, “cycling” and “driving”. Errors ranged from 0.27m to 12.50m at the 95% confidence interval. A web-based questionnaire indicated that municipal geographic information users are unhappy with the quality of their data, and as yet do not require data in real-time. Future research involves investigating alternative ways of capturing spoken commands, the effect that mobile computing has on the cognitive abilities of the user, and wireless connectivity required for real time access to spatial data.

A c k n o w l e d g e m e n t

To Kerry, my wife and partner in life; to two bright lights in our lives: Lauren and Ellissa; to the light that is growing ever brighter; Thank you for your love and support, without you I would not be who I am today.

To Dr. Brian Ballantyne for his guidance and never-ending support, both intellectually and emotionally.

To Dr. Naser El-Sheimy for leading the way in the final stretch.

To Mum and Dad for giving me the support and the determination to see this through.

To Austin and Molly-Jo and the rest of the family for accepting me, putting up with me, and letting me be me.

To GEOIDE, the Department of Geomatics Engineering, and the New Zealand Institute of Surveyors for providing funding. Without your support, this could never have happened.

To Bruce Wright for the coffee and talks.

To Dr. Tak Shing Fung for helping me to make statistical sense.

To Dr. Vincent Tao for getting me underway.

Table of Contents

List of Tables

List of Figures

CHAPTER 1

1. INTRODUCTION

Geographical Information Systems (GIS) have gone mobile. Emerging technologies such as the Internet, wireless communication and mobile computing devices are changing the way GIS is being used by moving GIS from the desktop into field users' hands (Wilson, J.D., 2000). The advent of mobile GIS poses challenging research questions. What is the best means of interacting with a GIS in a mobile environment? Which of the broadening array of technologies are best suited to the mobile environment? What are the capabilities of Mobile GIS? Focusing on the theoretical foundations of Mobile GIS now will accelerate its development and ensure that the tools placed in the hands of those in the field can provide them with more meaningful and timely information.

By empowering field personnel with the responsibility of data acquisition, editing and verification, Mobile GIS applications have the capability of bringing field and office activities into a collaborative environment that can further improve productivity, reduce costs and minimize project completion timeframes. Making the technology truly effortless and natural to use should empower new communities of users, thus increasing the value of the software and databases being built now and in the future by government and the private sector.

GIS architectures have traditionally focused on a static environment in which users sit at workstations to perform spatial analysis. With the advent of ubiquitous computing[1], this setting has and will continue to change dramatically. Devices that combine a hand-held or wearable computer with a GPS receiver, a cellular phone and modem, and other technologies such as digital cameras, laser range finders, miniaturized gyroscopes and inertial navigation systems, to name a few, should enable users to integrate spatial analysis into their daily lives, opening GIS to the mass market.

The objective of this research is to develop a tool that simplifies the acquisition and

---

[1] Ubiquitous computing with respect to this research is the ability to be able to perform a computer based activity anywhere, without being constrained by place, or network connectivity.

maintenance of spatial information. The medium that has been used to test the primary question and objectives of this work is the Local Government environment, in particular the maintenance of street condition information. The City of Calgary Roads Business Unit carries out street condition surveys each Fall (Biensch, B., 2000). The purpose of these surveys is to plan maintenance programs for the following Spring. Specifically, the surveys identify and grade defects within the road surface, adjacent footpaths, curbs and other ancillary road furniture. At present the data is captured manually and subsequently entered into a relational database. According to City of Calgary Street Technician, Mr. Bernie Adams, who managers the Street Condition Surveys, this is a time consuming process in which there are few checks, save for next year's inspection.

1.1. Motivation for Research into Mobile Geographic Information Systems

Traditionally, due to the high cost of field acquisition, the capture of real world spatial information for inclusion in a Geographic Information System has been undertaken using techniques such as digitization or scanning of paper maps. These maps are essentially static abstractions of the real world at a specific epoch.

From a cartographic perspective the objective of these maps has been to create a model of reality that is primarily for metrical use and analysis. That is, measurements taken from the map will approximate closely those that would be attained were the same analysis carried out directly on the mapped environment. However, the scale of the map, the data collection and manipulation techniques utilized, the symbolization employed by the cartographer, and the medium upon which the map is produced all determine the accuracy of any measurement, and therefore the accuracy of any data derived from the measurements. Data accuracy is further compounded by the need to produce maps that are easily understood. This often requires that the cartographer rectify, enlarge or move elements on a map in order to clarify a situation. This practice leads to local differences in scale, rotation and translation (de Knecht *et al.*, 2001). The result is that the positional quality of data digitized from maps can be more than map accuracy requirements promoted by organizations such as the US Geological Survey[2].

Personal experience also indicates that geographic information (GI) users are inclined to believe that the accuracy of their spatial data is better than it really is. For example, while working for Truebridge Callender Beach Ltd. capturing water facilities for Wellington City it was found that the digitized positions of features with respect to positions obtained from the ex-Terralink International City Mapping Database had a positional accuracy of 2.38m at a 99% Confidence Interval (CI) for Johnsonville (Truebridge Callender Beach Ltd., 1999). The Council required an accuracy of 0.15m (the accuracy of the City Mapping Database) at the 99% CI, and refused to acknowledge that the plans provided for conversion could not achieve their accuracy requirements.

The conversion of paper maps to digital geographic features requires that information be converted to a digital data model. As described by Goodchild (1992) there are two fundamental (conceptual) ways of representing geography in a digital world: discrete objects and fields. In the discrete object view, the world is empty, except where it is occupied by objects with well-defined boundaries that are instances of generally recognized categories, characterized by their attributes. In the field view the world can be described by a set of spatially continuous functions, each measurable at any point on the surface, and changing in value across the surface. Objects are distinguished by their dimensions, and naturally form categories of points, lines, or polygons. Fields, on the other hand, can be distinguished by what varies, and how smoothly.

Most models used by today's GIS follow the cartographic tradition of organizing spatial data as points, lines, and polygons (Egenhofer *et al.*, 1999), otherwise referred to as the Vector data structure, being the logical model equivalent of the conceptual Discrete Object Model. From the perspective of municipal data, the Vector data structure is the traditional structure used both from a modeling perspective and from a cartographic perspective.

The process of converting paper maps to the Vector data structure, whether by digitization or scanning requires that certain procedures be maintained so that acceptable

---

[2] The US National Mapping Standard requires that the horizontal accuracy of not more than 10 percent of the points on maps of publication scales larger than 1:20,000, shall be in error by more than 0.8mm, measured at the publication scale, and for maps on publication scales of 1:20,000 or smaller, 0.5mm.

accuracy specifications are met throughout the life of a data acquisition project. The procedures will include an initial review of the data to be converted so that any data scrubbing[3] requirements can be determined, specification of registration requirements to ensure that any inherent scale distortions in the plans are either removed or minimized, specification of a scanning resolution to ensure position accuracy is maintained throughout the raster to vector conversion process, and the specification of adequate quality control requirements to ensure that the final product satisfies the needs of the user. As has been widely reported by authors such as Aronoff (1989), Burroughs (1998) and Montgomery *et al.* (1993), data acquisition is a costly process often surpassing 50% of GIS implementation or maintenance costs. Hence this is another reason why a one-stop data acquisition and maintenance tool is being investigated, as it is anticipated that fewer people will be required to perform the same task thereby reducing the cost of acquiring data.

If data has been acquired from a range of mapping frameworks they will require transformation to a common map projection. However, it has not been uncommon for organizations to maintain local or independent mapping systems, which, because the relationship between these mapping systems is unknown, may not allow the use of projection algorithms to transform data to a common map projection. In these instances the data must be integrated with existing data sets by other means. This merging of data can be performed in a number of ways but is largely determined by the information that is available on the source plans themselves. A common method employed is conflation -- meaning to fuse, to bring together, and to combine into a composite whole.

Conflation makes use of algorithms that are able to merge similar geometrical elements. The algorithms search for identical object structures in two data sets and use them for an exchange of attributes or for homogenizing geometry (Walter *et al.*, 1999). This approach is only able to handle data sets that are captured using exactly the same data structure. Essentially the spatial accuracy of a data set can be improved by stretching the less accurate data set so that it overlays the more accurate data set. Often the cadastral framework, that is a Digital Cadastral Database (DCDB), is utilized to control the

---

[3] Data Scrubbing is the cleaning up of line work on a plan so as to minimize the effort required to convert features contained thereon into digital form.

conflation process to ensure that consistency of position is maintained throughout the Geographic Information System.

Most early DCDB's have a spatial error in the vicinity of ±3m (The Geospatial Technology Report, 2000; Land Information New Zealand, 2002a). The accuracy of As-Built documents, from which most municipal data is acquired[4], in particular the earlier vintages, are at times doubtful, largely because of the methods employed by contractors to prepare these plans[5]. Given both factors, it is apparent that many of the features contained within a dataset do not meet accuracy requirements desirable to Local Government organizations[6].

We are now seeing a move by a number of municipal and national organizations towards a survey-accurate digital cadastre (AltaLIS Ltd., 2002; Falzon *et al.*, 2001; Survey Quarterly, 2002). This will allow the custodians of spatial data sets to migrate their existing DCDB based data to these improved cadastral datasets in order to take advantage of the increased spatial accuracy.

As indicated by the Data Accuracy Requirements Survey undertaken for this research conflation is commonly used to ensure that the relative location of features is maintained. However, it is unlikely that the positional accuracy of features will in reality match the accuracy of the upgraded cadastral data; accuracy could deteriorate as a result of inconsistencies between the existing DCDB and its upgraded variant.

A question that must therefore be asked is: How can the position of a feature be reliably captured or upgraded and remain within the accuracy specifications expected by municipalities? The most reliable method requires the physical location of a feature on the ground. Personal experience indicates that the primary drawback with this is that the physical location of a feature using current survey technology is relatively slow and labour intensive, and therefore expensive. Field surveys require a number of processing phases to be performed (capture, conversion, verification) before the information can be used within a GIS environment. Typically each phase is the responsibility of a different individual.

Experience has shown that if procedures are not clearly defined and well understood

4 Greater than 50% of data according to the Data Accuracy Requirements Survey undertaken for this research.
5 Refer to Truebridge Callender Beach Ltd. (1999) for examples of the likely errors obtained from As-Built plans.
6 Refer to the Data Accuracy Requirements Survey undertaken for this research.

by each person involved with the conversion process, and if each person does not have a good understanding of what the others are doing and the problems that they are encountering, then errors and/or omissions become regular occurrences. Each individual claims responsibility for a certain range of tasks. If this results in responsibility gaps within the process, a blame mentality ensues with regards to who should rectify these errors. As highlighted by Clampitt *et al.* (2000), this leads to low morale, high turnover, disorganization, and ultimately, frustration on the part of the client. However, the current trend within the workplace has been to reduce task fragmentation, foster internal job mobility and make work groups or teams responsible for the whole activity that is being performed (Bélanger, J., 2000).

How does this relate to Mobile GIS? With a one-stop data acquisition tool the field operator must take ownership of the whole data acquisition process, thus improving workflow by removing both duplicated effort and the need to communicate difficulties that may have been encountered in the field, but may not be obvious to office based colleagues. Therefore, if traditional data acquisition methods do not meet the accuracy specifications of municipal organizations, and traditional field survey techniques continue to be an expensive means of acquiring data, how can data be obtained that meet a user's accuracy requirements at a cost that is not prohibitive?

This thesis investigates a distributed Mobile GIS so as to determine its suitability as a data acquisition tool. The Mobile GIS has been designed around a wearable computer, which utilizes a multi-modal interface. Speech recognition and Text-to-Speech (TTS) technologies have been combined with the traditional keyboard, mouse, and visual display unit interface in an effort to create a hands-free computing environment. Wireless communication has been incorporated to provide increased mobility by allowing the GIS to be free from location constraints such as physical network connections, and distributed in the sense that data and/or services may be distributed among a number of computers.

It is anticipated that by mobilizing GIS, inspection and verification of captured data can be carried out at the same time data is being acquired, rather than sending out independent field crews to verify the accuracy of the data once it has been processed by office staff. It is proposed that by implementing a multi-modal interface, auditory verification of acquired feature attributes will simplify the verification process.

It is the intention of this thesis to ask if the use of network technologies such as the Internet and wireless communication, will allow the integration of field and office based activities thereby providing an opportunity to improve operational productivity in the Asset Management Arena.

This raises another series of questions: Why Speech Recognition? Why wearable computers? Why wireless communication? Since the advent of the computer, speech has been considered to be the most natural means of interacting with computers (Licklider, 1960). Speech recognition has enormous potential for changing and improving our interaction with computers in a hands-busy, eyes-busy environment (Murray & Jones, 1996; Tyfa & Howes, 2000). Speech enabled computing has been shown to improve productivity by up to 500% (Dàtria Systems, 2001) by allowing the operator to access information while continuing to work rather than having to stop and review maps or diagrams part way through a process. Wearable computers also provide visual feedback via head mounted displays while allowing the user to maintain an awareness of their physical environment, which is expected to translate into improved safety for the field operator.

The wireless component is the enabling element of a Mobile GIS. Wireless data access allows users to be more productive by providing access to information they need wherever they are, and permits information to be disseminated between field operators and process management personnel with minimal delay. Wireless networks provide the flexibility and freedom required to seamlessly integrate computing with field-based activities. As highlighted by Hunt *et al.* (2001), the elimination of redundant work effort, the ability to provide real time response to any query while in the field, and improved data through increased accuracy, are significant benefits in terms of cost reduction afforded by wireless communication.

1.2. Aims of Thesis

As we currently lack a model that allows real time mobile access of spatial data in a highly distributed computing environment, the aim of this research is to investigate the interaction of Mobile GIS with spatial information in such an environment and the integration and interoperation of multi-modal interfaces. However before this architecture can make any contribution within the commercial arena a fundamental question must be answered:

> *Is a Mobile GIS, that includes speech recognition and wireless connectivity for real time access to spatial data, a tool capable of working adequately for data acquisition?*

This question can be broken down into a series of objectives that will be addressed in the following chapters:

- To develop an architecture for Mobile GIS, using a wearable computer, based on the principles of interoperability;
- To investigate whether speech recognition can be used to capture spatial features and their attributes by determining if speech recognition responds with sufficient accuracy to meet geographic information users' requirements; by ascertaining if speech recognition responds in a timely manner (less than two seconds) so as to ensure accuracy of position;
- To investigate the positional accuracy of captured features using different modes of transportation being standing on a feature while capturing its position, and walking, cycling and driving over a feature while capturing its position. The goal is to determine the capabilities of such a tool in light of end user requirements relating to positional accuracy;
- To explore real time access and transmission of spatial information over a wireless communication interface. This will involve the investigation of the current state of wireless technology, and determination of the volume of data that can be reliably transmitted within an acceptable time frame.

In order to determine a number of hypotheses developed for this research a Data Accuracy Requirements Internet Survey (see Appendix A) was created and an email was sent out to a number of GIS and governmental bulletin boards such as those managed by GITA[7],

[7] Geospatial Information & Technology Association (GITA) is an organization created to provide information exchange on the use and benefits of geospatial information and technology in telecommunications, infrastructure, etc.

URISA, AURISA[8], NZ Local Government Online, New Zealand Institute of Surveyors, the NZ ESRI User Group and the GISList maintained by the GeoCommunity[9] directing them to the survey at the beginning of June 2001. The Internet survey format was chosen over more traditional mail and telephone methods so as to reduce the time required to conduct the survey and avoid the often error prone and tedious task of data entry (Medin *et al.*, 1999). The purpose of the survey was to determine: the spatial accuracy needs of end users; data capture methods; 'time to use' requirements and the level of validation of captured data. The survey was intended for Utility, Local, Provincial and Federal/National Government GIS project managers.

A review of the organizations to which this survey was directed may imply a heavy emphasis towards New Zealand institutions. The New Zealand groups have been included as this research has been partly funded by the New Zealand Institute of Surveyors, which necessitates that a New Zealand perspective is obtained to determine/ensure that New Zealand User requirements are not substantially different from those of North American users. Nevertheless, both the GIS List and GITA Lists are reported to have in excess of 1400 members registered, which is substantially more than the few hundred people registered with the New Zealand based groups.

1.3. Contents of the Thesis

The thesis consists of a total of eight chapters. A brief overview of chapters' two to eight follows.

Chapter two provides an introduction to Mobile GIS. Specifically, it sets out the requirements for a Mobile GIS application from the perspective of the end user. While the prototype developed for this thesis is based around a wearable computer, it has been the intention of this research to develop an architecture that could be transferred to any mobile platform with a minimum of effort. Current commercial mobile applications are also reviewed.

---

8 AURISA is the Australian equivalent of URISA, Australian Urban and Regional Information Systems Association.

9 The GeoCommunity can be found at http://www.geocomm.com/. It is a web site specializing in Geographic Information Systems (GIS), CAD, Mapping, and provides access to data, software and industry news.

Chapter three discusses the theory behind the speech recognition component. The chapter reviews design criteria for speech recognition, the limitations of speech recognition and gives an overview of how speech recognition and text-to-speech works.

Chapter four examines the literature pertaining to Wireless technology and Global Positioning Systems. The wireless discussion is focussed on technology currently available to users, while the GPS discussion provides a broad overview of the technology with an emphasis on Differential GPS and Real-Time Kinematic GPS.

Chapter five describes the Data Model and Server component proposed for the Mobile GIS. The Client/Server architecture is the major execution model in networking and distributed systems. It is widely used in the computer industry, ranging from database access, file and printer sharing, and desktop windowing systems to information resource sharing. The primary challenge of a Mobile GIS is to create an architecture that can provide the same functionality no matter what device is being utilized in the field. Interoperability is discussed, as is the Geography Markup Language (GML), being the OpenGIS interface for spatial data interoperability.

Chapter six describes the Mobile GIS prototype developed for this research, highlighting the important issues addressed in chapters' two to five. A GML data model is developed specifically for a City of Calgary Street Condition Survey.

Chapter seven evaluates the Mobile GIS prototype. Each of the tests performed on the system are described, as are the acceptance criteria to be used. The first experiment looks at the viability of speech recognition; the second examines the positional accuracy of captured features.

Chapter eight serves as to link the earlier chapters and analyzes the process that has been undertaken in the development of a speech aware mobile GIS application. The chapter analyzes the questions posed regarding the aims of this thesis and assesses the viability of the Mobile GIS architecture investigated. The chapter concludes with a discussion of the mobile GIS prototypes limitations and suggests some areas for further work.

Chapter 2

2. Mobile Computing and GIS

This chapter presents an overview of mobile computing with particular reference to its adaption to Geographic Information Systems (GIS). It commences with an examination of requirements of more generic business based mobile systems which are then transposed into a geographical based computing environment.

2.1. The World of Mobile Computing

The desire of the corporate world to search out more economic business processes coupled with continuous advances in digital communication technology and the proliferation of portable powerful computers have led to a paradigm shift in the computing arena. Mobile computing has arisen from this change in the boundaries of traditional computing.

Interest in mobile computing commenced in the early 1990's in an attempt to facilitate mobile workers by providing them with access to corporate databases via a laptop and modem. The primary business case drivers for mobile computing applications has been (and will likely continue to be) increased employee productivity; faster decision-making; reduced operational expenses; improved customer service; and streamlined data maintenance (Wilson, 1998).

This mode of computing activity is intrinsically different from more traditional activities that use laptop computers, which tend to support a stationary work environment at different locations. To expand this concept further, mobile computing is the use of computing devices - which usually interact in some fashion with a central information system – at some distance from the normal, fixed workplace. Mobile computing technology enables the mobile worker to: create, access, process, store, and communicate information without being constrained to a single location (Zimmerman, 1999). Therefore we could say that mobile computing is about allowing people to get the information and data they want without being constrained by place, and that it is somewhat of an umbrella term to describe technologies that enable people to access information and data via network services - anyplace, anytime, and anywhere.

As such, mobile computing must be viewed as a combination of three important and related properties: computation, communication and mobility. Computation includes the computing devices at either end of the network, which together provide the necessary processing power; communication systems include the different wireless and wired networks that link the computing devices; and mobility is an aspect of user behaviour (Liu *et al.*, 1995).

Mobile computing is synonymous with ubiquitous computing, a concept first advanced by scientists at Xerox Corporation's Palo Alto Research Centre in the mid 1980’s. They believed that people live through their practices and tacit knowledge so that the most powerful things are those that are effectively invisible when in use (Wieser, 2000a). It is for this reason that one of the principal concepts of mobile computing must be to make the computer, and its applications, so imbedded in our everyday practices that we use them without even thinking about them.

According to Weiser (1991), if computing is to become truly ubiquitous, three prerequisites must be met: computers must be small, inexpensive, low-powered and contain convenient displays; the network that supports ubiquitous computing must be robust and efficient; and there must be an adequate number of software systems that support ubiquitous applications. It is abundantly clear that in order for mobile computing to become imbedded in our everyday work practices, it too must fulfil all of Weiser’s conditions.

So what exists today that may allow us to meet these requirements? Computing devices that fall into this criteria are collectively known as Personal Digital Assistants (PDA’s) and Handheld Personal Computers (H/PCs or Palmtops), which include devices such as the Palm Pilot, Psion, Win CEs devices and Smart Telephones. Although other mobile computers such as Pen Tablets, Ruggedized and Wearable computers may not meet the cost criteria, these devices pack more storage and computing punch (Jonas *et al.*, 2000). What is significant about these devices is that they differ substantially from traditional desktop or workstation computing devices.

True mobility requires wireless communication, such as a radio or infrared connection. These communication technologies provide greater convenience in terms of mobility as shown by the rapid adoption (by police departments in particular) of mobile radio systems soon after the invention of radio or wireless communication systems (Yacoub, 1993). However, while mobile wireless technology has existed for over 100 years[10], it does have some drawbacks; wireless networks are costly, they have limited bandwidth and provide a lower quality connection to wired networks, with additional interference (Lo *et al.*, 2000). Consequently, disconnection, whether intentional or not, occurs occasionally. This limits the communicationcapabilities of the mobile device.

So what is different about mobile computing? Is it only the size of the computer and the manner and speed with which bits are transported? In an effort to answer this question Satyanarayanan (1996) defined four characteristics that are intrinsic to mobile computing devices:

- While mobile devices will improve in terms of absolute ability, they will always be resource poor relative to their desktop cousins;
- Mobility is inherently hazardous, because mobile computers are more vulnerable to damage, loss or theft;
- Mobile connectivity is highly variable in terms of performance and reliability. Some buildings may offer reliable, high bandwidth wireless connectivity, while others may only offer low bandwidth connectivity. Outdoors, the mobile device may have to rely on low bandwidth networks that are not contiguous; and
- Mobile devices rely on a finite energy source. While battery technology will undoubtedly improve over time, the need to be sensitive to power consumption will not diminish.

Together, these constraints complicate the design of mobile applications. They require that the software design process for mobile devices must consider the resource limitations of the device within which the application is to be run. Of primary concern are

[10] In 1895, Italian inventor Guglielmo Marconi built the first wireless radio equipment and transmitted electrical signals through the air from one end of his house to the other, and then from the house to the garden (National Inventor Hall of Fame, 2001), and before the turn of the century he established the first mobile radio link between a land based station and a tugboat, over a 30 kilometre (18 mile) path (Yacoub, 1993).

the unique operating system environments such as Palm, Enabling the Provision of Open Courseware (EPOC), Java 2 Micro Edition (J2ME) and Windows CE; limited on-board memory; lower processor speeds and lack of storage space. It also requires that software include only those functions that are necessary for the task to be performed so as to conserve computing resources and extend battery life.

2.2. MOBILE COMPUTING BENEFITS

As with any new technology, the benefits of mobile computing can be classified into two categories – tangible and intangible. Tangible benefits are those benefits that can be more readily identified and are capable of being appraised in terms of an approximate value, typically money saved. Many mobile computing applications involve automating sales, improving customer service or gaining a competitive advantage - all of which are are intangible benefits that tend to be difficult to quantify.

The savings that result from staff reductions are probably the most obvious economic benefit associated with mobile computing. Mobile computing can lead to increased individual productivity, increased sales, more service calls and less time spent on administrative work, all of which can ultimately translate into a reduction in total time required to complete an activity (Intel Corporation, 2001; MobileInfo.com, 2001c; Zimmerman, 1999).

The capacity for mobile computing to improve field user's access to information is a result of improved information flow both to and from central information systems brought about via the use of wireless networks. This ability to access centralized information and make queries of corporate databases enables employees to get the information they need to complete projects without having to return to the office for data that was either not anticipated prior to commencing an activity or had been left behind by mistake. The mobile computer also enables transmission (or uploading) of current operational data from the mobile device to a central information system. Once uploaded, the data can be processed, and made available for all other users who have access to the central server. Thus, the information available to a mobile user reflects current information from other mobile users as well. In essence, mobile computing eliminates the delay that occurs when an employee must physically return to the office at the end of the day and submit paper forms so that data entry personnel can enter the information into the central information system.

Even employees who are not continuously connected to the server via a wireless link should experience improved information accessibility through mobile computing (Drinnan, 2002). [One phone call at the end of the day from the mobile user, via modem, is all that is required to transmit the entire day's transactions to a server, saving travel and data entry time (Dhawan, 1997)]. Additionally, any scheduling or assignment changes for the mobile employee for the following day can be transmitted to the employee during the same phone call.

In connected or weakly connected (intermittent connection) modes of operation, this means that mobile employees may be contacted throughout the workday via the mobile computing device. Additionally, it means that the employee has access to other mobile employees via email or other messaging schemes.

The direct measurable results of improved information accessibility - both to and from the mobile worker - are many. They include: improved customer service (Spencer, 2001); reduced cycle times – data is available as soon as it is acquired; greater accuracy; fewer complaints; and a reduction in required intermediate support staff (data entry staff are not required, for example, due to the implementation of a mobile computing system for the City of Calgary Street Department’s annual Road Condition Survey. This would remove approximately 110 man days to manually enter survey data (Interview with Mr. Bernie Adams, City of Calgary Street Department)). Improved information accessibility can also support many other improvements such as: elimination of extra travel; and a reduction of selling times (Dhawan, 1997).

Mobile computing enables improvements in the operational efficiency of organizations that integrate the technology into their fixed information systems (Hunt *et al.*, 2001). It enables the computing power and information contained within the fixed information system to be structured around the optimum work flow of a mobile worker, instead of altering the mobile worker's work flow to meet an optimum configuration for computing. The mobile computer stays with the mobile employee, instead of the employee being required to travel to the computer.

As an aside, individuals who use mobile devices, unconnected, wired or wireless, have also found that the benefits of accessing all their applications while on the move exceed the ergonomic shortcomings of small keyboards, small screens, short battery life, and variable access to network connections (Francis, 1997; Dàtria Systems, 2001; Hunt *et al.*, 2001).

2.3. Mobile Geographic Information Systems

With the convergence of powerful inexpensive hardware, standardized communication protocols and innovative software, it is now becoming possible to deploy GIS functionality in a mobile computing environment. Leading the mobile GIS movement are utility and infrastructure organizations (Wilson, 1998). However, GIS applications are only just starting to reach mainstream field operations. The slow implementation of field based GIS relative to other types of mobile business systems is primarily because they tend to be large and complex, and are often difficult to implement in the office let alone in the field, where the integration of different technologies compounds the problem. Nevertheless, by introducing GIS applications into the field, more meaningful information can be put in the hands of field personal. By empowering field personal with the responsibility of data acquisition, editing and verification, mobile GIS applications have the capability of bringing field and office activities into a collaborative environment that will further improve productivity, reduce costs and minimize project completion timeframes (Weber, 2000).

At the heart of mobile computing lies a need to deliver intelligence to the field to improve productivity and provide a competitive edge in the marketplace (Wilson, 2000). In order for a mobile GIS to be successful it must emulate existing field practices and eliminate repetitive time-consuming tasks, because the purpose of a mobile GIS should be to streamline work processes. Given that field crews traditionally have little computer training (Wilson, 1998), in order to minimize the leap from paper based processes to computer-based applications, mobile applications must be intuitive and transparent; the invisible servant.

2.4. Mobile GIS Components

The core components of a mobile GIS are the same as those found in more generic mobile business systems. There are three fundamental components (MobileInfo.com, 2001b): hardware, software and the wireless network, which connects the mobile device to a centralized data repository.

The hardware component consists of the mobile device; a suitably configured wireless modem; a Web Server with wireless support, i.e., a WAP Gateway, a Communications Server and/or a Mobile Communications Server Switch so that the mobile device can communicate with the Internet or an Intranet; and an application or database server that contains application logic and databases.

The software component includes the mobile device operating system (Windows 98/2000/NT, PalmOS, Win CE, EPOC, etc.); the mobile application user interface, which may be run through an Internet browser or microbrowser depending on the mobile device; application server and/or database server software; application middleware if the mobile device needs to communicate with legacy (predecessor) systems or web-based application servers; and wireless middleware that links multiple types of wireless networks to application servers.

The wireless network component may be either a private network such as that used by law enforcement or emergency services, or a public shared network that is provided by Canadian telephony organizations such as Bell Mobility, Telus Mobility, Roger's AT&T, Cityfone or Microcell. Connectivity to wired networks or wireless LANs may also be included depending on the requirements of the mobile application.

Although most currently installed field GIS systems use Windows based software running on notebook or pen-based PC's (Wilson, 2000), there is a trend to more innovative end-to-end business solutions that include work management systems (WMS), customer information systems (CIS), and GIS mapping and query tools, all based around thin clients[11]; wireless connectivity; and mobile integration. Wilson (2000) reports that

[11] In client/server applications, a thin client is designed to be especially small as the bulk of data processing occurs on the server, as opposed to fat clients which are designed to perform a considerable amount of data processing.

ruggedized notebooks (a computer strengthened for better resistance to wear, stress, and abuse) and pen based PCs have not been the most ideal computing solutions for field systems. They may be portable, but they are still heavy, cumbersome and expensive. Conventional wisdom holds that field-computing applications require a different breed of hardware (Wilson, 2000). These systems are used in conjunction with difficult tasks, often requiring heavy physical labour, and they must be able to withstand exposure to rain, snow, mud, dirt and extreme temperatures. While cost and convenience make devices such as a PDA or Palmtop attractive alternatives, their limitations in terms of disk space, memory and battery capacity can impose considerable restrictions on mobile applications.

2.5. Wearable Computers

An alternative to either the PDA or ruggedized classes of computers is the wearable computer. In order to convey how a wearable computer differs from a PDA or Palmtop, a more specific definition is that wearable computers have many of the following characteristics (Rhodes, 1997):

- Portable while operational: The most distinguishing feature of a wearable computer is that it can be used while walking or otherwise moving around. This distinguishes wearable computers from both desktop and laptop computers.
- Hands-free use: Military and industrial applications for wearable computers emphasize their hands-free aspect, and concentrate on speech input and heads-up display or voice output. Other wearable computers might also use chording-keyboards, dials, and joysticks to minimize typing. This distinguishes wearable computers from desktop computers, laptops and PDA's.
- Sensors: In addition to user-inputs, a wearable computer should connect to components such as wireless modems, GPS recievers, digital cameras, and microphones.
- Proactive: A wearable computer should be able to convey information to its user even when not actively being used. For example, the computer should be able to communicate that a new email has arrived and who the email is from.

- Always on, always running: By default a wearable computer is always on and working. This is opposed to the normal use of PDAs, which sit in one's pocket and are only woken up when a task needs to be done.

With wearable computers the information flow from human to computer, and computer to human, runs continuously to provide a constant user interface (Mann, 1998). Wearable computers are by their nature highly portable, but their main distinguishing feature is that they are designed to be usable at any time with the minimum amount of cost or distraction from the wearer's primary task, i.e., work or recreation. Traditional computing paradigms are based on the notion that computing is the primary task. Wearable computing, however, is based on the notion that computing is not the primary task. A wearable computer user's primary task is to perform a specific function within their environment while the computer works in a secondary support role providing information necessary for the user to fulfil their function (Man, 1998).

2.6. Wearable Computing Benefits

Wearable computers typically use sophisticated voice recognition technology, with headset-mounted communication and display capabilities, eliminating the need for keyboards or other interface devices. This means that by speaking into a microphone, information can be accessed and presented on a lightweight, heads-up display. Traditionally, when field crews require access to information they must stop what they are doing and reference hardcopy drawings and/or manuals. This method of accessing information reduces their productivity. Conversely, speech enabled computing can improve productivity by allowing the operator to access information while continuing to work. Speech enabled computing also adds significant value when a task involves walking and manoeuvring in tight spaces, using tools and ultimately using a computer to complete the task. This saves time and increases productivity by allowing information to be accessed while the user continues to work uninterrupted.

Hands-free operation may reduce data entry and retrieval times while on the move. It also allows information processing to occur at the same time - and in the same place - as the task itself. This results in faster, more accurate data capture, thereby allowing information to be made available when and where it is needed.

Xybernaut offers a wearable computer that weighs 1.5 kilograms and can be strapped to a belt or vest. With virtual-screen eyepieces or pen screens that attach to the wrist, Xybernaut provides flexibility over conventional handheld computers. When compared to larger conventional laptops, or pen computers, the wearable computer represents the difference between a toolbox and a tool belt. All the information traditionally found at a computer terminal, on paper maps and plans or microfiche, such as infrastructure records, manuals, codes of practice, can reside in this belt-on computer. Its lightweight modular design allows the physical configuration of the computer to be worn in the most comfortable, and efficient manner. Xybernaut's Mobile Assistant IV (MA IV) is powered by a 233 MHz Pentium MMX processor and can have up to 160MB of memory and 8GB of hard disk storage installed. The use of commercial technology and widely used industry standards such as the PCMCIA interface means the MA IV provides virtual plug and play capabilities for upgrades and enhancements (Xybernaut, 2001).

Workers who use wearable computers must deal with the overhead of a more complex operating system and interface. However, these devices pack more storage and computing power punch. From a safety perspective when comparing wearable computers with traditional computing paradigms, an added benefit of wearable computers is that a speech enabled computing application allows the user to maintain an awareness of their physical environment while focusing their attention on a task in their virtual environment. If the user is suddenly confronted by an adverse situation, they can quickly switch their concentration to the physical environment, ignoring the virtual. However it should also be kept in mind that recent studies such as those performed by Strayer *et al.,* (2001) have provided evidence which shows that those engaged in phone conversations missed twice as many simulated traffic signals as when they were not talking on the phone, and that they took longer to react to the signals that they did detect. Is talking on the phone any different than talking to a computer in terms of user safety?

PDA technology has yet to be speech enabled, largely due to the limited resources maintained by these devices. Therefore when operating in a mobile capacity one hand is needed to hold the PDA, the other hand manipulates the pen for scrolling, and both eyes are focused downward at the screen, thus increasing the likelihood of operator injury. With respect to traditional laptops and ruggedized computers that could incorporate speech recognition, they still require the user to use both hands to either hold the computer and/or manipulate a pen while both eyes are focused downward at the screen, which could compromise the user's safety.

2.7. FEATURES OF A MOBILE GIS APPLICATION

It is clear that mobile devices differ considerably from traditional desktop computers. They are produced in a wider array of forms and processor types. Screen sizes vary dramatically as do input methods. Most mobile devices are limited by disk space, memory and battery capacity, which can impose considerable restrictions on mobile applications, and the most significant difference is the labyrinth of connectivity options - dial-up, wireless, LAN, docking, and the Internet.

But what of mobile software applications? Is there a need to develop a new software engineering paradigm that will better serve the needs of mobile GIS and the myriad of mobile computing devices? Fundamentally, software engineering is about building special kinds of machines that can be installed within a computer to interact with the external world of the user in order that the user may perform a set of defined functions (Ostroff, 2000). The software process model commences with an activity, or function, in a recognized application domain, which is refined into a conceptual model that describes a solution for the activity. The conceptual model is then transformed into a formal model that defines what the software is to do, and that can be validated against the user's requirements. This is, in essence, an extension of the traditional waterfall model, i.e., the problem-solving paradigm, the first step of which is to decide what is to be done. Once the objectives have been determined, we next must decide how to achieve them. This is followed by a step in which we do it, whatever it was determined to be. We must then test the result to see if we have accomplished our objective. Finally we use what we have done (Blum, 1992).

Based on this development process, it would appear that we can continue with current software engineering methods. However, the mobile workforce is a new class of user that may have little computing experience, which in itself can present major challenges in retraining as they leap from paper to new sophisticated applications that are often based on new or reworked fieldwork processes (Wilson, 1998). Mobile workers are accustomed to working in rugged, and often remote areas, where wireless coverage may be intermittent. These factors require that software applications should be designed to cope with a wide range of working environments. The software should provide the user with the ability to gather information, execute functional activities specific to their job, provide quick access to external data, update the data stored on the mobile device, and synchronize the data with the external datasets. The application should be able to be used while in motion; it should also be uncomplicated to learn; and easy to customize, and facilitate self-reliance.

Other desirable features of a software application could include:

- Support for open standards – full support for open standards will reduce future application dependencies when any changes are required as the needs of the users mature;
- Support for a large number of users – the application must be able to handle a large number of users concurrently;
- Ability to work on-line and off – the application must allow users to work either off-line or on-line, which means that the mobile device must be provided with facilities to manage a subset of any database being used while the user is working off-line. This means that synchronization capabilities must be implemented in order that master databases can be easily updated;
- Support for a wide variety of networks - the application should be provided with the capability of working over various communications networks, such as Internet, dial-up, wireless or serial connections;
- Multi-functional – the application should support local and central database query, as well as the synchronization of information and two-way messaging;
- Integration with other applications - the application must be able to seamlessly integrate with existing information systems, without requiring any changes to be made; and

- Security – the application must support standard network security mechanisms that provide full authentication and security for access to the device as well as the network.

Given these features, mobile software applications should also embody Schumacher's Dictum[12], Small is Beautiful. A minimalist approach should be applied when developing the base application, as the more specific an application, the less likely it is to be overloaded with tools that are not utilized in day to day operations. Software flexible enough to meet a multitude of application needs will only tie up valuable resources in computing devices such as PDA's that are already resource-scarce.

A preferred implementation methodology is to integrate specific data acquisition, mapping and spatial analysis tools into applications packages or components that are only loaded on as as-required basis. When used this way, the tools will disappear as a separate program and emerge as functions in a broader system (MapFrame, 1998).

To summarize, a mobile application should help a mobile user to automate their entire workflow and improve their efficiency. The application solution should automate manual processes, and at the same time, eliminate redundant processes. However, a mobile GIS application should also support a number of primary and subordinate functions. Primary functions should include activities such as mapping and navigation (zoom and pan), data collection, query, update and transmission, remote data and component access (wireless), location determination (GPS), coordinate transformations, and speech to text capabilities.

A mobile application needs to offer functionality in a simple package, with the most important requirement being that the application should be able to work in the same environment that a prospective field user is currently working in. We can therefore refine the components of a mobile system so that it will function in a GIS based work environment (see Table 2-1). The three components of a mobile system remain unchanged; however there are two additions to the elements of the hardware and software components.

---

[12] Small is Beautiful is a commonly sited contraction of the title of E. F. Schumacher's 1973 book titled "Small is Beautiful: Economics as if People Mattered" in which Schumacher challenges the doctrine of economic, technological, and scientific specialization and proposes a system for Intermediate Technology, based on smaller working units, co-operative ownership, and regional workplaces using local labour and resources.

First, by adopting a wearable computing environment speech recognition technology must logically be included in the software component. Second, because position is of paramount importance in a GIS environment, a Global Positioning System (GPS) device should also be included as part of the standard hardware configuration.

Table 2-1: Mobile GIS Components

Hardware Components	*Software Components*	*Wireless Components*
Wearable Computer, Wireless Modem, Web Server with wireless support, i.e., a Wireless Gateway, a Communications Server and/or a Mobile Communications Server Switch, An application/database server, GPS.	*Windows 98 Operating System, Mobile GIS application, Application server and database server software, Wireless middleware, Speech Recognition software.*	*Public shared network that is provided by organizations such as Bell Mobility, Telus Mobility, Roger's AT&T, Cityfone, Fido or Microcell. Connection capabilities to wired networks or wireless LANs.*

2.8. Existing Mobile Applications

As of June 2001, there were a number of mobile applications on the market designed specifically for PDA's or handheld PCs. In terms of GIS functionality, ESRI's ArcPad and AutoDesk's OnSite are typical examples of mobile GIS applications, however other organizations such as MapInfo (MapinHand), Tadpole-Cartesia (Conic GIS), and GE SmallWorld have also launched mobile GIS products. There is also a group of mobile applications such as Datria's VoCarta Field, iMeadon's iM:Field and iM:Collect, and PointBase's Mobile Edition that provide database management facilities for mobile devices. However, these latter systems do not provide any GIS functionality.

The general purpose of each of these applications is to allow mobile users access to corporate databases while on the move, and to improve field user's efficiency by providing the user with up-to-date information, and with facilities to update external databases while in the field. A number of applications including those listed above are briefly reviewed in Appendix B to provide a general indication of the present status of mobile applications that are related to GIS.

2.9. SUMMARY

To conclude this chapter on mobile computing it is evident that this form of computing is significantly different from typical computing environments and that these differences complicate the development of an effective mobile solution. Mobile computing must allow the user to access, create and communicate information regardless of where the user is required to work. Although there are a number of mobile devices currently available, the majority of these devices do not pack sufficient computing punch to be able to perform the tasks necessary of a mobile GIS application. In essence this lack of computing performance requires that we revisit development problems which have, of recent times, generally been disregarded thanks to Moore's Law[13] and the explosion in computing power, capacity and performance. The desire for maximum computing power is the reason that the use of wearable computers is being investigated as an alternative mobile device.

Within the world of mobile computing it has been widely reported that the capacity for mobile computing to improve field user's access to information is a result of improved information flow both to and from central information systems. The ability to make the computer fit the function, rather than the reverse, allows the user to get on with the task at hand rather than requiring the user to function in a stop start manner as they alternate between their two work environments.

It has also been shown that the core components of a mobile GIS are the same as those found in more generic mobile business systems with the addition of a map viewer with certain GIS capabilities required by the user and GPS to determine location. As such, we are seeing an explosion of mobile applications for geographic information.

[13] In 1965 Gordon Moore predicted that the number of transistors on a chip would double every 18 months, i.e., the speed and power of computers double every 18 months.

CHAPTER 3

3. SPEECH RECOGNITION

This chapter summarizes the recent literature describing how and why speech recognition, works, what the limitations of speech recognition are, and what should be done to maximize recognition accuracy.

3.1. SPEECH TECHNOLOGY

Speech recognition is an intuitively appealing computer input mode, because it is the most natural means of interacting with computers (Licklider, 1960). However, adoption of the technology has been slow. Speech recognition systems are probabilistic in nature and therefore subject to misinterpretation. The Achilles' heel of such systems is the rate of errors and lack of graceful error handling (Oviatt *et al.*, 1998). Although speech technology often performs adequately in idealized conditions, current estimates indicate a 20%-50% decrease in recognition rates when speech recognition is implemented in a natural field environment (Oviatt, 2000). One of the difficulties faced by speech recognition developers is that speech engines are designed and trained in a controlled environment, whereas natural spoken language often departs from the training data, resulting in recognition errors. To compound this problem, field environments usually involve variable noise levels in that background noise may be relatively quiet at one moment and then relatively noisy the next, thus increasing the computing demands of the speech engine as it tries to recognize spoken commands; multitasking in that the user may be concentrating on other tasks at the same time as operating the computer; increased cognitive load as a result of multitasking; and human performance errors, or unconscious mistakes that occur from time to time (Gong, 1995).

Oviatt (2000) reports that during field use, two main problems contribute to the degradation of system accuracy. The first is environmental noise, which contaminates the speech signal, making it more difficult to process. The second problem is that people speak differently in noisy environments.

White noise (constant background noise) sources can be modeled by a speech engine at the commencement of a session and this allows the speech engine to effectively remove the noise from the speech signal. However, many noises in the natural environment are not constant; they can change abruptly or may involve variable phase in/phase out noise (frequency shift) as a user moves past a source. These noise sources cannot always be anticipated or modeled.

When in noisy environments speakers automatically increase the volume of their speech, they tend to speak more slowly, and they may also change the pitch of their speech. This is called the "Lombard effect" (Junqua, 1993). The Lombard effect is reflexive, and as such, difficult to suppress. While it makes speech more understandable to the human ear, speech recognition tends to degrade due to the increased departure from training data obtained in a quiet environment.

3.2. Why Speech Recognition?

Traditionally, when field crews require access to information they must reference hardcopy drawings or manuals. This method of accessing information reduces their productivity. Conversely, speech-enabled computing offers the opportunity to improve productivity by allowing the operator to access the information while continuing to work. It is also generally accepted that speech enabled computing may find a useful role in hands-busy, eyes-busy situations (Murray *et al.*, 1996) such as activities that involve walking and manoeuvring in tight spaces, using tools and using a computer.

Speech technology, particularly speech command technology, has been shown to enhance data input and improve quality control (Pray *et al.*, 1998) by allowing the user to immediately review the acquired data. An example of the interaction between the user and a speech enabled mobile device could progress as follows: User: "Defect Type: Ravelling"; Computer: "Defect Type: Rippling"; User: "Correct Defect Type: Ravelling"; Computer: "Corrected to: Defect Type: Ravelling".

As can be seen from the example, it is anticipated that a speech enabled device can provide an interface that is simple to use. By utilizing commands that the user is familiar with, the transition from traditional paper based processes to computerized processing can be simplified significantly, allowing the user to get on with the work at hand rather than becoming frustrated with the device and ultimately contributing to low user adoption of the system. As reported by Hamel (2002) following an analysis of 1450 public and private GIS implementations across North America, one of the four primary causes of GIS implementation failure was a lack of user participation and/or adoption of the GIS.

One of the reasons for speech recognition technology's allure in hands-busy, eyes-busy environments is its ability to receive natural instructions which would otherwise require a number of manual processes in order to complete the same task (Tyfa *et al.*, 2000). For example with speech recognition we can instruct the computer to open a particular database with the command "Open Road Database", whereas on typical Windows® based programs we would have to navigate through a series of menus and possibly directories in order to locate and open the Road Database. In effect speech recognition capability can provide a shorter transaction cycle than keyboard or mouse based commands, allowing the computer to appear to respond more effectively to the user.

3.3. Speech Interface Considerations

Murray *et al.* (1996) have identified a number of factors that affect the usability of speech recognition systems. The first is that speech recognition is probabilistic by nature, in that recognition is determined by statistical analysis, and as such it is inevitable that misrecognitions will occur. The difficulty is that users must learn to contend with this type of error, which would not otherwise occur in a more traditional computing environment (e.g. keyboard and/or mouse). A hands-busy, eyes-busy environment will require that user interacts with the system entirely verbally, or with limited prompts and feedback. Finally, users are likely to have speech habits that are deeply ingrained, and which may not be wholly appropriate to speech recognition technology. It is therefore important that the design of a speech interface takes into consideration these factors in order to diminish their effects.

The principles of Graphical User Interface (GUI) design have been well established, in particular by the Windows® operating system. By maintaining consistency in design, developers can help speed the acceptance of an application, as basic features often remain constant among diverse applications. However, speech recognition interface design is still searching for a design standard. As a starting point, the presence of a voice interface should be immediately obvious at all times (Microsoft Corporation,, 2002a); this is typically done by having a “What can I say” window always open. Commercial speech recognition development organizations such as SmartSoft, Lernout & Hauspies, IBM, and Microsoft are now standardizing core features of their speech applications. These include control of user profiles, vocabularies, microphone status, and help information. As with a GUI, maintaining consistency among applications will help to promote the overall usability of speech applications.

Most applications fall somewhere among simple command driven applications, large vocabulary dictation systems, and data entry systems that utilize a relatively small and well defined vocabulary. It is this last class of speech recognition applications that is of most interest to this research. In these systems limited word-sets, called a grammar, are used to enhance speech recognition performance. As you move from one window to another within an application, different grammars become active, thereby minimizing the number of words that must be recognized at any one time and reducing the probability of misrecognition (see section 3.7.5 for more details). Another advantage of multiple grammars is that words that are acoustically similar can coexist in a vocabulary provided they are in separate grammars. However, as reported by Jones *et al.* (1992) the use of multiple grammars can introduce the possibility of the user losing track of which grammar is actually active, resulting in what can amount to a crashed application as the user no longer knows which command to issue in order to proceed.

Error correction is an essential feature of interface design. Jones *et al.* (1992) have identified four possible types of errors that may occur in a speech recognition application. Routines should be developed to handle: substitution errors when the speech recognition engine wrongly identifies a word that has been spoken correctly; insertion errors where an unrelated sound (a cough, the banging of a door, etc.) is matched to an active grammar; rejection errors when the speech recognition fails to respond to an utterance; and user errors when the user’s input or response is inappropriate to the task being performed. In general, routines should allow for immediate correction and/or backtracking with selective editing.

Research by Microsoft Corporation, (2002a) has found that in order for speech recognition applications to be adopted by users, error correction should reflect how errors are handled in a real application environment, much like that encountered when talking to a (human) customer services agent. For example, have the application say, “I’m sorry” after the first misrecognition. The assumption being that it was a recognition error caused by the person speaking too soon and that the command would be recognized the second time if it was repeated. If the application doesn't understand again, the system could say, “Sorry, could you please rephrase that”, in the event that they're talking out of grammar and they need to issue a different command. The third time, the application could say, “I’m having trouble understanding you. Try speaking clearly please”. This serves two purposes, the application is trying to correct the problem and to appear cooperative. This type of interaction is best determined by undertaking a usability study to determine how people talk in the particular environment for which an application is being developed.

Consideration must also be given to other types of human-computer interaction (Jones *et al.*, 1992). Currently, interaction with a computer is via visual aids on the computer screen, whether it be the use of a mouse, or an assistant such as Clipit, the Microsoft Office helper. However, if it is accepted that speech recognition is most advantageous in hands-busy, eyes-busy situations then an alternative method of communication must be provided between the user and the computer. The logical means of communication could take place in the auditory domain, as all commercial speech engines provide features that convert text to speech. Jones *et al.* (1992) also reports that the use of visual aids as an appropriate medium when using a wearable computer with a head mounted display. Appropriately sized text, possibly with animation, helps to draw the users’ attention away from what ever their current task is in order to convey a message or notify the user that something is required.

Speech recognition is not a replacement for the keyboard and mouse. In some, but not all circumstances, it is a better input device than either the keyboard or mouse. However, speech recognition is an ineffective pointing device (Haller *et al.*, 1984), just as the mouse makes a terrible text entry device. Generally speaking, every feature in an application should be accessible from all input devices, keyboard, mouse, and speech recognition

(Dragon Systems, 1999). Users will naturally use whichever input mechanism provides them the quickest or easiest access to the feature.

The number of voice commands that must be recognized at any given time can be significant. Therefore, to assist the user in locating the correct command, Dragon Systems (1999) suggests that an application prompt the user for the most common voice responses through visual aids or text-to-speech. For example, the application could say, “Do you want to save the file? Say Yes or No”. If the application does not recognize a command, it can also provide more extensive help. For example, “'Please say either Yes or No, or say “Help” if you need more help”. Both Dragon (1999) and Microsoft (2002a) recommend that whenever a voice command is spoken, the application must provide feedback to the user indicating that the command was understood and acted upon.

3.4. SPEECH RECOGNITION LIMITATIONS

The most commonly cited limitations to the adoption of speech recognition include physical difficulties from speaking continuously, and disruption caused by environmental noise, including other people speaking within the vicinity of a speech recognition user. There are also more subtle difficulties that arise when using speech recognition as a means of interacting with a computer. Shneiderman (2000) reports that the emotive content of natural speech, which is conveyed by prosody (defined as the pacing, intonation, and amplitude of spoken language), while important with respect to human-human interaction, can be disruptive to human-computer interaction. At present, commercial speech recognition software has no means of being able to adjust to the users mood. Hence the emotive content of natural speech is not only lost on the computer, it also does not match the way a user typically speaks during a training session in which they read a script to the computer.

Shneiderman (2000) also highlights cognitive difficulties. Their research has found that cognitive resources available for problem solving and recall are limited when speech input/output consumes short-term working memory. Basically, the portion of the brain that stores temporary information and solves problems also supports speaking and listening. This is why tough problems are best solved in quiet environments. However, because physical activity is handled in another part of the brain, people can perform physical activities at the same time as solving problems. In short, Shneiderman says that humans speak and walk easily but find it more difficult to speak and think at the same time. In a computing perspective, humans find it easier to type and think at the same time than they do to speak and think.

Of the more typical sources of error the microphone is considered to be the primary source of recognition errors. Of the many microphones currently available, the headset, or boom microphone, is most commonly used for speech recognition, but it must be located in the correct position (Dragon Systems, 1999, Microsoft Corporation,, 2000; Microspeech.com, 2002), about a thumb width from the corner of the users' mouth. By placing the microphone close to the mouth background noise can be minimized. Most speech engines available today include microphone wizards, which help the user to position the microphone correctly and make sure it is working properly.

An alternative to the boom microphone is a throat microphone which sits on the neck below the larynx and produces a signal related to vocal fold vibrations and sound pressure in the trachea (Askenfelt, 1980). The advantage of throat microphones, according to Askenfelt (1980) is that they are not significantly affected by environmental noise because the microphone is in direct contact with the neck.

Speech recognition engines are designed to "hear". Therefore background noise can sometimes be interpreted by the speech engine as words. There are a number of methods that can be employed so that these types of errors are minimised (Microsoft Corporation,, 2002b; Dragon Systems, 1999). Commands can be implemented to put the microphone to sleep when not in use; or the computer can be given a name that the user must say prior to speaking a command, so that the computer knows it is hearing a valid command; or the computer can verify every command with the user, so that if the user does not confirm the command within a certain time then the computer will not act upon it.

The final major limitation regarding speech recognition is computer hardware. Speech recognition USENET groups such as comp.speech.users currently recommend a Pentium® III 600 MHz processor, 384 MB of Ram, and an operating system such as

Microsoft® Windows® 2000 Professional to ensure that adequate performance is achieved. For more details regarding hardware requirements refer to Appendix C.

3.5. Commercial Speech Engines

At present there are a number of speech recognition applications available on the market. From a review of computing magazines there are three recommended commercial products being Dragon NaturallySpeaking 6, which is now being offered by ScanSoft; IBM's ViaVoice 9.0; and Lernout & Hauspies Voice Xpress Professional 5. Of these three products Dragon NaturallySpeaking and ViaVoice appear to be the best performers, with accuracy results ranging from 95% to 98% according to reviews undertaken by PC Magazine and CNet.com (Keizer, 2002; Alwang, 2000, Alwang, 2002a and 2002b). All three developers produce Software Development Kits (SDK), which can be used to extend their speech application software. In general the SDKs contain both ActiveX® controls and SAPI (Speech Application Programming Interface) methods for speech recognition, text-to-speech and telephony applications. Speech recognition engines are improving significantly from year to year. PC Magazine reports that the error rate is currently being halved each year (Jecker, 1999).

3.6. Command and Control Speech Recognition

Command and Control speech recognition allows the user to speak a word, phrase, or sentence from a list of phrases, and then has the computer perform a task related to the command. For example, a user might instruct the computer to add a new layer to a map view, zoom to a feature, or pan by issuing the following spoken commands, “Add Layer”, “Zoom in”, or “Pan”. In general, Command and Control recognition should be implemented to make an application easier to use; to make features in an application easier to get to; or, to make the application more realistic to use (Murray *et al.*, 1996; Microsoft Corporation, 2002b).

Command and Control recognition is typically used to provide answers to questions; activate macros; access large lists; prompt the user for required information; and facilitate hands free computing (Dragon Systems, 1999). Many database applications implement command and control functionality as a means of speeding up data entry as it is much easier for users to read data to the computer. This is particularly successful in situations where the data being entered is limited to predefined lists. When a database contains fields that are mutually exclusive, that is each attribute used in a database is unique, implementation of Command and Control recognition can remove the need for a particular data field to be in "focus"[14] in order for it to be populated. The speech engine simply "hears" the command and automatically determines which field the attribute belongs to.

3.6.1. Command and Control Grammar

Prior to a Command and Control recognizer "listening" for commands it must be provided with a grammar, or list of commands, to listen for. If the user speaks a command as written in the grammar supplied to the recognizer, very few errors will be generated. However, if the user diverges from the grammar supplied to the recognizer, for example by saying "add a theme" instead of "add a layer", and the computer thinks that it has recognized a valid command then there is a good chance that the computer may hear "close window". If a command is not issued correctly the recognition engine can not make a reasoned guess as to what the command should have been. Hence if the speech recognition engine does recognize an incorrect command, the probability of the engine recognizing the correct (intended) command is low. In order to minimize command recognition errors every endeavour should be made to implement commands that are intuitive to users. Lists of available commands should also be readily accessible from anywhere in the application.

Recognition can be improved if each of the commands sounds different. Generally, the more phonemes (a single distinctive speech sound) that differ between two commands the greater the likelihood of them sounding different to the computer[15]. See Appendix D for a summary of English phonemes.

Typically, speech engines cannot tell who is speaking, nor can they detect multiple

[14] A window, form, or database field is in "focus" when it is the active control.

[15] For example, "no" (phonetic spelling: \‘nO\; phoneme representation: N OW; phoneme symbols: 33,35) and "go" (phonetic spelling: \‘gO\; phoneme representation: G OW; phoneme symbols: 25,35) only differ by one phoneme, and are more likely to be mixed up than would commands such as "no way" (phonetic spelling: \‘nO\ \‘wA\; phoneme representation: N OW W AE; phoneme symbols: 33,35,46,11) and "go there" (phonetic spelling: \‘gO\ \‘[th]ar\; phoneme representation G OW TH AA R; phoneme symbols: 25,35,42,10,38).

speakers. Speakers with accents or who speak in non-standard dialects will obtain a higher proportion of recognition errors. Research indicates that multiple speakers, and speakers with accents, etc., will observe a 10% to 50% decrease in recognition rates (Babin, 1999). In order to minimize these sources of errors, training of the speech recognition engine should be undertaken by all users.

3.6.2. Text-to-Speech

Text-to-speech is a process through which text is rendered as digital audio and then "spoken" by the computer. Most text-to-speech engines can be categorized by the method that they use to translate phonemes into audible sound. Typical methods include (Microsoft Corporation,, 1998):

1. Concatenated Word: This is the most commonly used text-to-speech system. In a concatenated word engine, the application provides recordings for phrases and individual words that are pasted together in order to speak a sentence or phrase;
2. Synthesis: The text-to-speech engine generates sounds similar to those created by the human vocal cords and applies various filters to change the sound of the speaker.
3. Subword Concatenation: A text-to-speech engine links short digital-audio segments together and performs inter-segment smoothing to produce a continuous sound. In diphone concatenation, each segment consists of two phonemes, one that leads into the sound and one that finishes the sound.

Text-to-speech is used to communicate information to the user when digital audio recordings become inadequate due to the size of audio recordings, the high cost of obtaining recordings, or when the application does not know what information is to be communicated to the user. Text-to-speech is useful for phrases that vary too much to record and store using all possible alternatives; for proofreading or verification; notifying the user of events; and providing audible feedback (Microsoft Corporation, 1998).

While text-to-speech engines perform adequately when "speaking" individual words, they can become difficult to listen to when long passages are spoken. This is generally because text-to-speech engines still lack realistic human prosody.

3.7. How Speech Recognition Works

Speech recognition consists of five broad processes[16] as depicted in Figure 3-1 below, being the issue of a command by the user; the capture of the sound waves by the microphone; the conversion of sound waves into digital form; the transformation of digital audio from a sound card into the frequency domain so as to obtain a better acoustic representation, thereby making it easier to determine the phonemes that have been spoken, which is followed by a statistical matching of recognized phonemes with a grammar that has been supplied to the recognizer creating the spoken words; and finally a speech–aware application process' the recognized words and performs some function.

3.7.1. Transformation to Pulse Code Modulation (PCM) Digital Audio

Microphones convert sound waves into a series of electronic pulses which are transformed into PCM digital audio. In its raw form PCM digital audio is not particularly useful as it is very difficult to identify patterns that can be matched to what has actually been spoken. To

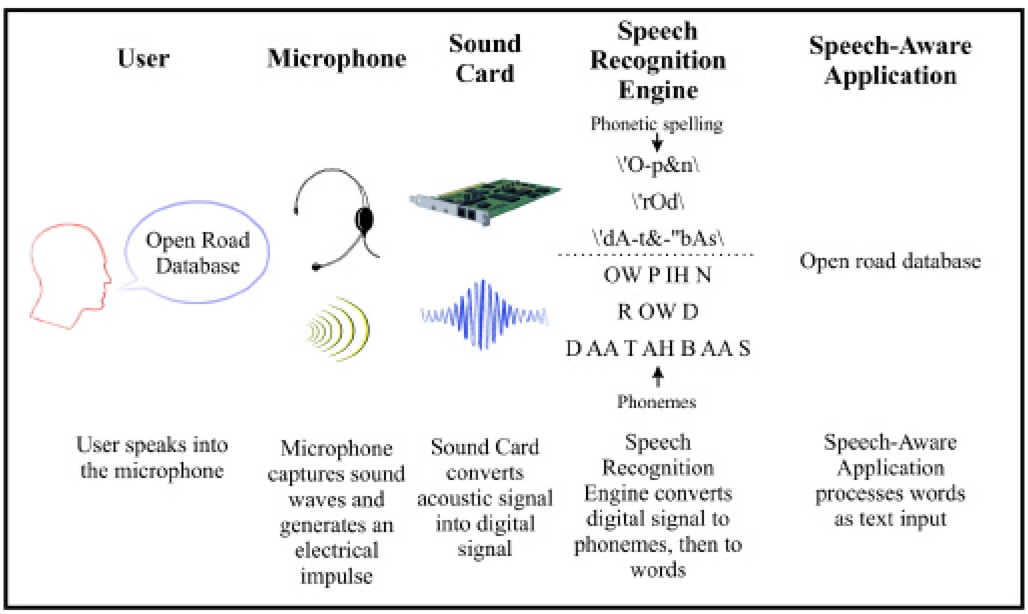

Figure 3-1: Speech Recognition Process Flow

16 This discussion on the workings of Speech Recognition and Text-to-Speech has been summarized from documentation provided with Microsoft's Software Development Kit. Microsoft's methodology has been reviewed as all speech recognition software that has been investigated as part of this research essentially enhances the Microsoft Speech Recognition Engine that is shipped with Microsoft Operating Systems.

make pattern recognition easier, the PCM digital audio is transformed into the “frequency domain” using a band limited Fast-Fourier Transform (FFT). The FFT is a class of algorithms that computes the magnitude and phase of energy versus frequency for a given signal. A FFT does this by assuming the time domain signal is composed of a sum of sinusoids of various frequencies. The algorithm computes the amplitude of each of these sinusoids and the result is plotted as magnitude versus frequency (Lathi, 1992). Figure 3-2 depicts this process. By converting the signal to the frequency domain it is possible to identify the frequency components of a sound and therefore approximate how a person may hear the sound (Microsoft Corporation, 1998).

A sound card such as the SoundBlaster Live series will typically sample an audio stream anywhere between 4,000 and 48,000 times per second (SoundBlaster.com, 2002). In order to reduce processing time, a FFT samples the PCM audio stream every 1/100th of a second and converts the audio data into the frequency domain. The results from the FFT are often displayed in the form of a graph of the amplitudes of frequency components, which describe the sound heard for that 1/100th of a second (see Figure 3-2).

The speech recognizer contains a database, or codebook, of graphs that identify different types of sounds the human voice can make. The sound is identified by matching it to its closest entry in the codebook and producing a set of “feature numbers” that describes the sound. Normally, multiple feature numbers are required to describe each graph obtained from the Fast Fourier Transform.

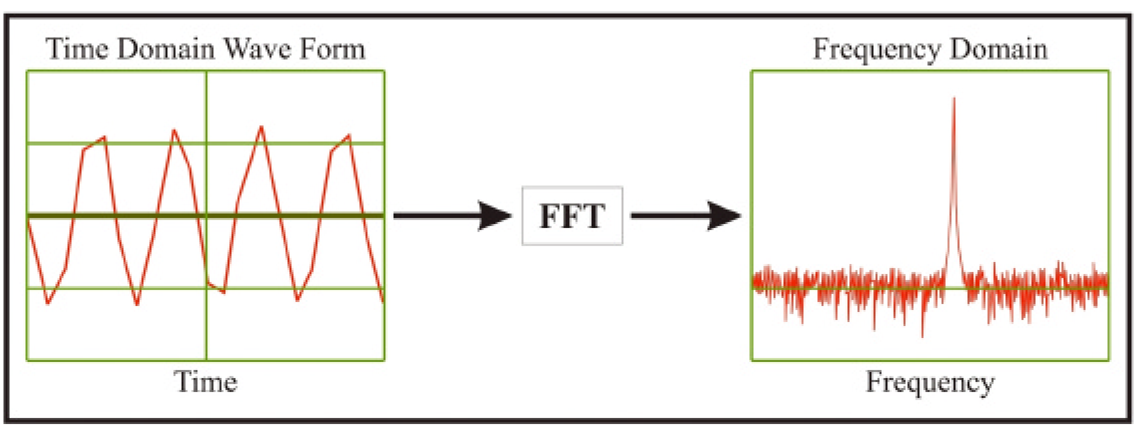

Figure 3-2: Frequency Component Determination using Band Limited FFT

3.7.2. Determination of Phonemes

Ideally, each feature number obtained from the FFT process would be matched to a phoneme, however this rarely occurs. It is very difficult for a speaker to produce exactly the same sound for a phoneme each time it is spoken, or background noise can vary resulting in the microphone hearing a different sound for the same phoneme, or the sound of a phoneme can change depending on what phonemes are either side of it, and the sound produced by a phoneme is seldom constant from beginning to end.

Background noise and speaker variation problems are resolved by allowing a feature number to be used to describe more than one phoneme. The spoken phoneme is then recognized using statistical modeling. Statistical analysis is possible because a phoneme often lasts for 50 to 100 feature numbers, and it is likely that one or more sounds are predominant during the time that can be used to predict the phoneme spoken. The speech recognizer must also determine where one phoneme ends and the next starts; this is solved using Hidden Markov Models[17]. Tri-phones are used by speech recognition engines to determine a phoneme that differs in sound because of the phonemes that surround it. A tri-phone is a phoneme in the context of surrounding phonemes. There are approximately 50 phonemes in the English language, which equate to around 125,000 tri-phones. If all tri-phones were included in an analysis then application performance would suffer, therefore similar tri-phones are grouped together. To get around the problem of a phoneme sound not being constant, speech recognition engines subdivide each phoneme into a number of segments called senones. The process of recognizing senones is the same as that used by a speech recognizer to identify phonemes.

Determination of a phoneme by a speech recognizer works by hypothesizing a number of different states at once. Each state contains a phoneme with a history of previous phonemes. The state with the highest score (statistically most likely state) is used as the final recognized phoneme.

---

[17] The Hidden Markov Model is a finite set of states, each of which is associated with a probability distribution. Transitions among the states are governed by a set of probabilities called transition probabilities. In a particular state an outcome or observation can be generated, according to the associated probability distribution. It is only the outcome, not the state, that is visible to an external observer, as such states are “hidden” to the outside; hence the name Hidden Markov Model (Duran, 1997).

3.7.3. Word Recognition

Once all phonemes have been identified they are compared against a dictionary of pronunciations to determine the word that was spoken. However, if a word is not pronounced as described in the dictionary it is probable that no match will be found, or the recognizer may select an incorrect word.

By reducing the size of a vocabulary the number of hypothesis that need to be generated to determine a word are greatly reduced. For example a 10-word vocabulary may only require 10 hypotheses, whereas a vocabulary of 60,000 words will require significantly more hypotheses, the number of which may increase considerably as additional phonemes are recognised.

3.7.4. Vocabularies and Templates

The words that a discrete speech recognition engine resolves are its vocabulary. A vocabulary consists of one or more templates for each word in the vocabulary. In speech recognition, a template is a pattern that can be used to identify or match to a speaker's pronunciation of a word. When a speech recognition engine tries to resolve a word, it compares the audio input stream to its templates until it finds a match or determines that no match is available. If a match is found the engine notifies the application that a speech command is recognized.

Templates may be speaker-independent in that they contain multiple pronunciations of a word or phrase, and require no user training; or they may be speaker-dependant, in which case a template uses a single pronunciation for each word or phrase. Pronunciations are trained by the user to improve recognition rates. Speaker-independent applications are used extensively for automated telephony purposes where the user (caller) is unknown. These applications require anticipating the varied responses to a question or system prompt. For example, if an application needs to be able to decipher an affirmative or negative response, the vocabulary might include “yes”, “sure”, “yep”, and “OK” for an affirmative response and “no” or “nope” for a negative response. Many speaker-independent applications are field-driven. That is, the application will “anticipate” certain vocabulary only when a particular field has been activated, or knows that a particular word can only belong to one field, negating the need for a particular field to be in focus prior to populating it with data.

In the Street Condition Survey application created for this research the user need only say the phrase “Defect rippling” to populate the defect field regardless of the actual field that is in focus. By limiting the number of acceptable values for this field, accuracy and computing efficiency are maximized. By utilizing unique commands, these inputs are not acceptable in any other field, minimizing the risk that they will be inserted in the wrong place by accident.

3.7.5. Grammar

A grammar is a subset of the available vocabulary that contains only the words used by a portion of the application. A grammar is normally created to avoid comparing a spoken command to all the words in speech applications vocabulary. When a user moves from one application context to another, the active grammar changes to the commands required for the new context. Managing grammars efficiently makes speech-enabled applications easier to use and speech recognition more accurate as the smaller the active grammar, the better the speech recognition. The active grammar consists of global command grammar and application command grammar. As the name implies global command grammar is continually active. It provides quick access to often used applications and system controls, such as “Zoom In” and “Identify feature”.

Microsoft Corporation, (1998) recommends a number of guidelines with respect to the selection of a grammar to ensure that speech recognition works effectively. Firstly the grammar should be consistent with the domain that an application is being developed for. Use words that are intuitive, and that identify the task that the application is to perform. The number of commands that can be issued for a particular view should be limited where possible, and they should be easy to remember. Avoid commands that sound alike, as this may result in false matches, and use the vocabulary supplied with the speech recognition engine.

3.8. HOW TEXT-TO-SPEECH (TTS) WORKS

The TTS engine takes text input from an application and synthesizes speech output. It then sends the speech output to the speakers. An application communicates with the voice-text object to provide TTS codes, as well as control codes that adjust speed, pitch, or cadence of the text spoken. The voice-text object controls the TTS engine that sends synthesized speech output to speakers. If you create a TTS notification sink, the voice text object can also send detailed information to your application about the timing of the text as played. Figure 3-3 illustrates a basic TTS system.

Voice responses that use recorded human speech phrases and/or units produce a more natural human sound, but require a large amount of memory to store recorded voice vocabulary. Reducing the size of the voice vocabulary reduces the memory needed, but it also limits the number of available responses. Voice responses that are computer-generated use the same phonemes that a speech recognition engine searches for. Computer-generated text-to-speech does not sound as life-like as recorded human speech. However, memory and vocabulary limitations are removed.

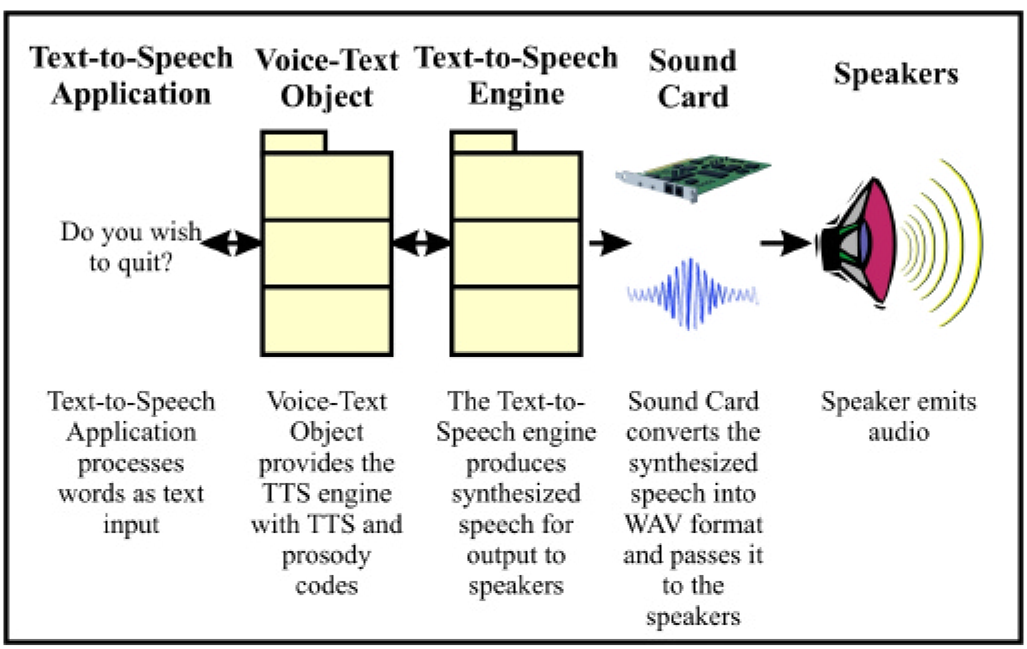

Figure 3-3: Text to Speech Processing

3.8.1. Text to Speech Processing

Text-to-speech converts text into PCM digital audio by performing text normalization, homograph disambiguation[18], word pronunciation, prosody and concatenation of wave segments. Text normalization converts text into a series of spoken words. That is, a string such as “Do you wish to quit?” is converted to “Do”, “you”, “wish”, “to”, “quit” along with flags to indicate punctuation. The text is then scanned for numbers, times, dates and symbols that need to be converted to words, i.e., “$54.32” is converted to “fifty four dollars and thirty two cents”. Lastly abbreviations are located and converted based upon a database of abbreviations contained in the Text-to-Speech engine. Once the text has been normalized and simplified into words it is parsed to a homograph disambiguation module.

Text-to-speech engines typically have a number of methods that are used to establish a single semantic or grammatical interpretation of a pronunciation. The most robust method is to determine the context of the text and select the pronunciation accordingly. This is generally carried out by looking at the word endings or by looking the word up in the text-to-speech lexicon. Once the homographs have been disambiguated they are parsed to a word pronunciation module.

The pronunciation model converts the text to a sequence of phonemes by looking the word up in a pronunciation lexicon, or if the word is not there, by using “letter to sound” rules built into the text-to-speech engine. A “letter to sound” algorithm segments a word into phonemes by determining which letters produce which sound, then the phonemes are matched to phonemes contained in the text-to-speech engine lexicon. The lexicon phonemes have sounds associated with them from which words can be assembled. Often combinations of adjacent phonemes will be required to determine which lexicon phoneme should be used. This process can pronounce any word, even if it is not included in the training set supplied with the text-to-speech engine. Once word pronunciations have been generated they are parsed to the prosody module.

18 Homograph disambiguation is the determination of a single semantic, or grammatical, interpretation of a word that is spelt similarly to another but is different in meaning, or derivation, or pronunciation. For example the word “read” may be pronounced \‘rEd\ as in “I would like to read a book”, or \‘red\ as in “I have already read that book” (Merriam-Webster’s On-Line Collegiate Dictionary, http://www.m-w.com/cgi-bin/dictionary).

Without prosody, text-to-speech sounds very robotic. Usually the prosody module will identify the beginning and end of a sentence so that pauses can be placed between sentences. In spoken English, pitch tends to fall near the end of a statement and rise for a question. Volume and speed tend to be higher when a person starts speaking and falls off towards the end. By locating the start and end of sentences these features can be built into the generated speech. Algorithms are then used to try and determine which words are important so that they can be emphasized. The output from the prosody module is a list of phonemes with the pitch, duration, and volume for each phoneme.

The final stage is to convert the list of phonemes with the pitch, duration, and volume for each phoneme into digital audio. The creation of digital audio is typically generated by concatenating recordings of phonemes. The difficulty with this process is that volume and pitch can vary significantly from one phoneme to the next resulting in noticeable changes, although these glitches can be minimised by blending the pitch and volume at the end of a phoneme so that it matches the start of the next. Once the digital audio segments have been concatenated they are parsed to the sound card.

3.9. SUMMARY

Speech technology, particularly speech command technology, has been shown in studies by Jones *et al.* (1992) and Murray *et al.* (1996) to enhance data input and improve quality control by allowing the user to immediately review the acquired data via text-to-speech functionality. However, speech recognition complicates application development as user interface rules are, as yet, poorly defined. One of the major issues with speech recognition technology is that speech recognition engines are designed to “hear”. Oviatt (2000) has shown that this design constraint can lead to a reduction in performance, particularly in high noise environments such as those found in urban street environments.

Managing grammars efficiently makes speech-enabled applications easier to use and speech recognition more accurate as the smaller the active grammar, the better the speech recognition. This facility can be implemented through the use of individual grammar sets that only function in specific parts of an application.

Chapter 4

4. Wireless and GPS Components

This chapter briefly summarizes the literature on the other major components to be integrated when developing a mobile GIS application. Firstly, we look at the state of wireless technology currently available to an organization or individual user. It is an overview of the various systems that could be implemented by a prospective developer. Next, a review of GPS is presented with particular emphasis on Differential and Real-Time Kinematic modes of GPS being the most appropriate means of determining location for this research.

4.1. Wireless Communication

Access to data via a wireless connection allows users to be more productive by allowing them to get the information they need wherever they are, and disseminate information between field operators and process management personnel. Wireless networks provide the flexibility and freedom required to seamlessly integrate computing with field-based activities.

Ideally, a mobile device should be able to select the network (LAN, the Internet, PCS, satellite, etc.) that best meets the user's requirements (Liu, 1995). However, a number of difficulties need to be overcome if wireless networks for mobile users are to prompt more extensive use. Firstly channel capacity normally available in wireless networks is considerably less than that which is available in wired networks due to the limited spectrum available, power restrictions and poorer signal to noise ratios. Secondly, security is of greater concern in a wireless network than in a wired network, as information is transmitted through space (Varshiney, 2000).

Performance and interoperability of wireless networks are also affected by the Media Access Control (MAC) utilised by a network. As yet no agreement has been reached between different carriers. The MAC protocols used in Cellular and PCS systems in the US and Europe differ considerably. The US standards use FDMA[19] (in Advanced Mobile Phone Service (AMPS)), TDMA[20] (in Personal Communication Systems (PCS)), and CDMA[21] (IS-95), while Global Systems for Mobile (GSM) uses TDMA/FDMA over different frequencies (Varshiney, 2000).

While wireless technology is the communication means of choice for this research, this section is but an overview of the networks that are currently in use or are being implemented at this time. Although a number of networks do exist today, most users are not given a great deal of choice as to which network they can use. For example, at the time of purchasing equipment for this research (October 2001), the only way in Alberta to use a wireless network as a means of communication in the field was via a wireless handset connected to a computer using a serial cable. This was not feasible due to the wearable computer only having one serial port (required for the GPS). Therefore, the only other means of communication was via a Cellular Digital Packet Data (CDPD) PC card. It should be noted that Novatel Wireless now (as of May 2002) supports a wireless PC card, the Wireless Merlin G100 PC Card[22].

4.2. Overview of Terrestrial Wireless Technologies

The wireless field has amassed decades of diverse research activities from the earliest experiments with radio waves to the recent inception of 3G wireless protocols. Currently the wireless industry and academia are concentrating efforts in developing 3G technologies that promise an increase in performance over first and second-generation standards that are in use today, in terms of both data transfer speed and user capacity. The following section presents an overview of pre-3G standards.

4.2.1. AMPS/CDPD

Advance Mobile Phone Service is a first-generation cellular telephone system standard that

---

19 In Frequency-Division Multiple Access (FDMA) protocol the spectrum is divided into sub bands and each sub band constitutes of a channel which can be dedicated to a particular user.

20 In Time-Division Multiple Access (TDMA) protocol, time is divided into slots and these are clustered into frames. Each slot is dedicated to a particular user.

21 In Code-Division Multiple Access (CDMA) protocol, different terminals transmit with different codes. Hence, a receiver tuned to the code of one specific transmission will interpret other transmissions as noise.

22 See http://www.novatelwireless.com/support/support_merlinG100.html for details (accessed 12 May 2002).

was developed by Bell Labs and AT&T during the1970's and 1980's. This analog-based system uses frequency bands around 800 - 900 MHz with channel bandwidth of 30 kHz (Lucent Technologies, 2002). Cellular Digital Packet Data (CDPD) is a packet-switched[23] data service that uses the existing AMPS network to transmit data at a rate of 19.2 kbps. CDPD supports network applications based on the Transmission Control Protocol/Internet Protocol (TCP/IP) and Connectionless Network Protocol (CLNP). CDPD provides a peer network (a direct computer to computer link) extension to existing data communications network. It is designed to operate as a transparent overlay on the AMPS system (Budka, 1997).

4.2.2. GSM

Global System for Mobile is a second-generation standard for cellular telephone systems that was developed to replace disparate first-generation European cellular systems (GSM World, 2002). The primary data service GSM offers today is circuit-switched[24] that provides data rates up to 14.4 kbps. A new higher-speed alternative service, called High-Speed Circuit-Switched Data (HSCSD), offers download speeds of up to 42.3 kbps and upload speeds up to 14 kbps by combining two to four of the six time slots in each frame. This service is available from operators such as Orange in the Switzerland, SmarTone in Hong Kong and Sonera Corp. in Finland (GSM World, 2002a). Most operators are not pursuing HSCSD and are instead awaiting the 2.5G technology called General Packet Radio Service (GPRS) (Rysavy, 1999).

GPRS is an IP-based packet, data only, system recently activated in Canada that has a maximum theoretical rate of greater than 171.2 kbps using all six time slots, but service providers will likely limit GPRS to two or three slots, giving speeds between 28 kbps and 56 kbps on downloads (Buckingham, 2000). GPRS service in Canada is currently provided by Microcell and Rogers AT&T for most metropolitan areas. For example in Alberta,

---

[23] Packet switches take a user's data stream, break it down into smaller segments, called packets, add network control information, and then transmit the packets through the network in bursts. Packet switches do not require dedicated paths over which the data must travel, unlike the more common circuit switched service.

[24] Circuit switching provides a physical, dedicated path -- called a slot -- for a call when it goes through the switching matrix. Because this path is dedicated to the call, no other callers can use that switch path until the call is ended.

GPRS service is available in Calgary, Edmonton, Okotoks, Banff and along Highway 2 between Calgary and Edmonton. Local service is provided by Fido using the Merlin G100 PC card.

A new radio interface, called Enhanced Data Rates for Global Evolution (EDGE), will theoretically propel GPRS to rates of 384 kbps under optimum radio conditions. EDGE, compared with GPRS, will be a costly upgrade for operators as they will need to replace 30kHz Base Station Radios to support the 200kHz data Channel required by EDGE. It is anticipated that many operators will leapfrog EDGE and go directly to 3G systems (Rysavy, 1999; LaForge, 2001). As yet EDGE has not been implemented in North America[25].

4.2.3. IS-54/IS-136

Developed in the United States, the IS-54 standard uses Time Division Multiple Access. The IS-136 TDMA air interface standard is a further development of IS-54 and is specified as a PCS technology for both the cellular (850 MHz) and PCS (1.9 GHz) spectrums. The peak data rate for a circuited-switched IS-136 connection is 9.6 kbps (Sollenberger *et al.*, 1999). Because of its similarity to GSM, IS-136 can incorporate the EDGE interface that will theoretically boost the data rate to 384 kbps.

4.2.4. CDMA

The Code Division Multiple Access Digital Cellular Standard was first commercially deployed in 1995 (Qualcomm, 2002). Today, CDMA networks are based on IS-95, a standard offering circuit-switched data up to 9.6 kbps. Operators in Japan and Korea have adopted an enhanced version of the standard, IS-95B, which increases data rates to around 64 kbps and is packet-based. CDMA is a spread spectrum technology in that its intended signal is spread over a bandwidth significantly in excess of the minimum bandwidth required to transmit the signal (Yacoub, 1993). A standard call starts at 9.6kbps and is

[25] According to correspondence with Ms. Geena Cabasug, Marketing Administrator for 3GAmerica, an organization that represents operators and vendors in the Americas for the following wireless technologies - TDMA, GSM, GPRS, EDGE, and UMTS (WCDMA) http://www.3gamericas.org/

spread and transmitted at 1.2Mbps. When the signal is received it is returned to a bit rate of 9.6kbps (CDG.org, 2002).

4.3. Wireless Summary

Most of the Canadian wireless telephony providers (Bell, Fido, Rogers AT&T, and Telus Mobility) offer PCS service which is a combination of AMPS and CDMA. Fido is the only Canadian operator of GSM services. All Canadian wireless providers except Fido support existing CDPD networks. CDPD networks continue to remain a logical choice for wireless data transmission as data transfer rates are in the same range, or better than other competing wireless technologies and CDPD use is generally charged using a flat monthly rate, as opposed to a time or packet based rate. Other notable data networks currently in the marketplace are the 800 MHz DataTAC® network and the 900 MHz Mobitex network[26].

Fido and TELUS Mobility are the only wireless communication providers currently available in Alberta. TELUS Mobility offers PCS services through PCS Online networks and TDMA (iDEN) services through MIKE Online. However, these are Dial-Up services, which implement circuit switched data transfer. Therefore the system acts essentially as a cellular phone. In order to use either the MIKE or PCS network with a computer the system requires a serial cable that connects your computer to a MIKE or PCS phone.

TELUS Mobility implemented an enhanced voice/data network in the spring of 2002. The service is called Velocity Wireless and operates on their national 1X network. Velocity Wireless is based on CDMA technology. It is reported that 1X can provide maximum data transfer speeds of 144kbps, with a normal transfer rate in the range of 40 to 60kbps. The network is accessible via a Sierra AirCard® 555 network card (Telus Mobility, 2002). This upgrade should not be confused with stage one of cdma2000 deployment which also uses the acronym 1X to indicate the use of one 1.25MHz channel as opposed to the second phase implementation of cdma2000, which is to use three 1.25MHz channels and goes by the acronym 3X.

Digital wireless coverage in Alberta is restricted to Calgary, Edmonton, Banff and

[26] DataTAC and Mobitex are packet-switched, narrowband PCS networks operated in Canada by Bell Mobility and Roger’s AT&T respectively, specifically for Research in Motion (RIM) products.

Highway 2 between Calgary and Edmonton. Both Telus and Fido report coverage is to be extended to include the Trans-Canada Highway between Calgary and Banff. There are now a number of wireless PC Card modems that may be plugged into a mobile device, which can be utilized in conjunction with either the TELUS Mobility or the Fido wireless network. The vast majority offer transmission speeds of 9.6Kbps or 19.2Kbps.

To conclude, CDPD services continue to remain a viable candidate for the near future. Because of the widespread use of CDPD and the recent uncertainty in the telecommunications sector it is likely to be maintained by service providers for some time. CDPD provides the ability to connect directly to the Internet via the TCP/IP protocol. This simplifies Internet based application development by allowing the developer to open a Windows socket on both the client and server and then to pass information between them without having to worry about the underlying carrier technology that is being used. In addition, applications developed using the TCP/IP architecture will be extensible as the proposed 2.5 and 3G telecommunication technologies all maintain TCP/IP layers within the communication stack.

Table 4-1 summarizes system parameters for different terrestrial wireless technologies available now or currently under development in North America.

Table 4-1: Terrestrial Wireless Technologies

Wireless Method	*Data Rate*	*Availability*	*Channel Bandwidth*	*Frequency Band*
CDPD (AMPS)	*19.2 kbps*	*Now*	*30 kHz*	*800-900 MHz*
GSM Circuit-switched	*9.6 to 14.4 kbps*	*Now*	*30 kHz*	*900 MHz*
GSM HSCSD	*28.8 to 56 kbps*	*Not yet*	*200 kHz*	*900 MHz*
General Packet Radio Service (GPRS)	*171.2 kbps*	*Now*	*200 kHz*	*900 MHz*
Enhanced Date Rates for GSM Evolution	*to 384 kbps*	*Not yet*	*200 kHz*	*900 MHz*
IS-136 Circuit-switched	*9.6 kbps*	*Now*	*30 kHz*	*900 MHz*
CDMA Circuit-switched	*9.6/14.4 kbps*	*Now*	*1.25 MHz*	*900 MHz/1.9 GHz*

4.4. Global Positioning Systems

The Navigation Satellite Timing and Ranging (NAVSTAR) Global Positioning System (GPS) achieved its initial operating capability in December 1993 (Leick, 1995)[27]. The system is funded and controlled by the U. S. Department of Defense. While the system was designed for the U. S. military, GPS has grown in popularity.

GPS was conceived as an all-weather ranging system from known positions of satellites in space to unknown positions on land, sea, or in space (Hofmann-Wellenhof *et al.*, 1997). A total of 24 satellites, or Space Vehicles (SV's), make up the GPS operational constellation, which orbit the earth in approximately 12 hours. The satellites have nearly circular orbits with an altitude of approximately 20,200km. The constellation consists of six orbital planes equally spaced 60 degrees apart, with an inclination of fifty-five degrees with respect to the equatorial plane and four SV's in each plane. There are also four active spare satellites which will be used to replace any malfunctioning SV. This constellation typically provides the user with at least SV's visible above a 15° elevation at any time of day, from any point on the earth.

The satellite orbits are observed by a monitoring network of five Monitor Stations around the world, with the Master Control facility located at Schriever Air Force Base (formerly Falcon AFB) in Colorado. These monitor stations measure signals from the SV's which are then incorporated into orbital models for each satellites. The Master Control facility collects this tracking data and computes precise orbital data (ephemeris) and SV clock corrections for each satellite. The Master Control station then passes this data to three Ground Control Stations which upload the ephemeris and clock data to the SV's, which in turn transmit subsets of the orbital ephemeris data to GPS receivers over radio signals.

Each SV transmits two microwave carrier signals known as L1 and L2. Both signals are derived from a fundamental frequency (f_0) of 10.23 MHz. The L1 signal is obtained by multiplying f_0 by 154 to obtain a frequency of 1575.42 MHz; L2 is obtained by multiplying f_0 by 120 to give a frequency of 1227.60 MHz.

---

[27] This discussion on Global Positioning Systems is an overview of the technology as described in the texts by Leick (1995) and Hofmann-Wellenhof *et al.* (1997).

Three binary codes are applied to a GPS signal to shift the L1 and/or L2 carrier phase. These are the Coarse Acquisition code (C/A), the Precision code (P-Code) and a Navigation message. The C/A Code (Coarse Acquisition), which has a frequency of $f_0/10$, modulates the L1 carrier phase. The C/A code is a repeating 1.023 MHz Pseudo Random Noise (PRN) Code and has a code length of 1023 bits. There is a different C/A code for each SV, which is used to identify a satellite. The C/A code that modulates the L1 is designated as the Standard Positioning Service (SPS) signal. The P-Code, designated as the Precise Positioning Service (PPS) modulates both the L1 and L2 carrier. The P-Code has a frequency of f_0 and is repeated approximately once every 266.4 days. In the Anti-spoofing[28] (AS) mode of operation, the P-Code is encrypted into the Y-Code by performing a modulo 2 sum[29] of the P-Code and the encrypting W-Code. The encrypted Y-Code requires a classified AS Module for each receiver channel and can only be used by authorized users who have access to the cryptographic keys. The P (Y)-Code is the basis for the PPS. The Navigation Message modulates both the L1-C/A code signal and the L2 signal. The Navigation Message is a 50 Hz signal consisting of a total of 1,500 data bits that describe the GPS satellite orbits, clock corrections, and other system parameters. In general a receiver requires at least 30 seconds to lock on to a satellite in order to receive the navigation message.

GPS receivers convert SV signals into position, velocity, and time estimates. Four satellites are required to compute the antenna position and receiver clock error. GPS observables are pseudo-ranges derived from code or carrier phase measurements. In differential mode the accuracy of code ranges are at the metre level, whereas the accuracy of carrier phases is at the millimetre level. For this reason carrier-phase tracking of GPS signals is the method of choice within the land surveying profession. L1 and L2 carrier cycles have a wavelength of ˜ 19 and ˜ 24.4 centimetres. If tracked and measured these carrier signals can provide ranging measurements with high relative accuracies under special circumstances. Tracking carrier phase signals provides no time of transmission

---

28 Anti-spoofing is a means of denying civilian users full use of the system. When activated the P-Code is encrypted to all but authorized users.

29 Modulo 2 sum is the equivalent of an "exclusive or" operator (XOR). It yields true if exactly one (but not both) of two conditions is true, i.e., "T" XOR "T" = "F" and "T" XOR "F" = "T"

information. The carrier signals, while modulated with time tagged binary codes, carry no time-tags that distinguish one cycle from another. As such, the measurements used in carrier phase tracking are differences in carrier phase cycles (Integer Ambiguity) and fractions of cycles over time.

4.4.1. GPS Error Sources

Although Selective Availability[30] has been removed from the GPS signal, there are still several errors that affect GPS accuracy. These errors can be categorized into three main groups. The first consists of satellite-induced errors, which include satellite clock inaccuracies and ephemeris errors. The second group consist of atmospheric induced errors, which include ionosphere and troposphere anomalies. The final sources of errors are receiver related and include receiver clock inaccuracies, antenna phase centre variation, receiver noise and environment errors generated by multipath conditions.

Systematic induced errors such as those related to the clock and ephemeris are relatively consistent and long term, but environmental errors can occur at any time, and are generally inconsistent and difficult to accurately predict or model. The systematic errors can be modeled and included in the observation equations as additional terms. Systematic errors can also be eliminated by appropriate combinations of observables. Differencing between receivers eliminates satellite specific errors, and differencing between satellites eliminates receiver based errors. As such double-differenced pseudo-ranges are generally free of systematic errors originating from satellites and receivers. However, with respect to atmospheric based errors we can assume that these are removed if there is little spatial decorrelation along the baseline. In addition, ionospheric errors caused by refraction of the GPS signal as it passes through the ionosphere can be minimized by observing the effect of the ionosphere on the code pseudo-ranges with respect to the carrier phase pseudo-ranges. Carrier phase pseudo-ranges tend to be measured short compared to the geometric range between a satellite and receiver, and code pseudo-ranges tend to be measured long. The difference is generally the same in both cases. The best means of minimizing multipath errors is to avoid, as far as possible, reflecting surfaces in the neighbourhood of the

[30] Selective Availability was a variable error that was purposely added to the GPS signals by the US Military to reduce its accuracy

receivers and by utilizing GPS receivers that feature multipath rejection algorithms.

The configuration of satellites ensures that the relative position of one satellite to another on a different orbital plane is changing constantly. GPS determines positions through trilateration techniques by determining distances between a GPS receiver antenna and the orbiting satellites. As such, the configuration, or geometric shape formed by the satellites affects how well positions can be determined in much the same way as it affects positional accuracy in traditional triangulation techniques. Therefore consideration must also be given to the configuration of satellites during an observation session in order to minimize positional errors. Satellite configuration is described by the Dilution of Precision (DOP), which is the mathematical representation of the quality of GPS data being received from satellites. DOP is mainly controlled by the number of visible satellites and their relative positions in the sky. The most commonly used dilution of precision is Position Dilution of Precision (PDOP), which is the combination of Horizontal Dilution of Precision (HDOP) and Vertical Dilution of Precision (VDOP). A PDOP value of 1 indicates an optimum satellite constellation and good quality data. Data quality decreases as PDOP increases (Dana, 1999).

4.4.2. Differential GPS

Differential GPS (DGPS) corrects bias errors at an unknown location (Rover) with measured bias errors at a known position (Base). DGPS assumes that the two receivers being utilized are fairly close to each other; say within 20 to 30 kilometres (Denys, 2002). The signals that reach both of them will therefore have traveled through virtually the same slice of atmosphere, and should therefore have virtually the same atmospheric errors. By observing the same satellites at both the Base and the Rover positions, clock errors will be substantially eliminated and ephemeris errors will be mitigated. Both the Base and the Rover must track at least four common satellites simultaneously.

The Base station receives the same GPS signals as the roving receiver but instead of working like a normal GPS receiver it resolves the equations backwards. Rather than using timing signals to calculate its position, it uses its known position to calculate timing. It determines what the travel time of the GPS signals should be, and compares it with what they actually are. The difference is the pseudo-range correction. In addition the Base also calculates pseudo-range rate corrections. The Base receiver calculates these values for all satellites in view, as it does not know which satellites the Rover can see, and sends the correction information to the Rover via telemetry, normally using the Radio Technical Commission for Maritime Services 104 (RTCM-104) standard. The Rover then makes the appropriate adjustments to the timing of each signal received from a satellite in its view that is included in the correction list received from the Base station. Positions with sub-decimetre relative accuracy are then calculated based on the adjusted timing signals.

Because individual pseudo-ranges must be corrected prior to the formation of a solution, DGPS implementation requires both software in the Base Station that can track all SV's in view and form individual pseudo-range corrections for each SV, and software in the Rover which must be capable of applying these individual pseudo-range corrections to each SV used in the position solution.

Table 4-2 summarizes typical error budgets for single point GPS and Differential GPS. Figure 4-1 depicts these errors graphically.

Table 4-2: Typical GPS Error Budget (in metres)

Per Satellite Accuracy	*Single Point GPS*	*Differential GPS*
Satellite Clocks	*1.5*	*0.0*
Ephemeris Errors	*2.5*	*˜ 0.0*
Ionosphere	*5.0*	*0.4*
Troposphere	*0.5*	*0.2*
Receiver Noise	*0.3*	*0.4*
Multipath	*0.0 – 0.1*	*0.0 – 0.1*

Source: Trimble Navigation Ltd, (2002a) and Radovanovic, (2000)

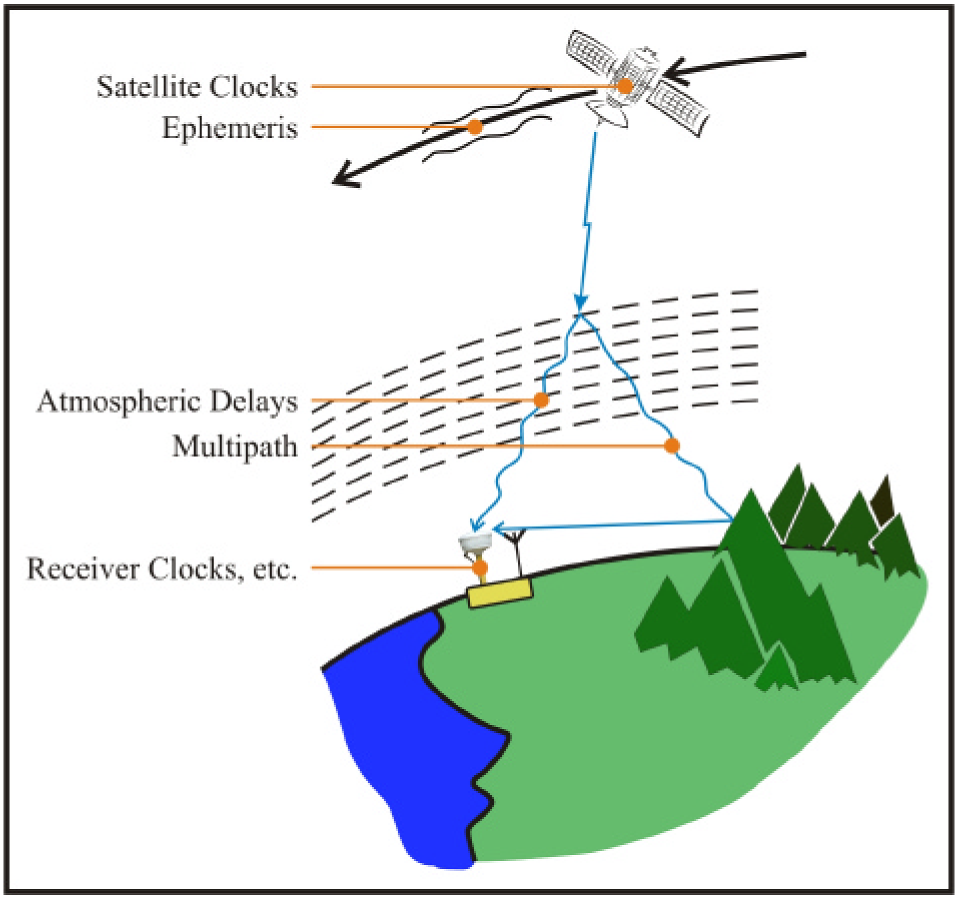

Figure 4-1: Summary of GPS Error Sources

4.4.3. Real Time Kinematic Positioning (RTK)

The *modus operandi* of kinematic positioning is to have two carrier phase receivers, both observing signals from the same satellites. As with DGPS, one receiver is located on a mark of known coordinates for the duration of a survey. Carrier phase differencing algorithms effectively cancel out errors related to satellite ephemerides, satellite and receiver clock errors, and ionosphere/troposphere errors. One of the fundamental differences with the kinematic technique using the carrier phase observables compared to the pseudorange positioning techniques (single point positioning and DGPS) is the necessity of determining "carrier phase ambiguities". Essentially, each GPS reciever must first be initialized in order to resolve mathematically the carrier phase ambiguities, or the differences in carrier phase caused by ionospheric refraction. When ever cycle slips occur as a result of signal disruption carrier phase ambiguity must be redetermined.

RTK involves determining position every time the receiver logs GPS data. The user determines the interval between positions by specifying an epoch, or measurement time interval. An epoch may be set to 1, 5 or 10 seconds (1, 0.2, or 0.1 Hz respectively), etc. The precision of an RTK determined point would be expected to be better than 10cm with a nominal precision in the order of ±3-5cm RMSE (Denys, 2002). Although the actual positioning technique is at the centimetre level, additional sources of error, such as the height of the antenna above the mark and the verticality of the antenna degrade the final coordinate slightly.

As with DGPS positioning, RTK requires a radio link to transmit corrections. Therefore this system also requires near line of sight visibility between the base and the rover receiver, which can limit the coverage of a survey. Coverage is also limited by the power of the radio system. To ensure the radio has sufficient power to transmit around (minor) obstructions and provide adequate coverage, RTK often requires the use of a radio for which an annual radio license must be purchased. Apart from the line of sight limitation on a radio system, there are some restrictions on the distance between the rover and the base station. For RTK, Denys (2002) recommends approximately 10km.

This GPS discussion concludes with Table 4-3 listing nominal precisions that can be obtained from a number of different GPS techniques along with the observables used by each method. As can be seen kinematic based surveys provide a high level of accuracy and are therefore the mode of survey employed for this research.

Table 4-3: GPS Techniques and Nominal Precision Characteristics

GPS Technique	*Nominal Precision*	*Observables*
Continuous GPS	*<±10mm*	*L1 + L2*
Static	*±5-30mm*	*L1 + L2 or L1 only*
Real Time Kinematic	*±1-5cm*	*L1*
Differential	*<±1m, ±1-5m*	*L1 + C/A, C/A*
Autonomous	*±2-10m*	*C/A*

Source: Denys (2002)

4.4.4. NMEA 0183 Interface Protocol

GPS integration has typically used the NMEA[31]-0183 interface protocol, created by the US National Marine Electronics Association, to allow marine navigation equipment to share information. For this reason, and because of the simplicity of the standards data structure and its wide support by major GPS vendors, the GPS component for this research utilizes the ASCII based NMEA-0183.

As described in Bennet (2000), data is transmitted in the form of “sentences”. Each sentence commences with a “$”, a two letter or “talker ID”, and a three letter “sentence ID” that indicates the format of data in the remainder of the sentence. The remainder of the sentence consists of a number of data fields separated by commas, and terminated by an optional checksum, and a carriage return/line feed. A sentence may contain up to 82 characters including the “$” and CR/LF. If data for a field is not available, the field is simply omitted, but the commas that would delimit it are still sent, with no space between them. Since some fields are variable width, or may be omitted as above, the data stream receiver can locate desired data fields by counting commas, rather than by character positions within the sentence. The optional checksum field consists of a “*” and two hex digits representing the exclusive OR of all characters between, but not including, the “$” and “*”. The sentences utilized by this application are the Global Positioning System Fix Data (GGA[32]) sentence and the Number of Satellites in View (GSV[33]) sentence.

A limitation of NMEA-0183 is that data can only be sent from one application to another – an application can not request data, it must wait for data to be sent from the GPS. This limitation can complicate a mobile field system as interpolation is then required to improve the spatial accuracy when the user is continuously moving (Harrington, 2000a).

---

[31] The National Marine Electronics Association (NMEA) is dedicated to the education and advancement of the marine electronics industry and the market which it serves. The NMEA standard defines a data protocol for communication between marine instrumentation, specifically GPS and marine navigation equipment

[32] The GGA data format is: Talker ID, GP sentence ID “GGA”, UTC, Latitude, Hemisphere, Longitude, E/W dir, Fix quality, Number of satellites being tracked, Horizontal dilution of position, Altitude, Metres, Height of geoid (mean sea level) above WGS84 ellipsoid, (empty field), Time in seconds since last DGPS update, (empty field), DGPS station ID number.

[33] The GSV data format is: Talker ID, GP sentence ID “GSV”, Number of sentences for full data sequence, Sentence number, Number of satellites in view, Satellite PRN number, Elevation, Azimuth, Signal strength - higher is better.

Although NMEA-0183 has a simple structure, information extraction is often limited by what information GPS manufacturers include.

4.5. SUMMARY

To sum up this chapter, an overview of wireless technologies and Global Positioning Systems has been presented. While most telecommunication providers offer wireless services, coverage of these services is restricted to major urban areas and the arterial routes that link them. Because of their relative simplicity, CDPD networks are currently the wireless service of choice, although Telus Mobility's recent launch of Velocity Wireless offers an alternative service for users requiring the transmission of larger data volumes. The Global Positioning System review outlined the three elements of the system and typical sources of error, which consist of satellite and receiver clock errors, ephemeris errors, atmospheric induced errors, antenna phase centre variation, receiver noise and environment errors generated by multipath conditions. Both Differential and RTK GPS positioning techniques were reviewed, being the modus operandi of choice for this research. RTK positions can be expected to achieve better than 10cm accuracy, but may have a nominal precision in the order of ±3-5cm. Lastly, the NMEA 0183 standard was introduced as a data interface between a GPS and a mobile GIS application.

Chapter 5

5. Data Models

This chapter introduces the concept of interoperability and the OpenGIS Consortium's implementation of a data transfer and storage interface called the Geography Markup Language (GML). GML has been designed to address technical issues relating to the interoperability of geospatial information.

5.1. Geospatial Interoperability

The heart of any GIS based application is its data model, which is used to describe and represent some aspect of the real world in a computer. As a guiding principle, the OpenGIS Consortium's (OGC) Interoperability program has been used in the development of the data model for the City of Calgary's Street Condition Surveys. In essence the vision of the OGC is to provide geospatial information users with the technology that allows them to access their information regardless of the network, application or platform that they are using, or that was used to generate the information (OpenGIS Consortium Inc., 1999a; van der Vlugt, 1999).

Through the 1970's and the early 1980's, most GIS applications were considered islands of information. They were self contained independent systems. In more recent times users have begun to realize that this batch orientated approach is inefficient (Bishr, 1998). From this realization has arisen the need for interoperable geographical information systems.

Interoperability is a challenging concept, as it involves not just technical issues but also institutional issues. The goal of interoperable GIS is to achieve an automated process that will allow Geographical Information (GI) users to access and make use of data and software services across the boundaries that the data collectors and designers envisioned (Egenhofer, 1999; Wiederhold, 1999). Users expect data exchange to be simple so that they shouldn't have to understand a great deal about how data is structured within a file, or how a particular import process works. The process of data exchange should be transparent in that any complexity associated with data transfer is hidden from the user. Data transfer should also be open and effective in as much as data exchange should be independent of the technology being used, and transfer must be reliable. Ultimately, users expect that data should be universal and that all geospatial databases should be accessible (Levinsohn, 2000).

Interoperable GISs can be decomposed into three broad elements (Goodchild *et al.*, 1998). The first includes technical aspects of GIS. This element deals with the compatibility of different computing environments, network protocols, data formats and techniques that can be utilized to remove implementation details from a user's problem. The second element deals with semantics. In reference to GIS, semantics deals with disparities between independent databases that contain the same objects or features (using OGC terminology), but have been defined using different world-views. The reliability of each world-view is dependant upon the creator's perception of the phenomena represented in a database (Mark, 1999; Martin, 1999; Peuquet, 1999; Raper, 1999). If each creator has a different perception of a particular feature then it is probable that the feature will be misinterpreted during the translation from one world-view to the other, if consensus is not first realized. To rephrase in the words of John Locke (1689: Book III: IX: 4), "…words serve not well for that end when any word does not excite in the hearer, the same idea which it stands for in the mind of the speaker."

Representation of GI by different groups of users, who each hold different world views of real world features is fundamental to the interoperability problem. A good example of different world-views of the same phenomena is soil type definitions. Fisher (1999) highlights the inherent complexity in defining soil by the fact "that many countries have slightly different definitions of what a soil actually constitutes." Bishr (1998), Leclercq *et al.*, (1999), and Kottman (1999) advocate the use of semantic translators that map the creators' world-view, via a correlation table, into a form that a recipient will understand.

The final element deals with institutional issues. That is an organization's willingness to be open and share versus protection of its interests; the added cost of achieving interoperability versus the benefits and value added by interoperability (which may likely be hidden); the right to know versus the right to privacy and protection of intellectual property; and the impacts of technological change on institutions that have been designed to achieve certain goals (Goodchild *et al.*, 1998).

The benefits of developing an interoperable system are numerous. Interoperability will simplify the interaction between the complex collection of formats and standards that exist within industry today. Interoperability will create a higher level of agreement of basic data models, which will provide transparency so that the user is no longer required to be aware of a data set's implementation details in order to utilise it. Software packages that are interoperable are likely to be stable since the same principles used in an initial application will need to be maintained in subsequent versions. Interoperability will also require a standardised theory of geographic data, which should ensure stability of software over time (Goodchild *et al.*, 1998).

Traditionally, interoperability has been achieved by simple translators in which translation relies on input and output data models being similar, otherwise loss of information results. However, regardless of the data model, direct translation also depends on the ability to read and write commercial data formats (Sondheim *et al.*, 1999). Vendors have made some format specifications publically available (ESRI's ShapeFile format), but not others (ESRI's ArcINFO Coverage format). When the number of data and service providers was small this approach was reasonable, however as numbers have grown, translation has become less manageable due to the increasing array of data formats and GI user groups who each hold to a different world-view. An alternative approach is interoperability by standardization (Landgraf, 1999). That is, all data and service providers would commit to a standard interface for GI. With this type of approach, the internal structure of the data is irrelevant. A standardized interface is able to provide or accept data in response to a request from a user - how it does it, it does not matter. There is no assumption that the data behind the interface must match the data provided to it or by it (Sondheim *et al.*, 1999).

This is the route that the OGC are pursuing. The OGC is developing an interface definition referred to as the OpenGIS Specification. Interfaces compliant with this specification can be incorporated directly into new systems and built into legacy systems

(Sondheim *et al.*, 1999). Two major components of the OpenGIS Specification include the Open Geodata Model (OGM) and the Services Architecture (OpenGIS Consortium Inc., 1998). The OGM incorporates fundamental geospatial data types, including their spatial representation, spatial reference and semantic content, and can be used to model the geospatial data needs of more specific application domains, using object-based and/or conventional programming methods. The Services Architecture provides a set of services by which individual objects and associated interfaces can be assembled into queries, transformations, analytical functions and presentation directives. It also enables the construction of catalogues that allow users to identify, evaluate, and interpret complex geospatial information dispersed throughout a network (OpenGIS Consortium Inc., 1998; Sondheim *et al.*, 1999). As such, OGC's Geography Markup Language (GML) has been selected as the interface between the mobile client and server.

5.2. Geography Markup Language

The Geography Markup Language is an eXtensible Markup Language (XML)[34] encoding for the transport and storage of geographic information, including both the spatial and non-spatial properties of geographic features. The specification defines the XML Schema syntax, mechanisms, and conventions (Córcoles *et al.*, 2001) that provide an open, vendor-neutral framework for the definition of geospatial application schemas and features. It supports the description of geospatial application schemas for specialized domains and information communities; it enables the creation and maintenance of linked geographic application schemas and datasets; and it increases the ability of organizations to share geographic application schemas and the information they describe (OpenGIS Consortium Inc., 2002a).

GML allows organizations to either store geographic application schemas and information in GML, as the schema can be directly mapped to a database application, or

[34] The eXtensible Markup Language is a subset of the Standard Generalized Markup Language (SGML). It is a language that describes the concepts and rules for the creation of specific mark-up languages. From a data-oriented standpoint, XML is the ASCII of the modern computing world. XML is an independent, global way to express any kind of information using constructs that can be accommodated to fit particular needs. Because the language shares common structures and concepts, it permits the interoperability and reuse of software that reads them.

they may decide to convert from some other storage format on demand and use GML only for schema definition and data transport. The GML specification, v. 2.1.1, is currently an OpenGIS recommendation paper, and has yet to be approved as a Technical Specification. This version also conforms to the current W3C[35] Recommendation for XML Schema, dated 2 May, 2001.

The GML specification requires that compliant XML instances shall be validated against a conforming application schema. A conforming application schema shall import the Geometry Schema (geometry.xsd), the Feature Schema (feature.xsd), and the XLinks Schema (xlinks.xsd) as base schemas.

The GML specification is based on the OGC Abstract Specification, which defines a geographic feature as "an abstraction of a real world phenomenon; it is a geographic feature if it is associated with a location relative to the Earth." (OpenGIS Consortium Inc., 1999b). That is a digital data model can be thought of as a set of features. The state of a feature is defined by a set of properties[36], where each property can be thought of as a {attribute, type, value} triple. According to this model, features represent real-world phenomena (such as streets, sidewalks, potholes, etc.), "attribute" specifies the relevant properties of a feature (such as the severity of a defect), "type" describes the named properties that a particular feature of that type has (the severity of a defect must be an integer between 1 and 5 inclusively), and "value" gives the specific qualitative or quantitative measurement pertaining to a particular attribute. A number of properties, in conjunction with a geographic feature type, establish the semantics of a feature. The properties are dependant upon the needs of the application that is being developed.

In OGC terms, a feature collection, such as the pavement surfaces making up the Calgary Street and sidewalk network, is a collection of features that can itself be regarded as a feature; as a consequence a feature collection has a feature type and thus may have distinct properties of its own, in addition to the features it contains (OpenGIS Consortium Inc., 1999b).

---

[35] W3C is the World Wide Web Consortium which was created in 1994 with the objective of leading the World Wide Web to its full potential by developing technologies (specifications, guidelines, software and tools) that promote its evolution and interoperability. The W3C currently consists of about 500 organizations.

[36] While it is common practice in the GI community to refer to the properties of features as attributes, for this chapter they shall be referred to as properties in order to avoid confusion with the attributes of XML elements.

As described in the GML Specification, GML is only concerned with simple features; “features whose geometric properties are restricted to ‘simple’ geometries for which coordinates are defined in two dimensions and the delineation of a curve is subject to linear interpolation.” (OpenGIS Consortium Inc., 2002a, pg. 4). The Simple Features object model consists of an abstract geometry class (the root class for this model) which includes traditional 0 (`Point`), 1 (`Curve`), and 2 (`Surface`) dimensional geometries, as well as collections of these geometries (homogeneous multi-point, multi-line and multi-polygon collections, or heterogeneous geometry collections). `MultiCurve` and `MultiSurface` abstract super classes were introduced by the OpenGIS Simple Features Specification for OLE/COM Revision 1.1 to generalize the collection interface to handle `Curves` and `Surfaces`. Each geometric object is associated with a Spatial Reference System (SRS), which describes the coordinate space in which the geometric object is defined. In all cases the parent geometry element is responsible for indicating the spatial reference system in which measurements have been made (OpenGIS Consortium Inc., 2001).

A generalization of the OGC Simple Feature Object Model for geometry is shown in Figure 5-1. It is Distributed Computing Platform (DCP) neutral (OpenGIS Consortium Inc., 2001) and uses the Unified Modeling Language (UML) notation. The figure shows the `Geometry` class as a generalization of `Point`, `Curve`, `Surface` and `GeometryCollection` classes, and that `Curve` is a generalization of the `LineString` class, etc. The figure also shows aggregation lines between the leaf collection classes and their element classes, i.e., a `Polygon` is an aggregation of one or more `LinearRing` features, and a `MultiPoint` collection consists of one or more `Point` features. GML’s implementation of the Simple Feature Specification does not include `Curve`, `Surface`, `MultiSurface`, and `MultiCurve` types.

By adopting the OGC Simple Features Model, GML could be viewed as somewhat restrictive in that “Simple Features” are assumed to only have simple properties (Boolean, integer, real, or string values) or geometric properties, and that geometries must be defined in a two dimensional SRS. As a consequence, simple features currently only support the

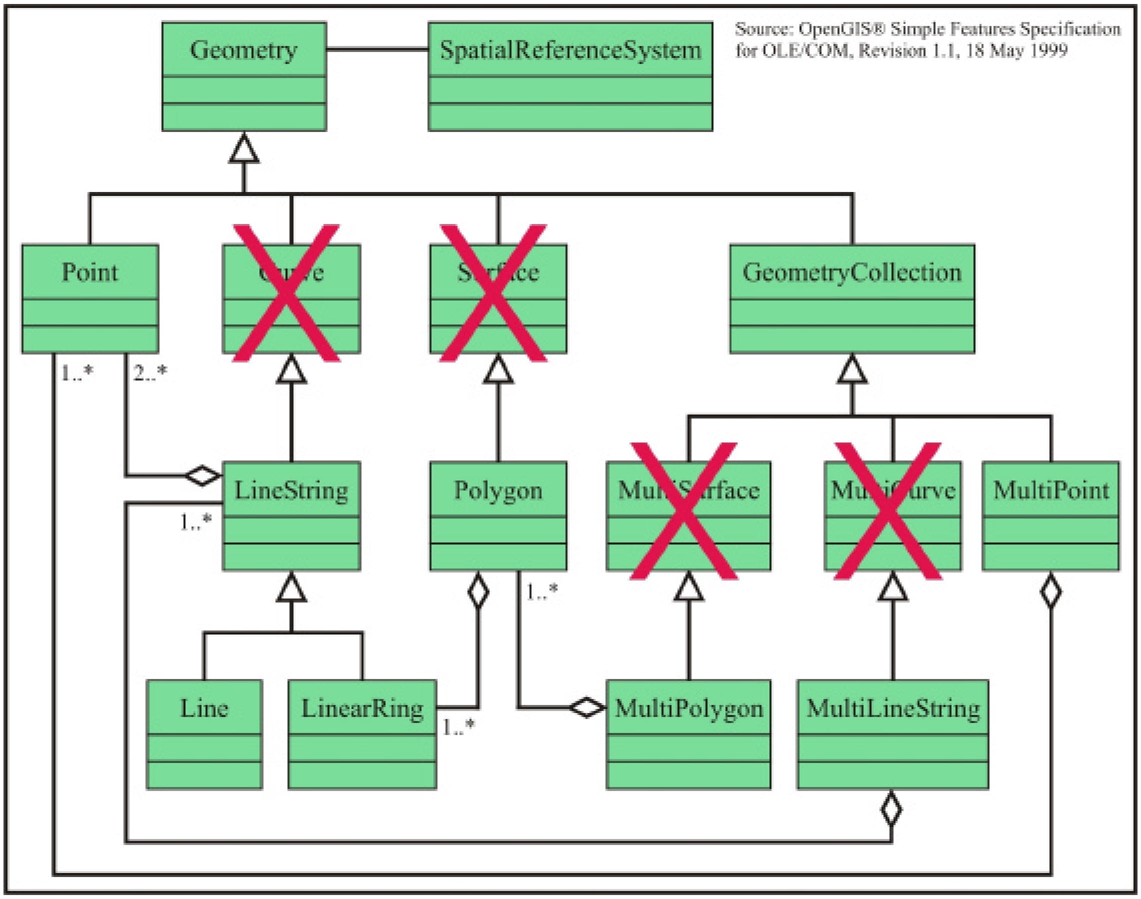

Figure 5-1: OGC Simple Feature Geometry Class Hierarchy

Vector data model and cannot incorporate topology[37] (although Application Programming Interfaces (API) based on OGC's Simple Feature Specification do provide functionality for common topological constructs such as intersect, difference, buffer, clip, convex hull, cut, union, etc.) (OpenGIS Consortium Inc., 2002b). However, GML has attempted to address some of these limitations with its latest release. Features may now include complex or aggregate non-geometric properties such as dates, times and addresses. Complex properties may also be composed of other complex and simple properties.

5.2.1. GeoSpatial Data Schemas

In general terms a Schema defines the characteristics of a class of features; in XML a

[37] It is anticipated that GML Version 3.0 will include an extension for topology.

schema also describes how data is marked up[38]. GML is designed to support interoperability and does so through its compliance with the XML Schema published by the W3C in two parts on 2 May 2001, being XMLSchema-1: Structures (World Wide Web Consortium, 2001a) and XMLSchema-2: Datatypes (World Wide Web Consortium, 2001b). GML extends XML by providing basic geometry tags (all systems that support GML use the same geometry tags), a common data model (features/properties), and a mechanism for creating and sharing application schemas (Córcoles *et al.*, 2001). GML has also been developed to be consistent with the the XML Namespaces Recommendation (see World Wide Web Consortium, 1999). Namespaces[39] are used to distinguish the definitions of features and properties defined in application-specific domains from one another, and from the core constructs defined in GML modules (OpenGIS Consortium Inc., 2002a). GML 2.1.1 defines three base XML schemas for encoding spatial information (see Figure 5-2). The Feature schema (feature.xsd) defines the general feature-property model (as feature types) and includes common feature properties such as `fid` (a feature identifier), `name` and `description`, the Geometry schema (geometry.xsd) includes the detailed geometry components, and the XLink schema (xlink.xsd) provides the XLink attributes

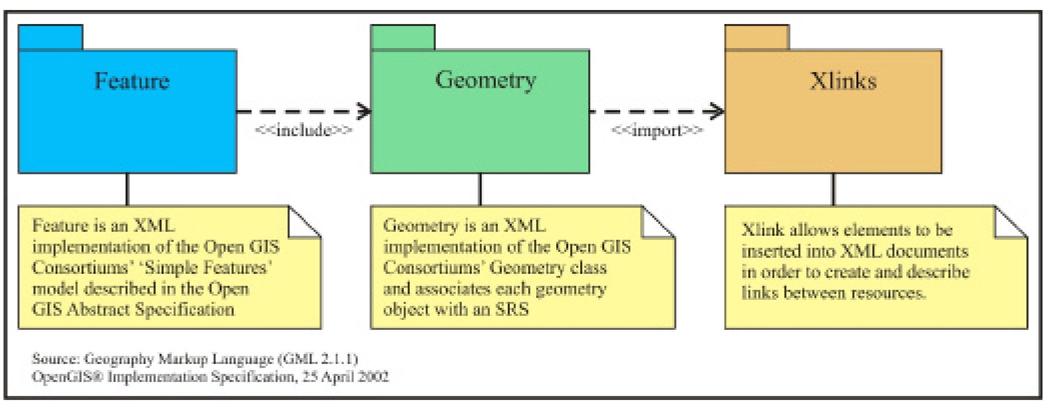

Figure 5-2: GML v. 2.1.1 Base Schemas

38 Markup is a term applied to a set of codes or tags added to the contents of a document in order to indicate its meaning or presentation

39 XML namespaces provide a simple method for "qualifying" elements as members of a particular domain, thus eliminating ambiguity. A namespace is identified by its URI reference (Uniform Resource Identifiers (URI) - a simple means for identifying a resource, e.g., http://www.ucalgary.ca/~ahunter/gml/defects.xsd). A namespace need not point to anything in particular; it is merely a way of uniquely identifying a set of elements.

used to implement linking functionality[40]. The XML Schema provides a set of primitive datatypes (e.g. string, boolean, float, month, etc.), and allows the creation of built-in and user-defined datatypes such as those provided by GML, which extends these basic types to include dates, times and addresses along with 0, 1, and 2 dimensional geometry types. The constructs used to pull together these schemas are the XML element `<include>` within the Feature schema which makes the geometry elements available for use in defining feature types, and the XML element `<import>` in the Geometry schema, which brings in the definitions and declarations contained in the XLinks schema. Figure 5-2 indicates the `<include>` and `<import>` relationships as stereotyped dependencies.

5.2.2. The GML Conceptual Framework

The current version of GML is based on XML 1.0, which is based on the notion of a "document" (Arciniegas, 2001; Lake, 2000; World Wide Web Consortium, 2001a). GML uses a `FeatureCollection` as the basis of its document. A `FeatureCollection` is a collection of GML features together with a `gml:boundedBy` element (which bounds the set of features), and a collection of properties that apply to the `FeatureCollection`. A `FeatureCollection` can also contain other `FeatureCollections`.

A feature is encoded as an XML element, as are feature instance properties, albeit at the next level in the Document Object Model (DOM) tree. In order to differentiate between a feature instance and its properties, GML adopts a uniform coding convention. Feature instances start with an uppercase letter (upper-camel-case notation) and tags that represent properties start with a lowercase letter (lower-camel-case notation); all embedded words start with an uppercase letter, e.g., `<Road>` is a feature instance, and `<fieldOperator>` is a property of `<Road>`.

---

40 An XLink linking element defines relationships between resources. A resource can be anything that is addressable on the Internet, including XML data internal to a resource. Examples include files, images, documents, programs, query results, and other schema. When a link associates a set of resources, those resources are said to participate in the link. One of the common uses of XLink is to create hyperlinks. It is the XML equivalent to the <a> tag in HTML; however it remedies a number of the shortcomings of the <a> tag, the most significant being the need to hard code both the source (<a href=…>) and the target (anchor), which makes HTML documents somewhat fragile and difficult to scale on large websites.

It is the intention of the OGC that feature definition will be left to the user to describe. However, as the OGC Abstract Specification defines a set of basic geometries, the OGC has chosen to include a number of these as elements within its Geometry schema as depicted in Figure 5-3. The Feature schema provides three levels of naming conventions for geometry properties in GML. The first are the formal names, which denote geometry properties in a manner based on the type of geometry allowed as a property value, i.e., `pointProperty` is the formal name of a `Point` geometry type. The second convention is descriptive names, which provide a set of "user-friendly" aliases for the formal names, for example the allowable descriptive names for `pointProperty` are `location`, `position` and `centreOf`. The last convention is application–specific names which are chosen by the user and defined in a GML application schema (for more details see Appendix B in OpenGIS Consortium Inc., (2002a)).

5.3. GML Application Schemas

As discussed in 5.2.1 above, three base XML schemas are provided by GML. These schema documents alone do not provide a schema suitable for constraining data instances; rather, they provide base types and structures which may be used by an application schema. An application schema declares the actual feature types and property types of interest for a particular domain, using components of GML. Broadly these involve defining application-specific types which are derived from types in the standard GML schemas, or by directly including elements and types from the standard GML schemas.

5.3.1. The Geometry Schema

The GML Geometry schema includes type definitions for abstract geometry elements, concrete `point`, `line` and `polygon` geometry elements, and complex type definitions for `GeometryCollection` types. Figure 5-3 is a UML representation of the Geometry schema. The root element of the Geometry schema is the `AbstractGeomety` element that has properties of `<gid>` and `<srsName>` which are a unique geometry identifier and a Spatial Reference System identifier. The `AbstractGeometry` element is shown as a generalization of concrete geometry elements and the

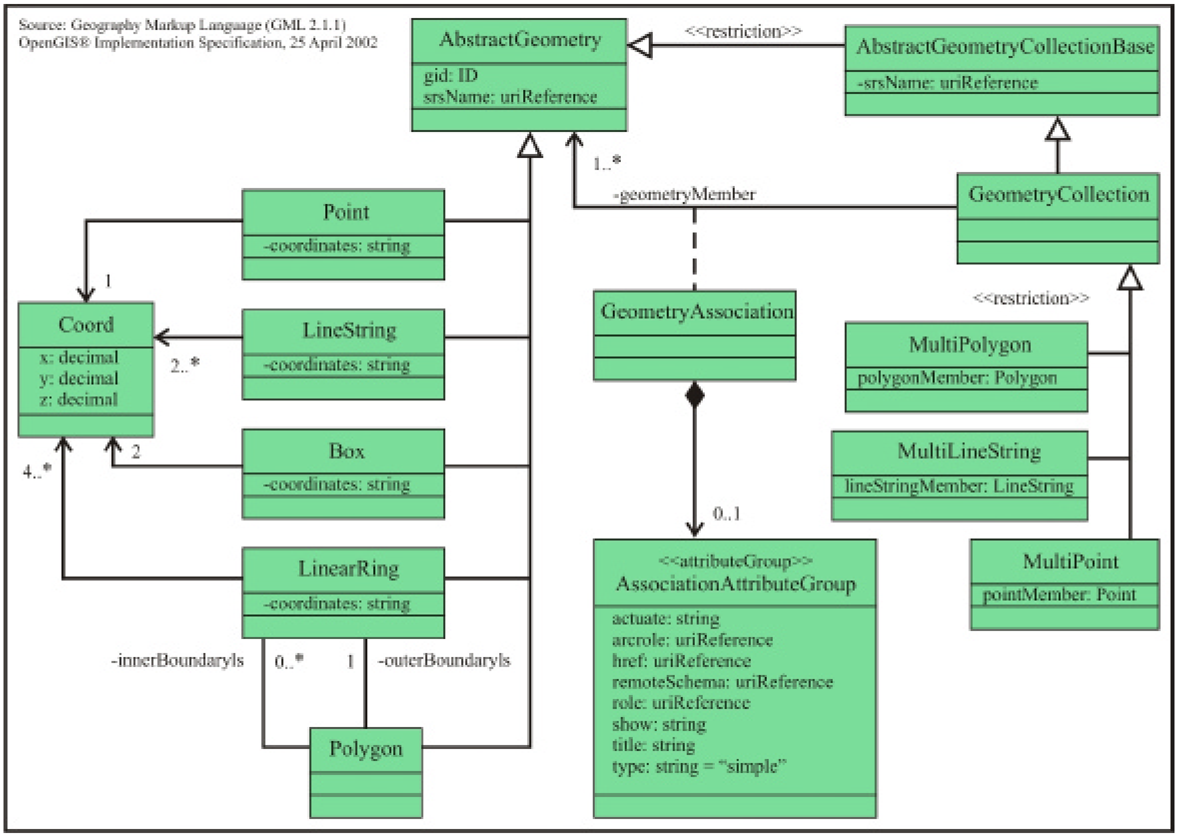

Figure 5-3: UML Representation of the Geometry Schema

AbstractGeometryCollectionBase element. Elements such as Point, LineString and LinearRing are associated with the Coord element. The multiplicity values attached to these associations indicate the number of coordinates required for each association.

The <<restriction>> stereotype applied to a generalization relationships indicates that a subtype defined in the schema is derived by restriction from its supertype. For example, a MultiLineString element is a geometry collection in which a member must be a LineString. The GeometryAssociation element is an association element of the GeometryCollection element and is composed of either zero or one AssociationAttributeGroup elements and its properties. The Geometry schema targets the "gml" namespace identified by the URI http://www.opengis.org/gml.

5.3.2. The Feature Schema

The Feature schema uses the `<include>` element to bring in the GML geometry constructs and make them available for use in defining feature types:

```
<include schemaLocation="geometry.xsd"/>
```

Figure 5-4 is a UML representation of the Feature schema. Like the Geometry schema, the Feature schema defines both abstract and concrete elements and types. The `AbstractFeature` element is the root element of the schema and contains the properties `name`, `fid`, `boundedBy` and `description`. With the Feature schema, a `GeometryProperty` is modeled as an association element so that a feature can be linked with a geometric type such as `PointProperty` or `MultiPolygonProperty`. A `BoundingShape` is also modeled as an association element as per the requirements of a `FeatureCollection` discussed in 5.2.2 above.

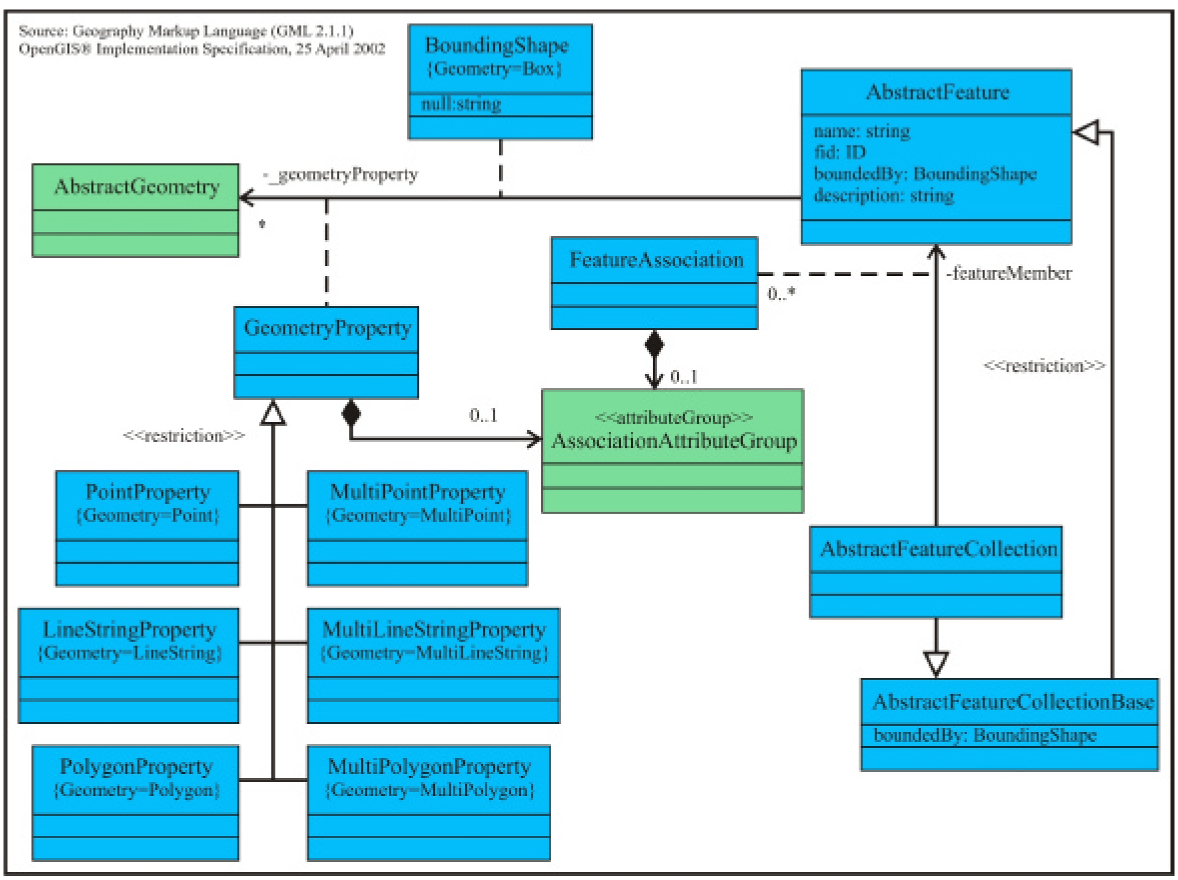

Figure 5-4: UML Representation of the Feature Schema

The abstract `GeometryProperty` element is shown as a generalization of concrete geometry types. Both the `GeometryProperty` and `FeatureAssociation` elements can be composed of zero or one `AssociationAttributeGroup` elements described in the Geometry schema. As per the Geometry schema example, the `<<restriction>>` stereotype applied to a generalization relationship indicates that a subtype defined in the schema is derived by restriction from its super type.

User communities may employ the Feature schema to develop application-specific schemas that define elements and/or types to name and distinguish significant features and feature collections from each other.

5.3.3. Geometry Elements

An essential component of a geographic system is a means of referencing the geographic features to the earth's surface or to some framework related to the earth's surface. The current version of GML incorporates an earth based Spatial Reference System (SRS) which is extensible and which incorporates the main projection and geocentric reference frames in use today. The `srsName` attribute of the geometry types are those described by the European Petroleum Survey Group (EPSG) as proposed by the OGC (OpenGIS Consortium Inc., 1999d), for example the `srsName` attribute for Calgary (Map Projection: UTM, Zone: 11, Datum: NAD83) is:

```
<gml:Box “srsName=http://www.opengis.net/gml/srs/epsg.xml#26711”>
```

All geometries must specify a SRS.

5.4. RULES FOR CONSTRUCTING APPLICATION SCHEMAS

Specifically, a conforming GML application schema must meet the following requirements: An application schema must conform to the development rules set out in § 5.2 of the GML, v. 2.1.1 specification (these requirements will be briefly reviewed in sections 5.4.1 to 5.4.7); an application schema can not change the name, definition, or data type of mandatory GML elements; an application schema must be made available to anyone receiving data structured according to the schema; and an application schema must target a namespace other than the “gml” namespace `xmlns:gml=“http://www.opengis.net/gml”` (OpenGIS Consortium Inc., 2002a)

5.4.1. Defining New Features

Any feature or feature collection defined in an application schema must be subtypes of either `gml:AbstractFeatureType` or `gml:AbstractFeatureCollectionType`, for example:

```
<complexType name=“RoadType”>
  <complexContent>
    <extension base=“gml:AbstractFeatureType”>
      <sequence>
        <!-- additional child elements are inserted here -->
      </sequence>
    </extension>
  </complexContent>
</complexType>
```

5.4.2. Defining New Geometry Types

If GML lacks an appropriate geometry type any geometry or geometry collection defined in an application schema must be subtypes of either `gml:AbstractGeometryType` or `gml:AbstractGeometryCollectionType`, for example:

```
<complexType name=“MyRoadGeometryType”>
  <complexContent>
    <extension base=“gml:AbstractGeometryType”>
      <sequence>
        <!-- additional child elements are inserted here -->
      </sequence>
    </extension>
  </complexContent>
</complexType>
```

Any user-defined geometry subtype shall inherit the elements and attributes of the base GML geometry.

5.4.3. Defining New Geometry Properties

Any geometry type or geometry collection may be encapsulated with its own properties as long as the properties are a subtype of `gml:GeometryPropertyType`, for example:

```
<complexType name="MyRoadGeometryPropertyType">
  <complexContent>
    <restriction base="gml:GeometryAssociationType">
      <sequence minOccurs="0">
        <element ref="dft:MyRoadGeometryType" />
      </sequence>
      <attributeGroup ref="gml:AssociationAttributeGroup" />
    </restriction>
  </complexContent>
</complexType>
```

An application schema may also apply a different name to a base type and use it instead, as follows:

```
<element name="potHole" type="gml:PointPropertyType"
  substitutionGroup="gml:pointProperty" />
```

5.4.4. Declaring a Target Namespace

Each application schema must have a target namespace within which all elements and their type definitions will reside. Validation of the schema will not be successful if a schema Instance document does not reside in the schema namespace. A target namespace (URI) need not point to anything concrete. A target namespace can be defined as follows:

```
<schema targetNamespace="http://www.ucalgary.ca/~ahunter/gml"
        xmlns="http://www.w3.org/2001/XMLSchema"
        xmlns:gml="http://www.opengis.net/gml"
        xmlns:dft="http://www.ucalgary.ca/~ahunter/gml"
        elementFormDefault="qualified"
        version="2.1.1">
<!-- import constructs from the GML Feature & Geometry
schemas -->
<import namespace="http://www.opengis.net/gml"
  schemaLocation="feature.xsd">
</import>
. . .
</schema>
```

5.4.5. Importing Schemas

A conforming Instance document can utilize constructs from multiple namespaces as indicated in Figure 5-5. As the Feature schema resides in the "gml" namespace along with the Geometry schema, it uses the `<include>` mechanism to access the Geometry constructs. However, as the RoadDefect schema is in a different namespace called "dft" in Figure 5-5, but must be connected to the "gml" namespace, it must therefore utilize the `<import>` element to use the GML constructs.

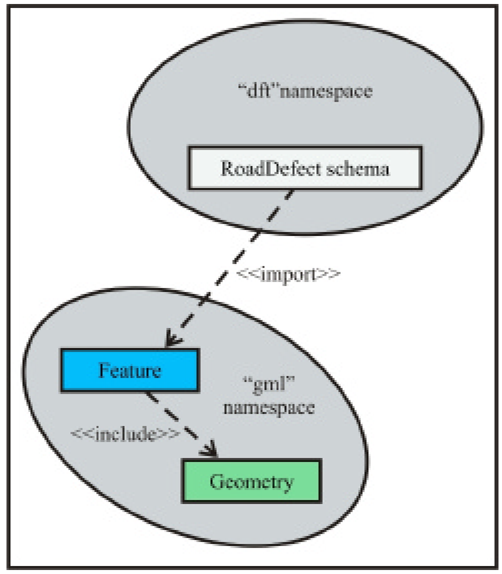

Figure 5-5: Using Schemas from multiple Namespaces

5.4.6. Using Substitution Groups

Any top-level element declaration can serve as a defining element, or head, for an element substitution group. Other top-level element declarations, regardless of target namespace, can be designated as members of the substitution group headed by this element. The following global declaration ensures that if `dft:SumpType` is a defined geometry, then a `<Sump>` can appear wherever the (abstract) `gml:_Geometry` element is expected, and is defined as follows:

```
<schema . . .>
  <element name="Sump" type="dft:SumpType"
    substitutionGroup="gml:_Geometry" />
. . .
</schema>
```

Identical elements declared in more than one complex type definition should reference a global element. If `<Sump>` is declared globally in the "dft" namespace (as shown above), it is referenced from within a type definition as follows:

```
<complexType name=“MyStormWaterType”>
  <complexContent>
    <restriction base=“gml:GeometryAssociationType”>
      <sequence minOccurs=“0”>
        <element ref=“dft:Sump” />
      </sequence>
      <attributeGroup ref=“gml:AssociationAttributeGroup” />
    </restriction>
  </complexContent>
</complexType>
```

5.4.7. Defining a New Feature Association Type

An application schema can create its own feature association types, however they must be derived from `gml:FeatureAssociationType`. The target instance must be a valid GML feature, and it may appear once (explicitly `minOccurs=“0”`, implicitly `maxOccurs=“1”`). A new feature association is implemented as follows:

```
<complexType name=“MyFeatureAssociationType”>
  <complexContent>
    <restriction base=“gml: FeatureAssociationType”>
      <sequence minOccurs=“0”>
        <element ref=“dft:MyFeatureType”
      </sequence>
      <attributeGroup ref=“gml:AssociationAttributeGroup” />
    </restriction>
  </complexContent>
</complexType>
```

Often a developer may wish to allow only certain feature types as members of a feature collection. Feature types can be restricted through the implementation of a “Feature Filter” by declaring a set of abstract elements to “label” allowable members in a feature collection. To implement a feature filter you must perform the following:

First, create a label to restrict a feature collection.

```
<!-- a label for restricting the RoadDefect Collection -->
  <element name=“_DefectFeature”
type=“gml:AbstractFeatureType”
    substitutionGroup=“gml:_Feature” abstract=“true” />
```

Next, define a filter by restricting `gml:AbstractFeatureType`

```
<xsd:complexType name=“DefectMemberType”>
  <xsd:complexContent>
    <xsd:restriction base=“gml:FeatureAssociationType”>
      <xsd:sequence minOccurs=“0”>
        <xsd:element ref=“dft:_DefectFeature”/>
      </xsd:sequence>
      <attributeGroup ref=“gml:AssociationAttributeGroup” />
    </xsd:restriction>
  </xsd:complexContent>
</xsd:complexType>
```

Lastly, label allowable features as they are declared globally:

```
<element name=“DefectMember" type=“dft:DefectMemberType”
        substitutionGroup=“_DefectFeature”/>
```

5.5. SUMMARY

There is no single way to correctly represent geospatial features. Therefore if data is to be transferred from one world-view to another there needs to be some means of translating one user community’s perception of the world to another’s without any loss of information. GML goes part way to resolving this. While it does not specifically address the issue of semantic interoperability with respect to how a common feature should be described using different world views, it does provide a means of describing a collection of features via a vendor neutral interface, thereby ensuring that loss of information does not occur. It is then up to the user to interpret the “fitness-for-use” of the data model for their geospatial data needs.

CHAPTER 6

6. *PROTOTYPE DEVELOPMENT*

This chapter discusses the mobile GIS prototype developed for this research. Each of the main components are addressed, being the systems hardware and software architecture; speech recognition; and a Road Defect GML schema. The chapter concludes with a brief discussion of a simple server architecture implemented for automated processing of data acquired in the field.

6.1. MOBILE GIS ARCHITECTURE

Figure 6-1 shows the mobile hardware required by the prototype that has been developed. The mobile GIS prototype implements the concept of real-time field-to-office data acquisition. The overall objective has been to develop a one-stop mobile survey system that simplifies the acquisition and maintenance of spatial information; a system that can meet typical user requirements in terms of positional accuracy (see Chapter 7 for more on this). The data acquisition components consist of a GPS unit for local position determination; a GPS Base Station to ensure suitable positional accuracy is obtained; a RTCM-104 capable radio to facilitate telemetry between the two GPS units; a

Figure 6-1: Mobile GIS System

computer in the form of a wearable computer from Xybernaut Corporation; a PCMCIA Wireless network card to provide a connection to the Internet via an Internet Gateway, and a server to process acquired data and store it in a centralized database. A public access Internet Gateway from Telus Mobility has been utilized. Figure 6-2 shows the mobile components of the data acquisition system implemented.

Figure 6-3 describes the software architecture that has been developed for this application. In essence there are two significant changes between this architecture and that used within a more traditional Internet based environment. The first difference is the inclusion of an intermediate interface, often called an Internet Gateway, between the mobile application and the server. The purpose of this interface is to convert information transmitted between a mobile device and a server from the carrier format used by the wireless network (CDPD in this instance) to a transfer format compatible with an Internet network, which is typically HTTP. As a wireless extension to an IP network, the Internet

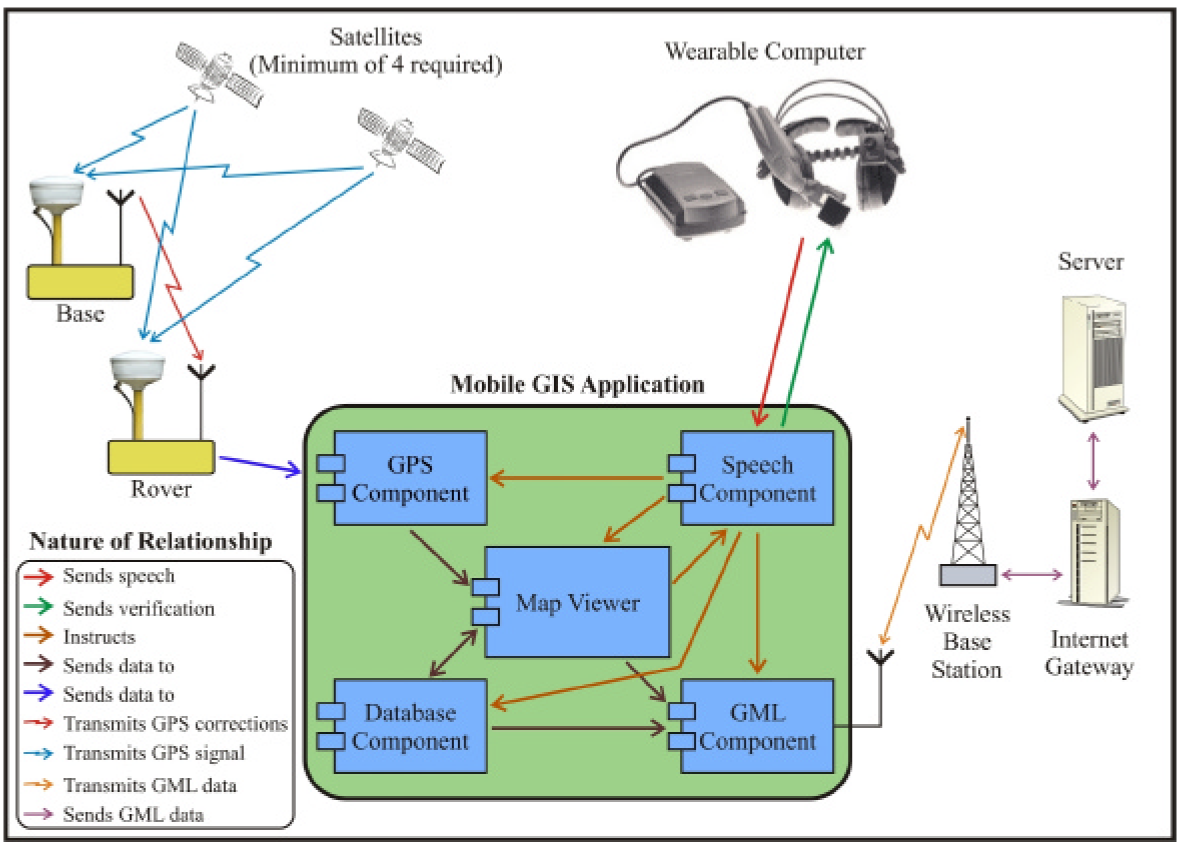

Figure 6-2: Mobile GIS Hardware Architecture

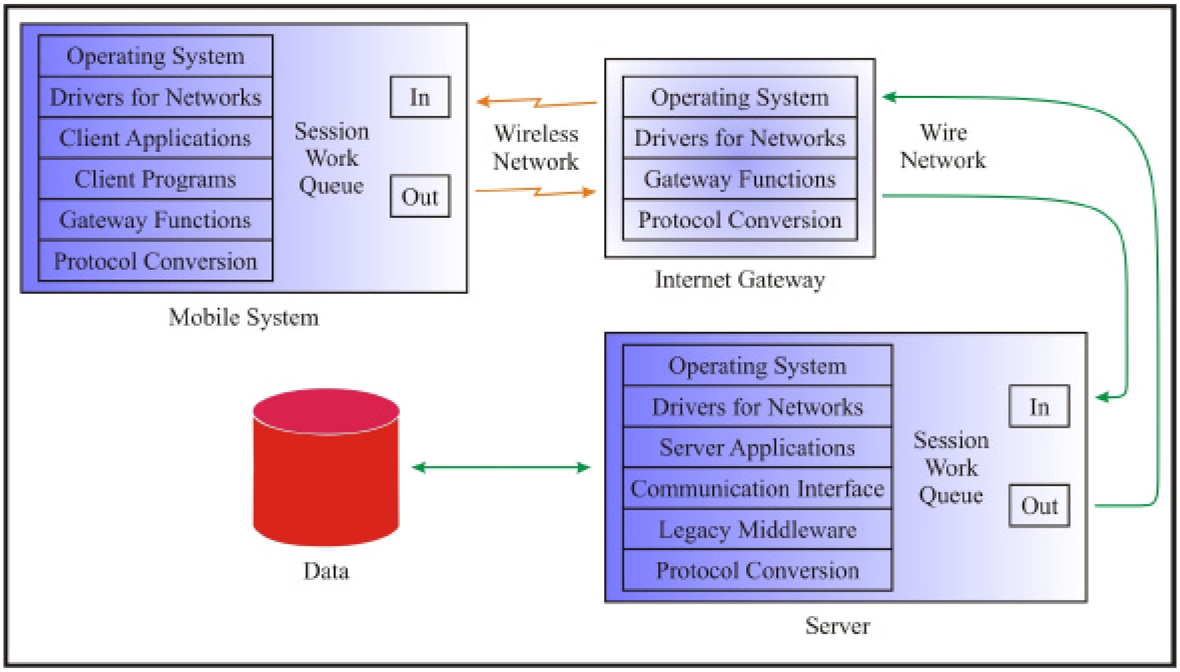

Figure 6-3: Mobile GIS Software Architecture

Gateway requires a four-octet (0.0.0.0) address for connections.

The second significant difference is the inclusion of a "Session Work Queue". One of the difficulties with wireless technology is that wireless coverage is not continuous, owing to the inability of cellular telephones to communicate with the local Cellular Base Station in certain locations even though the wireless modem/telephone is within the Base Station's coverage area. In order to ensure that data is not lost when a connection is broken, or is unable to be obtained, it is necessary to incorporate a mechanism to store the information to be sent while the mobile device obtains a new connection. Once a connection has been obtained the queue can be cleared and normal transmission of information can be resumed.

In essence the server can also be accessed via the Internet from a PC based computer or from another server connected to the Internet. The server itself has access to facilities for storing data acquired in the field. This model can also be extended to include access to application modules and other data that a user may require. However, at this stage these services have not been developed as they are not necessary for determining if the mobile GIS prototype developed can acquire spatial data adequately.

The Mobile GIS software component has been divided into five distinct components. In terms of GIS functionality the core component is the Map Viewer (see Figure 6-4). The Map Viewer has been developed in Visual Basic 6 using ESRI's MapObjects 2.1 for basic GIS functionality. Functions include the ability to pan, zoom to a layer or feature, zoom to extents, identify a feature, manipulate the cartographic display of features, add and remove feature layers, and find features based on simple SQL queries.

The speech component is the primary module for interacting with the computer. The speech component consists of three grammars. The global grammar includes commands to activate all standard functionality contained in the map viewer. There are also two Active Control grammars for managing the GPS component and data acquisition. The speech

Figure 6-4: Mobile GIS Viewer

component has been implemented using Dragon's Naturally Speaking Software Development Kit (SDK) Version 4. Essentially, Dragon's SDK sits on top of Microsoft's Speech API engine allowing access to some of its functionality. Additional details regarding the speech component and street condition survey vocabulary are described in section 6.2.

The base data shown in the map viewer was obtained from Campus Planning at the University of Calgary. The data was provided as AutoCAD R12 drawing files. No metadata was provided with the drawing files, as such, data quality information regarding lineage, completeness, spatial and attribute accuracy, and logical consistency are unknown. It is evident that the data set is not complete, as upon visual inspection, it was noted that a number of passive and active recreation areas (green polygons) were missing. However, the data set was considered adequate for this research as it was only used to assist the user in determining where they were.

The database component has been built around Microsoft Access for simplicity; however, it can be easily migrated to any other database application that allows ActiveX Data Object (ADO) connections. Figure 6-5 shows the interface for data acquisition. At the bottom of the view is a window that lists the attributes of features that have been captured. The cross-hairs show the current location. The small window shows the form used to acquire attribute information for each captured feature. All data entry can be performed via speech recognition, or key board and mouse. As features are captured they are displayed on the map using colour and symbology coding to indicate the defect type and its severity. On the mobile client side a database is maintained of all attributes for street defects that are captured during a session. This serves two purposes. The first is that the database acts as the Session Work Queue when connection to the Internet has been lost. Secondly, it provides a backup data set in the event data is corrupted while being sent to the server. The feature data is stored on the mobile client in Shapefile format and is linked to the MS Access table via unique identifier in line with typical hybrid GIS data structure models. On the server side the database component conforms to the Geographic Modeling Language (GML) Data Model for Road Defects as described in Section 6.3.

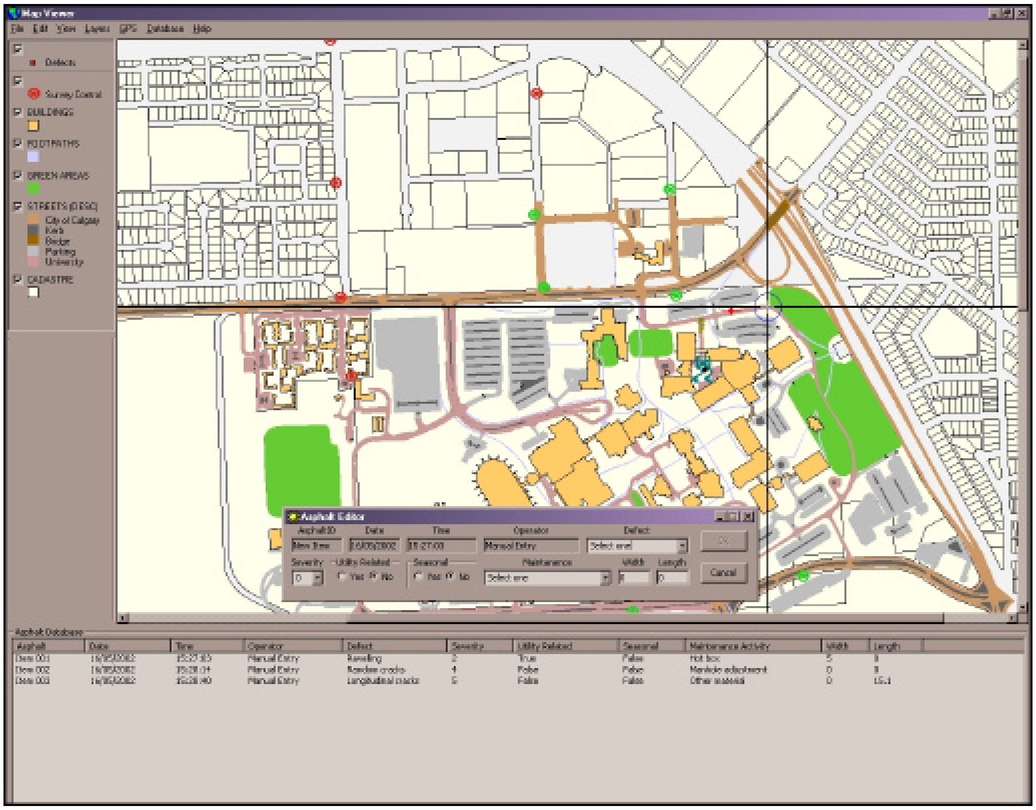

Figure 6-5: Data Acquisition Windows

The GPS component utilizes NMEA-0183 GSV and GGA messages. Position messages are passed to the mobile GIS application every two seconds Figure 6-6 shows the GPS interface used for this application. Aside from providing position information the interface also provides GPS quality factors such as Horizontal Dilution of Precision (HDOP), Position Dilution of Precision (PDOP), Vertical Dilution of Precision (VDOP), Signal to Noise Ratio for each satellite in view, and a map of the satellite geometry showing the location of each satellite, its direction from the user's current position and its elevation. Access to the Dilution of Precision quality factors depends upon the NMEA-0183 message utilized, and the NMEA messages available to the user are dependant on the type of receiver being used. For this research the GGA and GSV messages have been used as they are common to most GPS receivers, however there are a number of other messages that also provide useful information[41]. The GPS window also allows the user to define the

[41] For more information on the NMEA-0183 Standard please refer to NMEA 0183 Interface Standard, Version 3.01, published by the US National Marine Electronics Association.

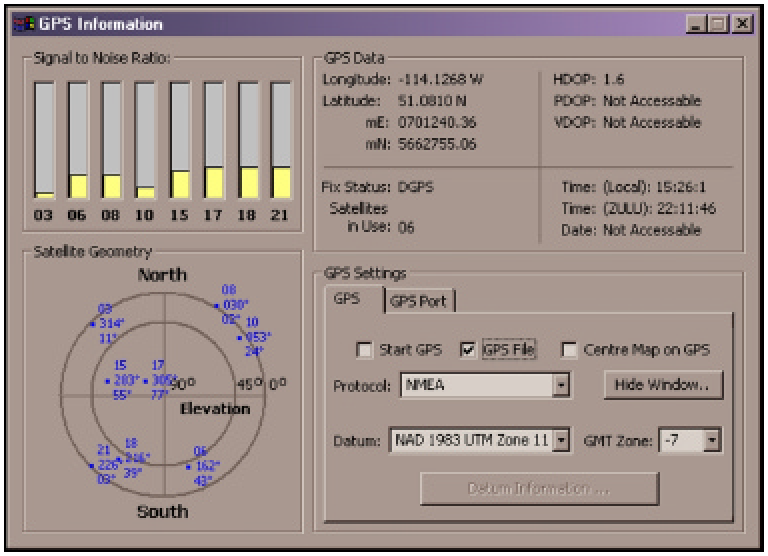

Figure 6-6: GPS Window

map projection of the coordinates for acquired GPS positions. After testing a number of acquisition rates it was determined that 0.5Hz was the highest rate that the wearable computer could process and still adequately service other processing demands such as speech recognition.

The GML component reads data from the MS Access database and reformats it to conform to the GML Schema developed for this research. Once the GML file is created a Windows Socket is opened on the client and a connection is made to the server. If the connection is successful the GML file is transmitted. The final component is the GML Server which utilises XMLDOM to read and process the GML file and is discussed in Section 6.4.

Data acquisition is performed by passing over a defect, which may be a hole in the road surface, cracking or rippling of the road surface, etc., and instructing the computer, when directly over the defect, to add a new defect feature, using the command “Add Defect”. The computer then calculates the position of the defect by interpolation based on the following ratio:

$$\left(x_{defect} = x_{GPS} + \Delta x \left(\frac{t_c - t_{GPS}}{t_{GPS+1} - t_{GPS}} \right), \quad y_{defect} = y_{GPS} + \Delta y \left(\frac{t_c - t_{GPS}}{t_{GPS+1} - t_{GPS}} \right) \right) \qquad (1)$$

where x_{defect} and y_{defect} are the interpolated coordinates of the defect, x_{GPS} and y_{GPS} are the coordinates of the last GPS position obtained prior to the issue of the “Add Defect” command, ? x and ? y are the change in x and y between the GPS position obtained prior to the issue of the “Add Defect” command and the first GPS position obtained after the issue of the command (i.e., GPS+1), t_c is the time that the “Add Defect” command was recognized, t_{GPS} is the time that the GPS position prior to the issue of the “Add Defect” command was obtained, and t_{GPS+1} is the time that the first GPS position after the issue of the “Add Defect” command is obtained. Once the position is calculated the data entry window appears and attributes can be entered orally. A field does not have to be in focus in order to be populated, nor does the data have to be entered in any particular order. Once the required fields (Defect Type, Severity, and Maintenance Type) have been populated the attributes can be saved, at which time a GML file is created, and connection to the server is attempted. If a connection can be made then the file is sent and processed by the server. If a connection cannot be made then a record is maintained of the unsent defects and the next time a connection is successful all unsent records are processed.

6.2. Street Condition Vocabulary

The vocabulary developed for this research (as shown in Table 6-1) consisted of 34 global commands which cover standard file menu features, map navigation functions (such as pan and zoom), commands for adding and removing layers and for controlling the speech recognition engine; two active control grammars that contained six and ten commands respectively were developed to control the GPS and data entry capabilities; and 133 data words for population of the Street Condition Survey database. The data words consisted largely of descriptions such as “Distortion”, “Ravelling”, “Wheel rutting” and numbers used to describe particular road surface defects that are monitored by the City of Calgary Streets Department. Global commands were activated by saying a word or phrase that intuitively represented an activity such as “Start GPS” or “Zoom to Layer”. For a complete list of Speech Recognition commands refer to Appendix E.

The system implemented one form of acknowledgement. If a command was recognized the computer passed the command to the user’s headphones. This method of acknowledgement was designed to serve two purposes; the first was to let the user know that the application was in fact responding in some way to a command; and the second was

Table 6-1: Sample of Vocabulary used for Research

Commands:

File Menu	*Commands for setting up working directories, adding layers and quitting.*
Edit Menu	*Commands to find features.*
View Menu	*Commands for map navigation, including Zoom In, Zoom Out, Zoom to Full Extents, Pan, Identify, etc.*
Layer Menu	*Commands to either remove the active layer or all layers, and access layer properties.*
Voice Menu	*Commands to activate the Audio Wizard, perform training, build vocabularies, train individual words, set the user and save speech files.*
Data Menu	*Commands to open and close data acquisition sessions.*

Active Control Grammars:

GPS Menu	*Commands to view, setup, activate, close and hide the GPS control.*
Data Entry Menu	*Commands that identify each of the fields that can be populated for each defect encountered in a survey.*

Fieldnames:

Defect	*Maintenance*	*Severity*	*Utility*	*Seasonal*	*Dimension*
Distortion, Rippling, Ravelling, Random, Cracks, etc.	*Hot box, Hand crew top, Hand crew base, Paver, Crack Sealing, etc.*	*Rated 1 to 5, 5 being worst*	*Yes, No*	*Yes, No*	*Number, 0 to 100m*

to provide a means of verifying the data that was being entered into the database. In essence this process replicates a traditional method of quality control where another person reads the data entered into a system back to the person who entered it, who then verifies what they have heard is in fact what they were supposed to have entered. Thus, verification can be considered independent of data entry as another person verifies the data that was entered. In certain circumstances the response message also requested confirmation of a command so as to ensure that critical actions were in fact intended. By ensuring that data fields were restricted to certain values it was possible to make sure errors were minimized with respect to data being placed in the wrong field, thereby improving data integrity.

6.3. Road Defect Schema Implementation

Figure 6-7 is a Universal Modeling Language (UML) diagram for the Road Defect data model. The `RoadDefectsModel` is the primary feature collection for this model and is a specialization of the `AbstractFeatureCollection` described in OGC's GML

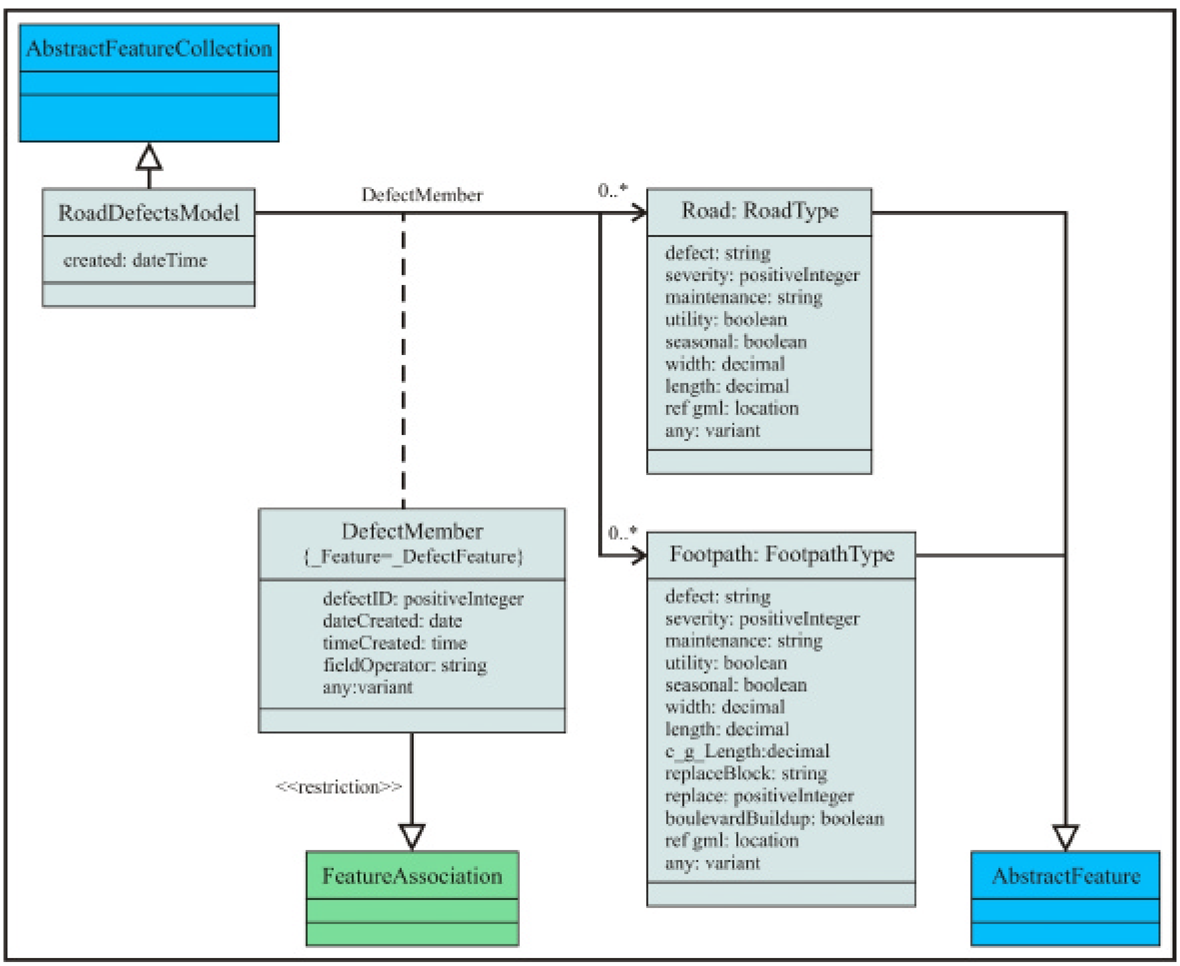

Figure 6-7: UML Diagram of the Road Defect Data Model

Feature Schema, feature.xsd. The RoadDefectModel has one property called `created` of type `dateTime`. The `DefectMember` acts as a Feature Filter as described in Chapter 5, Section 5.4.7. The filter allows instances of feature types Road or Footpath. This is modeled in Figure 6-7 by the <<restriction>> stereotype applied to a generalization relationship between the FeatureAssociation element (described in the OGC's GML Geometry Schema, geometry.xsd) and the DefectMember element. The DefectMember also consists of four properties describing when and by whom each defect was captured. Road and Footpath are specializations of `AbstractFeature` (described in feature.xsd) and each contain a number of properties that meet the requirements of the City of Calgary Street Condition Survey. For this model all features are defined as points and are included in the model by reference to the geometric property `ref gml:location`. `DefectMember`, `Road` and `Footpath` each include a property called `any`. In essence this property allows the data model to be extended by another user at any time. A `RoadDefectModel` instance may consist of zero or more `Road` or `Footpath` defects.

Four namespaces have been declared in this GML schema and are depicted in Figure 6-8 along with the majority of the elements and types used or created in each namespace. Table 6-2 lists the header of this schema. The target namespace is `http://www.ucalgary.ca/~ahunter/gml` and is the namespace within which the `RoadDefectModel` elements and types are created. This is also the default namespace.

Table 6-2: RoadDefectModel Schema Header

```
<xsd:schema
targetNamespace="http://www.ucalgary.ca/~ahunter/gml"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns="http://www.ucalgary.ca/~ahunter/gml"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:dft="http://www.ucalgary.ca/~ahunter/gml"
  elementFormDefault="qualified"
  version="2.1.1"
  xml:lang="en">
```

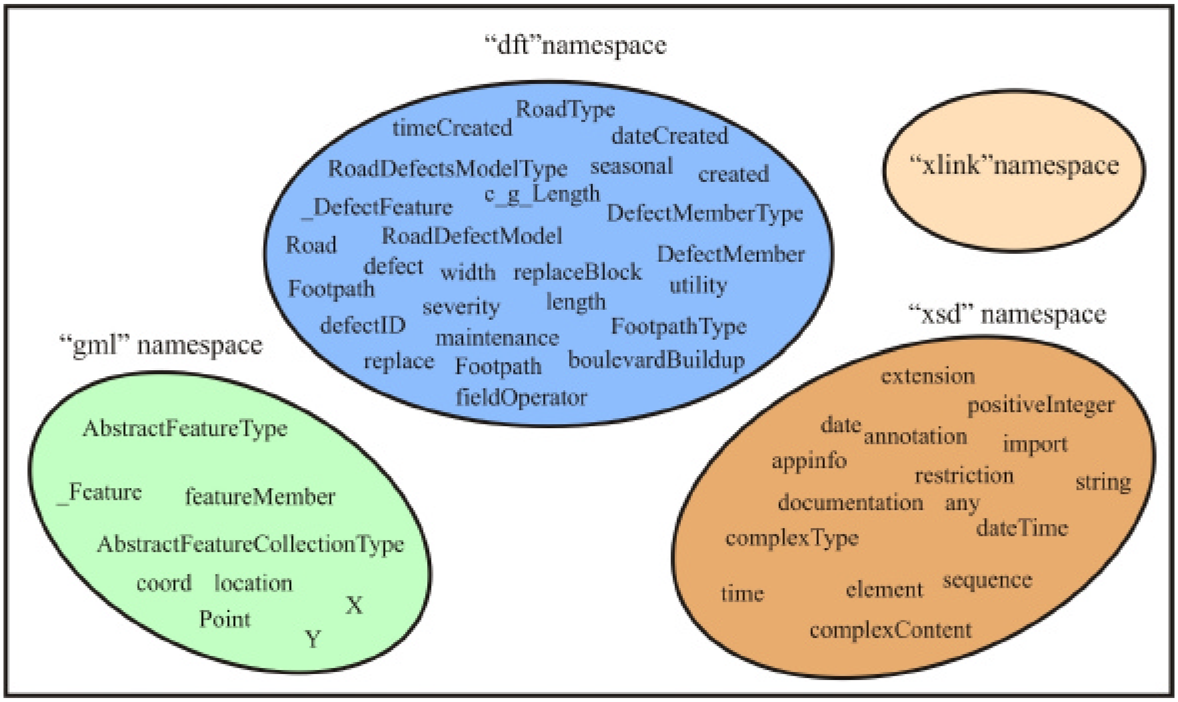

Figure 6-8: Road Defect Model Namespace Organization

By organizing the header in this manner, the namespace from which each element or type belongs must prefix each element/type and therefore provides additional clarity for the reader with respect to schema organization. The "xlink" namespace is created as part of the GML framework, but has not been used in this particular schema, hence the empty "xlink" namespace in Figure 6-8.

Figure 6-7 shows `DefectMember` being described by four properties or elements, being `defectID`, `dateCreated`, `timeCreated`, and `fieldOperator`. Of these properties `defectID` acts as a unique identifier for each defect. By utilizing XML's `key` element we can ensure that each `DefectMember` contained in an Instance document is unique. The implementation of this feature is shown in Table 6-3. A number of properties for both the `Road` and `Footpath` features consist of a defined list of options from which one can be selected. If the OGC Simple Features rules are applied few restrictions can be built into an application to restrict the data that can be captured, essentially a string property could be populated with any text based data.

Table 6-3: Unique Identifier Implementation

```
<xsd:element name="defectID" type="xsd:positiveInteger">
  <xsd:key name="dftKey">
    <xsd:selector xpath=".//defectID"/>
    <xsd:field xpath="defectID"/>
  </xsd:key>
</xsd:element>
```

However, by employing the XML constructs such as `restriction` and `enumeration` much greater control over the data that can be entered into a GML file can be built into an application. Table 6-4 lists an implementation of `DType` being a `restriction` on the `string` type which insures that the Defect type element is populated with valid defect descriptions that are of type `string`. A similar process can restrict the population of elements with integers between one and five, or decimal numbers between 0 and 100 shown to one decimal place. To review the GML schema in detail, refer to Appendix F.

Table 6-4: A String Restriction Implementation

```
<xsd:simpleType name="DType">
    <xsd:annotation>
      <xsd:documentation>
        Defects allowed on a carriageway, curb or footpath.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="Distortion"/>
      <xsd:enumeration value="Rippling"/>
      <xsd:enumeration value="Ravelling"/>
      <xsd:enumeration value="Random cracks"/>
      <xsd:enumeration value="Longitudinal cracks"/>
      <xsd:enumeration value="Wheel rutting"/>
      <xsd:enumeration value="Excessive patching"/>
      <xsd:enumeration value="Alligatoring"/>
      <xsd:enumeration value="Transverse cracks"/>
      <xsd:enumeration value="Sheet asphalt overlaid"/>
      <xsd:enumeration value="Tripping edge"/>
      <xsd:enumeration value="Catch basin displacement"/>
      <xsd:enumeration value="Cracks"/>
      <xsd:enumeration value="Crumbling"/>
    </xsd:restriction>
  </xsd:simpleType>
```

6.3.1. Road Defects Instance

Appendix G lists a simple schema-valid Instance document that conforms to Defects.xsd (Appendix F). The explicit reference to “defects.xsd” in the root element of the Instance document (i.e. the value of the xsi:schemaLocation attribute) is not required, but it does provide a hint to a Validating Parser[42] regarding the location of the relevant schema document. Both the schema and the Instance document for this Road Defect Model were validated using XML Spy v4.4 U (http://www.xmlspy.com).

The `RoadDefectModel` element is the root element of this Instance document and is the `FeatureCollection` within which all `DefectMembers` contained in the intance reside. As required by the Feature schema, the first element to follow the root element is `gml:boundedBy` (see Table 6-5 below for details), which is a `gml:Box` element defining the spatial extent of the data contained in the Instance. `gml:Box` consists of two `gml:coord` pairs defining the lower left and upper right corners of the `gml:Box`. The `gml:Box` geometry is expressed in the spatial reference system identified by the value

Table 6-5: An Instance of the gml:boundedBy Element

```
<gml:boundedBy>
  <gml:Box srsName=
      "http://www.opengis.net/gml/srs/epsg.xml#26711">
    <gml:coord>
      <gml:X>700380.875875919</gml:X>
      <gml:Y>5662672.84569231</gml:Y>
    </gml:coord>
    <gml:coord>
      <gml:X>700385.673562891</gml:X>
      <gml:Y>5662673.59199227</gml:Y>
    </gml:coord>
  </gml:Box>
</gml:boundedBy>
```

[42] A Validating Parser checks that a document is well formed, that is it follows the rules of the XML specification, and that the elements within it are correctly nested and part of a unique root. A Validating Parser also checks that the document conforms to the Markup declarations of a provided schema, in this case xsi:schemaLocation=“http://www.ucalgary.ca/~ahunter/gml/defects.xsd”.

of the srsName attribute: this URI reference points to a fragment in another XML document that contains information about reference system described by the European Petroleum Survey Group. The first feature member is an instance of Road representing a Defect "Distortion" of Severity "3", etc. The member has a defectID of "1385" and contains other properties describing the date and time the feature was captured and by whom. It has a geometric property called gml:location with a gml:point value (see Table 6-6). The gml:point geometry is expressed in terms of the same spatial reference system used by the gml:Box.

Table 6-6: An Instance of a gml:Location Element

```
<gml:location>
  <gml:Point srsName=
      "http://www.opengis.net/gml/srs/epsg.xml#26711">
    <gml:coord>
      <gml:X>700380.875875919</gml:X>
      <gml:Y>5662673.59199227</gml:Y>
    </gml:coord>
  </gml:Point>
</gml:location>
```

The second feature member is an instance of FootpathType representing a Defect "Tripping edge" of Severity "5", etc. As with the RoadType member it also has a geometric property called gml:location with a gml:point value and a series of properties describing the feature and when it was captured. The final element in the Instance document is created which is the date and time that the GML file was created in XML's dateTime format, i.e., <created>2002-05-26T15:23:57</created>.

6.4. A Basic GML Server

The Windows Sockets programming interface allows a client, or mobile computer, to connect to a remote machine and exchange data using either the User Datagram Protocol (UDP) or the Transmission Control Protocol (TCP). For this research TCP has been implemented as it allows the creation and maintenance of a connection to a remote computer, which can then be used to stream data between them.

Other communication technologies such as SOAP[43], SSH Secure Shell[44] and Windows 2000 Virtual Private Networking[45] were investigated. However, because of the standardized format of the data being sent over the wireless network and the Internet, the additional capabilities of these technologies, as opposed to a traditional Windows Sockets connection, did not warrant the added overhead that results from these more complex network protocols, nor does it appear that the added complication of setting up these technologies is warranted for this type of application. If security of information is an issue then simple encryption technologies can be invoked within the Visual Basic environment at less cost to the system. Another simple alternative is to substitute the GML tags for tags that would be indecipherable to anybody watching when data is being transferred, e.g., the `<defect>` tag could be substituted with a tag called `<purpleYogi>` by the mobile prototype as could the data that it contains. The server could then replace these with the correct data during processing.

However, if the format of the data that is to be sent from the field is unknown then technologies such as SOAP would likely be a better option, as SOAP applications, for example, can search out XML/GML Schemas posted on remote servers and format the data to be transmitted accordingly so that the data can be readily understood by the server application (Microsoft Corporation, 2002d). In effect they are somewhat more intelligent than the TCP/IP protocol implemented for this application, but the extra intelligence does not appear to be warranted as the information being transferred between the mobile client and the server is strictly controlled.

Microsoft's XML Document Object Model (DOM) and XML Parser (Msxml.dll) are

---

[43] SOAP is a lightweight protocol for the exchange of information in a decentralized, distributed environment. It is an XML-based protocol that consists of three parts: an envelope that defines a framework for describing what is in a message and how to process it, a set of encoding rules based on XML constructs for expressing instances of application-defined data types, and a convention for representing remote procedure calls and responses. If you have a well-formed XML fragment enclosed in a couple of SOAP elements, you have a SOAP message.

[44] Secure Shell is a technology used to secure TCP connections over the Internet by encrypting all transmitted data. It is typically used to secure FTP and Telnet connections. SSH provides data security via PGP (Pretty Good Privacy), provides system security by allowing only one secure point of entry to your network, and provides network security by preventing others from sniffing your network traffic (network wiretapping) or high jacking (taking over) your session via various authentication processes.

[45] A Virtual Private Network (VPN) enables you to send data between two computers across a shared network or the Internet in a manner that emulates a point-to-point private link. Data sent over the VPN are encrypted for confidentiality.

the basis of the server application. DOM for XML is an object model that exposes the contents of an XML document[46]. Microsoft's implementation of the DOM fully supports the W3C standard (Microsoft Corporation, 2002c) and has additional features that make it easier to work with XML files within the Visual Basic environment. Once an XML document has been read by the Parser, data can be extracted from the file by traversing the document tree (Microsoft Corporation, 1999). XML documents are hierarchical by nature, which makes it relatively easy to traverse a document. At this stage all data sent to the server is stored within a Microsoft Access database, the schema of which has been derived from the Defects Schema (Refer to Appendix F). Figure 6-9 provides a graphical interpretation of the hierarchical Document Object Model for a Road Defect Instance document as determined by XML Spy (http://www.xmlspy.com).

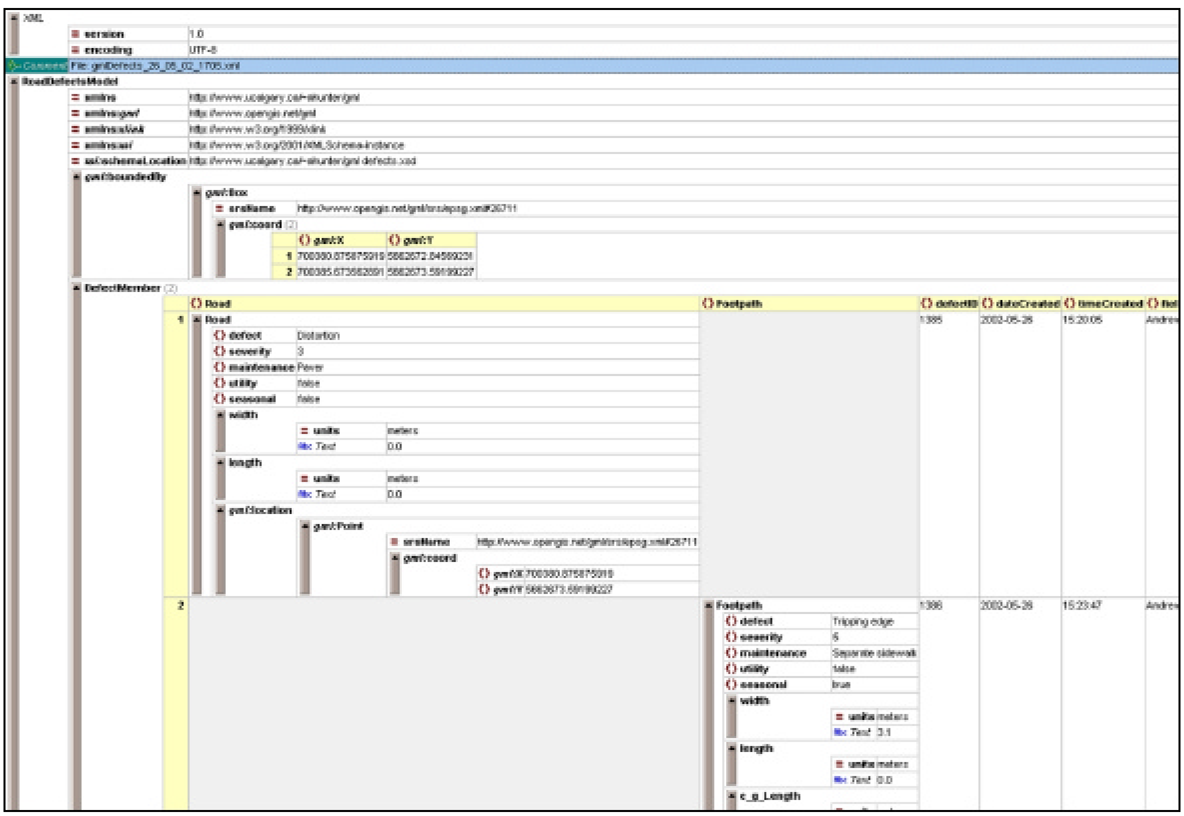

Figure 6-9: Road Defect Instance Document Object Model

46 The W3C's Document Object Model (DOM) Level 1 Specification (World Wide Web Consortium, 2000) currently defines what a DOM should expose as properties, methods, and events.

The Document Object Model commences with an XML declaration, in this case stating that the document conforms to XML version 1.0 and is encoded using UTF-8[47]. The next section is the `RoadDefectModel` Feature Collection. The initial attributes list the namespaces used by the document and the location of the validating schema. The attributes are followed by the `gml:boundedBy` branch showing the extent of the Instance document. The final branch, or child element, of the root is the `DefectMember` element, which itself branches off into a number of `Road` and `Footpath` child branches depending on the content of the Instance document.

6.5. SUMMARY

A mobile GIS data acquisition application for a wearable computer has been developed in Visual Basic using ESRI's MapObjects v.2.1 ActiveX component for GIS functionality. Dragon NaturallySpeaking ActiveX components have been integrated so as to allow speech recognition as the primary mode of interaction with the computer. The speech recognition component was developed using a multi grammar vocabulary so as to improve recognition performance. Verification of spoken commands was provided via audio feedback from the computer to the user. The data acquisition prototype incorporates a real-time kinematic GPS component for location determination. The GPS component implements the NMEA-0183 interface as for the transfer of position data from the roving GPS to the mobile GIS prototype. A GML schema for road defects has been developed to conform to the City of Calgary's Streets Department data acquisition requirements. In order to provide real-time, or near real-time data acquisition a wireless component has also been integrated that provided CDPD based access to Telus Mobility's wireless network. Lastly, a server, based around TCP/IP protocols and Microsoft's implementation of W3C's Document Object Model, has been developed providing a simple and robust environment in which to process well formed GML Instance documents.

---

47 UTF-8 encoding is defined in ISO 10646-1:2000. It preserves the full US-ASCII range, providing compatibility with file systems, parsers and other software that rely on US-ASCII values. That is, it is a universal encoding system recognised by all operating systems.

CHAPTER 7

7. TESTING AND ANALYSIS

This chapter reviews the testing methodology undertaken for this research and presents the findings. The analysis has been split into three sections. The first consists of a Data Accuracy Requirements survey that was undertaken to determine what the current status of data acquisition is in terms of accuracy and time to use, and what GI users actually desire. The second section deals with speech recognition testing where the reliability of speech recognition was determined in both office and field environments. The final test addresses accuracy, the objective of which was to determine positional accuracy while capturing data via different modes of transportation, namely standing, continuous walking, cycling and driving a vehicle.

7.1. DATA ACCURACY REQUIREMENTS SURVEY

By mid-October 2001, 80 responses had been received from GITA, URISA, AURISA, NZ Local Government Online, New Zealand Institute of Surveyors, the NZ ESRI User Group and the GISList. But is the sample representative of the targeted population? By its nature, Internet-based surveys are very attractive, but the biggest concern in Internet surveying is coverage bias, or bias due to sampled people not having, or choosing not to access the Internet (Kaye *et al.*, 1999; Crawford *et al.*, 2001). Despite exponential growth of the Internet there are still large numbers of people who do not have access and/or choose not to use the Internet, which can create problems in guaranteeing a random sample of respondents. By its nature, the Internet poses a unique set of problems in guaranteeing a random sample of respondents. Unlike telephone and mail surveys in which samples can be produced through random digit dialling of census lists, the Internet has no central registry of users (Kaye *et al.*, 1999). However, because the survey was directed towards a highly specific population, who must be assumed to be interested in the use of GIS and spatial data because they belong to these newsgroups and listservers, and because the responses were voluntary in that they were self selected, it was anticipated that the responses would be representative of the population.

Typically the response rate of a survey is an indicator of how representative a survey is (Babbie, 1990). However, the response rate of an Internet survey cannot be calculated because there is no way in which to know how many individuals might have seen the survey or its links but declined to participate. Only the number of completed surveys is known and not the number of refusals, because there is no record of how many email addresses are incorrect, or how many people actually check their email during the survey period. However, while the exact number of recipients can not be determined, contact with each of the bulletin boards/listservers indicates that at least 3257 emails were sampled giving a maximum response rate of 2.5%. As such, the response rate is low, although this is to be expected according to research by Crawford *et al.*, (2001).

The majority of the responses were from Canada (22) and the US (29), although 17 responses were obtained from New Zealand and 9 from Australia. The respondents were grouped into functional groups: AM/FM, Electrical and Gas (12 respondents), Business GIS (5 respondents), Environment (10 respondents), Local Government (36 respondents), Water Resources (4 respondents), and others (13 respondents).

7.1.1. Spatial Data Accuracy

One of the goals of this research was to develop a speech aware spatial data acquisition tool that could capture data at an accuracy of better than one metre. Thus the respondent's data was conflated into groups of who do meet, or who would like to meet this requirement, and those who don't. Then three different criteria (actual accuracy, practical accuracy[48], and desired accuracy) were compared using 2 x 2 Contingency tables and the $?^2$ test of independence in order to determine if higher accuracy data is a necessity when cost is a significant factor in determining a user's data accuracy requirements. The Null Hypothesis is that cost of data acquisition does not affect the accuracy demands of a user.

Table 7-1 and Table 7-2 below summarize and compare the responses in each of the

---

48 Practical accuracy was defined as the accuracy specification considered ideal for an application when cost of data acquisition is a consideration.

three categories analyzed. In order for the Null Hypothesis to be refuted, the calculated ?^{2} value must be greater than the critical value of ?^{2} being 3.841 (a = 0.05, d.f. = 1). Expected frequencies have been included (in bold text and underlined) in the tables below the observed frequencies.

Table 7-1: Actual and Practical Accuracy Requirements

	Actual Accuracy	*Practical Accuracy*	*Total*
Accuracy < 1m	*34*	*45*	*79*
	39.5	***39.5***	
Accuracy > 1m	*46*	*35*	*81*
	40.5	***40.5***	
Total	*80*	*80*	*160*

?^{2}	*2.500*
P	*0.114*

Table 7-2: Actual and Desired Accuracy Requirements

	Actual Accuracy	*Desired Accuracy*	*Total*
Accuracy < 1m	*34*	*63*	*97*
	48.5	***48.5***	
Accuracy > 1m	*46*	*17*	*63*
	31.5	***31.5***	
Total	*80*	*80*	*160*

?^{2}	*20.527*
P	*0.000*

Analysis of the data shows that there is no significant difference between the accuracy of data currently being used and end-users' practical accuracy requirements as the calculated ?^{2} value (?^{2} test with Yates' Continuity Correction = 2.500, *P* = 0.114, d.f. = 1) is less than the critical value of ?^{2} and the Null Hypothesis is therefore not rejected. However, comparing the respondents who already use data at an accuracy of less than one metre (with cost as a constraint), to the respondents who would like data with an accuracy of less than one metre if cost was not an issue, i.e., desired accuracy, it is evident that respondents desire more accurate data as the calculated ?^{2} value (?^{2} test with Yates' Continuity Correction = 20.527, P = 0.000) is greater than the critical value of ?^{2} and the Null Hypothesis is rejected.

The first test indicates that users accept the accuracy of the spatial data that they have, which is in agreement with personal experience. For example, while capturing base topographic data between 1995 and 1998 for the Government of Brunei Darussalam, we were expected to capture data from separations obtained from aerial photography that was flown between 1972 and 1986. Because of the lack of currentness of the source data there were a significant number of infrastructure features that could not be included in the spatial databases. Similar circumstances were also encountered in New Zealand.

Based on the responses obtained one would expect 60.6% (48.5 out of 80 responses) of the respondents to demand more accurate data regardless of cost, whereas 78.8% (63 out of 80 responses) of the responses were actually obtained. The second test therefore, clearly states that the cost of data acquisition and the positional accuracy of that data are significantly and positively associated.

7.1.2. Time-to-Use Requirements

With regards to Time-to-Use (TtU) requirements, Respondents were provided with a number of options ranging from 1 day or less through to 6 months or more, and were asked to indicate actual time-to-use and preferred time-to-use for their organization. Seventeen respondents (see Figure 7-1) are currently using data within one day; the majority (61) are using their data within one month. In terms of preferred time-to-use (see Figure 7-2), 32 respondents would like their data within one day and the majority (68) would like to use their data within 2 weeks.

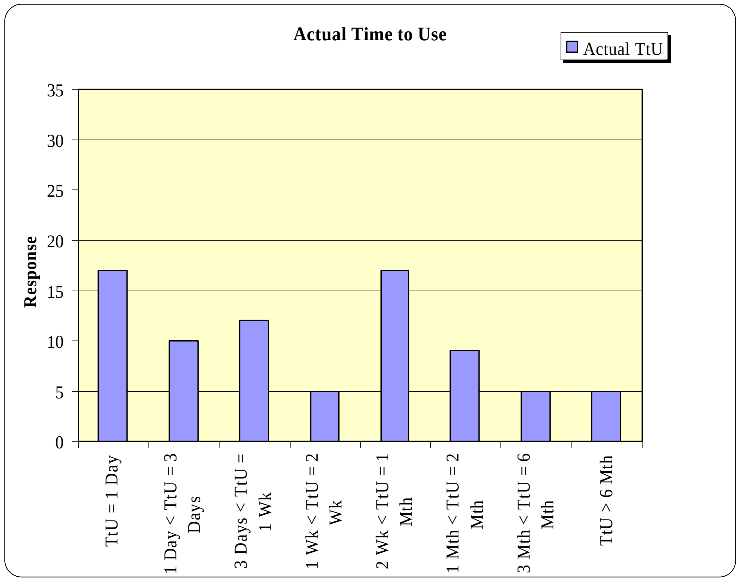

Figure 7-1: Actual Time to Use Requirements

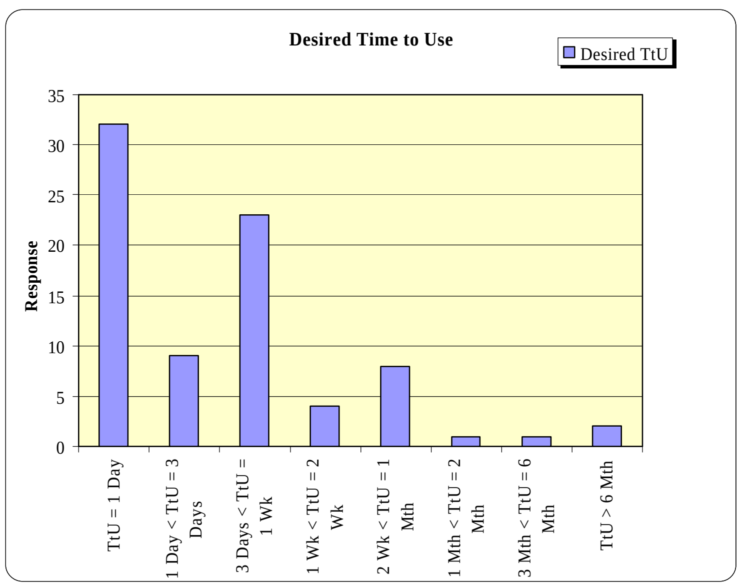

Figure 7-2: Desired Time to Use Requirements

An initial contingency table analysis of this data was undertaken to determine if there was any connection between users' actual TtU requirements and their desired TtU requirements. The Null Hypothesis for this analysis is that there is no difference between the time that it currently takes user's to acquire their geographic data and their desired time-to-use requirements. As such, the critical ?^{2} value is 14.067 (a = 0.05, d.f. = 7). The calculated ?^{2}

value was 21.805 (P = 0.003), which is greater than the critical ?^{2} value for this test. Therefore the Null Hypothesis is rejected that users are satisfied with the speed with which they can obtain and make use of data. It should be noted that six of the 16 expected frequencies were less than five, which implies that the result should be interpreted with some scepticism.

In order to verify this result the data was collapsed into 6 groups as described in Table 7-3 and the analysis was repeated. In this instance the calculated ?^{2} value was 21.212 (P = 0.001), which is greater than the critical ?^{2} value for this test, being 11.070 (a = 0.05, d.f. = 5). Therefore we can be confident that the Null Hypothesis is rejected, and that users would like to decrease the time that it takes them to obtain and make use of data.

Table 7-3: Actual Time to Use v. Desired Time to Use

Time Frame	*Actual TtU*	*Desired TtU*	*Total*
TtU =1 Day	*17*	*32*	*49*
	24.5	***24.5***	
1 Day < TtU =3 Days	*10*	*9*	*19*
	9.5	***9.5***	
3 Days < TtU =1 Wk	*12*	*23*	*35*
	17.5	***17.5***	
1 Wk < TtU =1 Mth	*22*	*12*	*34*
	17	***17***	
1 Mth < TtU =2 Mth	*9*	*1*	*10*
	5.0	***5.0***	
TtU > 6 Mth	*10*	*3*	*13*
	6.5	***6.5***	
Total	*80*	*80*	*160*

?^{2}	*21.212*
P	*0.001*

However, while the survey indicates most users would like data to be obtained more rapidly, what is of interest, with respect to this research, is which time frame is the most desirable to geospatial data users with regards to the acquisition and use of that data. To determine what the most desirable time was the data was collapsed into two groups, those who get, or want, data within a particular timeframe, and those who do not. The ?^{2} test was then used to compare the collapsed Actual and Desired TtU responses over a number of time frames, the results of which are summarized in Table 7-4. For these tests the critical ?^{2} value was 3.841 (a = 0.05, d.f. = 1)

Table 7-4: Comparison of Time to Use Requirements

Time Frame	?^{2}	*P*
Less than or Greater than 1 day	*5.766*	*0.016*
Less than or Greater than 3 days	*4.322*	*0.038*
Less than or Greater than 1 week	*15.698*	*<0.001*
Less than or Greater than 2 weeks	*15.744*	*<0.001*
Less than or Greater than 1 month	*9.952*	*0.002*
Less than or Greater than 2 months	*3.014*	*0.083*

The most desirable timeframe, as indicated by the greatest change between actual and desired time-to-use, is somewhere between one week and two weeks as indicated by the largest ?^{2} values and the smallest *P* values.

A comparison of Actual versus Desired TtU for counts less than or greater than one day and less than or greater than three days returns calculated values (?^{2} test with Yates' Continuity Correction = 5.766 (P = 0.016) and 4.322 (P = 0.038), d.f. = 1) that are greater than the critical value of ?^{2} therefore the Null Hypothesis is rejected that users are satisfied with the speed with which they can obtain and make use of data.

A comparison of Actual versus Desired TtU for counts less than or greater than two months returns a calculated value (?^{2} test with Yates' Continuity Correction = 3.014, $P <$ 0.083, d.f. = 1) that is less than the critical value of ?^{2} therefore the Null Hypothesis is not rejected for this time frame indicating that there is no evidence that Actual and Desired TtU demand are independent for this level.

A comparison of Actual versus Desired TtU for counts less than or greater than one week, two weeks and one month return calculated values (?^{2} test with Yates' Continuity Correction = 15.698 (*P* < 0.001), 15.744 (*P* < 0.001) and 9.952 (*P* = 0.002), d.f. = 1) that are greater than the critical value of ?^{2} therefore the Null Hypothesis is rejected in all three cases. They are also greater than the critical value of ?^{2} = 6.635; a = 0.01; d.f. = 1, therefore the Null Hypothesis is rejected at this level of confidence also.

When we compare expected frequencies with the observed counts we see that the greatest differences occur at the one week level, closely followed by the two week period (see Figure 7-3). This suggests that the most desirable time within which data is to be acquired with respect to a Local Government environment is within a time frame of one to two weeks. If the respondents are grouped into Local Government and "Others", we see that "Others" have a greater preference for two weeks or less (for one week or less ?^{2} test with Yates' Continuity Correction = 6.982, *P* = 0.008, d.f. = 1 as opposed to a ?^{2} test with Yates' Continuity Correction = 11.939, *P* < 0.000, d.f. = 1 for two weeks) and Local Government has a greater preference for one week or less (for one week or less ?^{2} test with Yates' Continuity Correction = 7.563, *P* = 0.006, d.f. = 1 as opposed to a ?^{2} test with Yates' Continuity Correction = 3.863, *P* < 0.049, d.f. = 1 for two weeks).

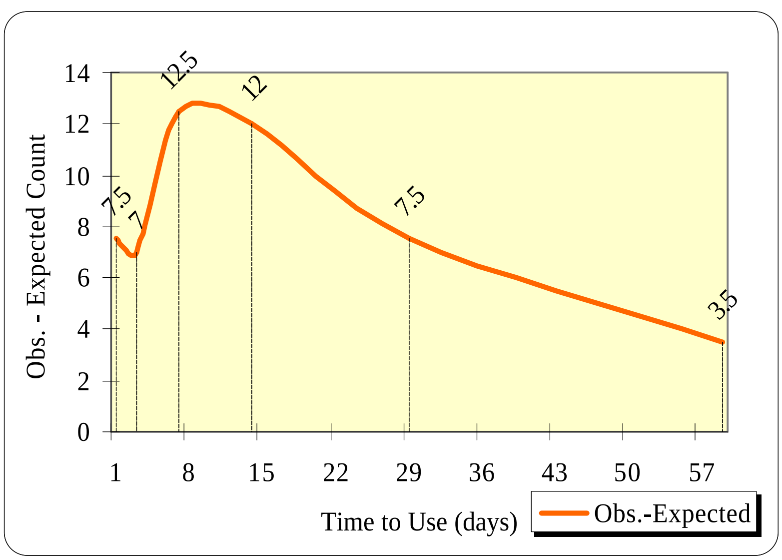

Figure 7-3: Observed/Expected Time to Use Differences

7.1.3. Other Results

Half of the respondents use conflation[49], of which 53% consider the results acceptable. Only 60% of the respondents undertake any form of validation, 53% digitize maps in order to acquire data, and 54% are not satisfied with the quality of their data. These figures indicate that there continues to be a considerable amount of spatial information acquired by methods which make it difficult, if not impossible, to ensure that end user spatial data accuracy requirements are met. With respect to litigation as a result of harm that is caused, or economic loss sustained (Onsrud, 1999), from the sale or supply of spatial data for a purpose for which it is not fit, it is surprising that only 60% of the respondents actually validate the data that they capture, given the potential of poor quality data to harm reputations. For example, relatively simple mistakes can have disastrous consequences when people depend on a map or chart for accurate representations of the real world. In Reminga v. United States, 695 F. 2d 1000 (6th Cir. 1982), the government was held responsible for an airplane crash when prosecutors proved that federal maps had inaccurately depicted the location of a broadcasting tower. In Indian Towing Co. v. United States, 350 U.S. 61 (1955), the federal government was found negligent for not maintaining a lighthouse marked on federal charts. In this case the lighthouse's location was marked correctly, but the government neglected to inform navigators that it was no longer operating.

While the survey did not go so far as to determine whether validation is carried out against the real world objects that a data set represents (i.e., an independent check), or if secondary sources only are used, personal experience would tend to support the latter rather than former. Clients have typically requested validation against secondary sources rather than incur the increased cost of field validation. This must lead to statements of quality being read with some caution.

[49] Conflation is the fusing or bringing together of two different sets of data into a composite data set, combining the best features of each set. When referring to spatial data another common term used to describe the same process is rubber sheeting.

7.2. Speech Recognition Testing

Speech recognition testing was undertaken by recording spoken commands with a Dictaphone and then comparing them with a log of commands recognised by the speech engine. Agreement is binary, in that the recorded command either matches the computer recognized command or not. In order to obtain a reasonable understanding of speech recognition performance, testing was carried out in three different environments with respect to background noise. The first test was performed in a quiet environment (office) where background noise was minimized. The second test was in an environment where the background noise was relatively loud but constant (tested while driving the car along Crowfoot Trail between 4:30pm and 5:15pm). The background noise in the final environment was variable in that there were moments of very low background noise and very high background noise. This test was undertaken on November 28, 2001, while walking along 32nd Avenue N.W. just to the North of the University between two and three o'clock in the afternoon. Quiet periods were observed when there was no traffic; noisy periods occurred when traffic lights turned green or when public transit buses passed. The environmental conditions were cold but sunny and the sidewalks were generally cleared of snow. In the quiet environment, the system was also tested on two different computers to determine if computing power affected speech recognition performance. The first was a Pentium II 450 MHz computer with 384 MB RAM, and the second was a Pentium III 700 MHz computer with 256MB RAM.

Table 7-5 summarises the results in each of the three environments described above:

Table 7-5: Speech Recognition Results

Environment	*Numbers of Commands*	*Technical Accuracy*	*Total Accuracy*
Quiet	*507*	*99.3%*	*98.2%*
Constantly Loud	*463*	*96.8%*	*95.5%*
Variable	*234*	*58.8%*	*57.3%*

In order to be precise about recognition performance, recognition rates were categorized in two ways. Technical rates were calculated on the basis of recognition errors resulting from vocabulary utterances (substitution and rejection errors) and insertion errors caused by extraneous noise (e.g. sneezing, cars driving past). This measures how well the speech system recognizes commands and screens out noise. Total rates include the categories described above but also take into account the non-recognition of words because of: speaking novel commands that are not in any vocabulary; speaking commands in the wrong context, i.e., not in the active vocabulary; issuing commands while the microphone is off but not realizing it; issuing a command which is not loud enough to be picked up by the microphone; commands judged to be heavily distorted (e.g., saying only half the command). Broadly speaking, total rates take into account those errors that can not be put down simply to poor technical performance but rather are, at least partially, due to inappropriate user behaviour or system design. While this distinction is not perfectly clear cut, it serves to differentiate approximately between technical performance and issues related more obviously to human factors and design.

Within a commercial environment it is normal practice to define acceptable accuracy operationally; that is acceptable speech recognition accuracy would need to be determined in relation to existing practices and requirements of an organization. As current accuracy rates of Street Condition Surveys undertaken by the City of Calgary have not been determined it is not possible to determine if the accuracies obtained are adequate or not. However, the results for the quiet and constantly loud categories are considered to be adequate; whereas the third category, the variable environment, is not, given that this is the environment in which most data acquisition will be performed. The poor result in the variable noise environment is attributed to the fact that the speech engine has to process all the sounds that it heard; if traffic was busy the computer captured this and tried to make sense of it. Speech recognition is extremely processor-intensive, so in times of high background noise it was found that it could take several minutes before the speech engine actually caught up and recognized a valid command. This must be considered unacceptable in an operational environment.

In a constantly noisy environment the speech engine can sample the background noise at the beginning of an exercise and then attempt to remove this from everything that it hears. As is evident by the results listed in Table 7-5 the major source of error is the

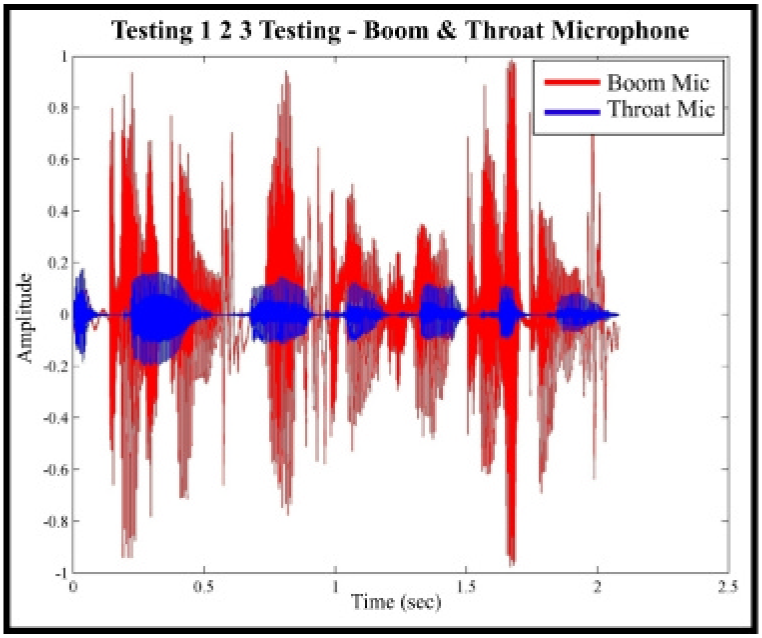

Figure 7-4: Boom and Throat Microphone Signals

microphone in that it captures all sounds within its range. The wearable computer comes with its own specialized directional microphone, but it is clearly not adequate for working in variable noise environments.

In an attempt to minimize microphone based errors a throat microphone from WirelessWorld was tested. However it was immediately apparent that the signal received by the speech engine was substantially different from that received from a boom microphone. Figure 7-4: Boom and Throat Microphone Signals depicts the phrase “Testing 1 2 3 Testing” as recorded by the boom microphone provided with the Wearable computer and the throat microphone. The signals are different. While the two signals are not synchronized it is clear that the throat microphone is not able to capture the high frequency components that the boom microphone does. This observation is supported by Figure 7-5 which depicts the frequency components obtained from a Fast Fourier Transform of the PCM signal generated by the sound card for the word “one”. This figure indicates that the boom microphone captures a substantially wider frequency range than does the throat microphone, and that the primary components captured by the boom microphone are at a much higher frequency. While to the human ear the sound recorded by the throat microphone is acceptable, it diverges substantially from the sound that the speech engine

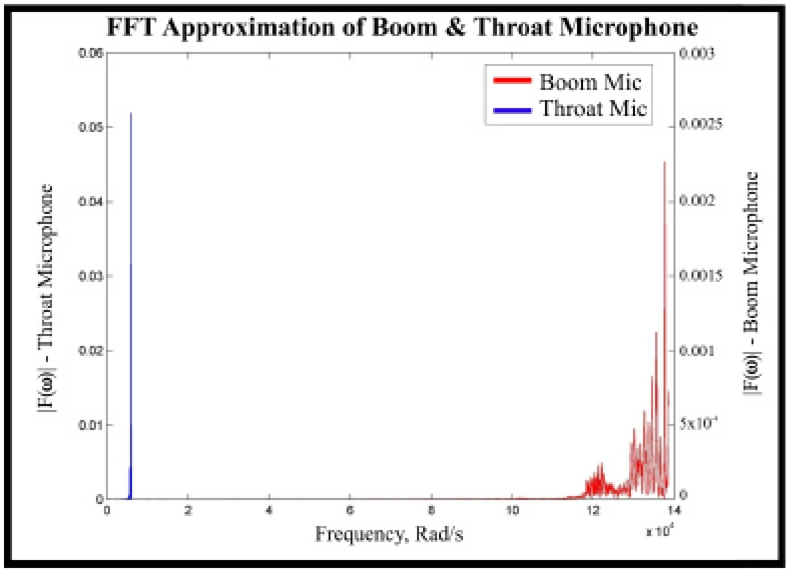

Figure 7-5: Boom and Throat Microphone Frequency Components for the Word "One"

expects to hear, hence the inability of the speech engine to function when used with the throat microphone.

7.2.1. Additional Speech Recognition Tests

Additional tests were undertaken to determine if computing power has an effect on the performance of speech recognition. The tests compared the time taken to initialize the speech recognition engine on the wearable computer and a desktop PC, and the approximate time taken to recognize a command. The desktop PC was a Pentium II 450 MHz computer with 384 MB of RAM. Three tests were performed to determine the time to recognize a command. Two of the tests, one for each computer, were performed in a controlled environment so as to minimize environmental noise, being EN E 228K at the Department of Geomatics Engineering, University of Calgary. The third test was in the constant noise environment described in Section 7.2 above. The desktop PC and the wearable computer were configured differently in that they were using different operating systems, being Windows 2000 and Windows 98 respectively, and the desktop PC was running five extra applications which occupied an additional 15,780K of memory. However, it was considered that these differences would have minimal effect on the outcome of these analyses.

The initialization test consisted of recording the times that the speech recognition engine commenced, and then completed, its initialization process. Ten tests were performed on each computer. The mean time observed over the tests on the PC was 24 seconds (s = 10 seconds), whereas on the Xybernaut MA IV it was 60 seconds (s = 5 seconds).

Initially the test to determine how long it took to recognize a command was to be performed by recording the recognized command and the time that it was recognized to a text file while recording the spoken commands to a wave file, and then compare the times differences between the spoken and recognized commands. However, the wearable computer could not record the speech to file, process the GPS and speech data, and transmit GML files at the same time. An out of resources message would be displayed or the wearable computer would crash. As such, the spoken commands were recorded on a Dictaphone as described in Section 7.2, and the time interval between each spoken command was determined using a stopwatch. It was estimated that the error due to the use of a stopwatch to record the time interval between the spoken command and the recognition of the command was 0.5 seconds.

The data used for the speech recognition speed test using the wearable computer was the same data that was used for the speech recognition accuracy test and consisted of 507 commands in the quite environment and 463 commands in the constantly loud environment. Both wearable computer tests lasted for 45 minutes. The PC based test consisted of 468 commands issued during a 30 minute session. Analysis of the time differences indicate that the Xybernaut MA IV recognizes speech commands approximately one second later ($\bar{x}$ = 1.5 sec., s = 4.5 sec.) than does the PC in a quiet environment ($\bar{x}$ = 0.5 sec., s = 0.6 sec.), this increases to approximately three seconds in a noisy environment ($\bar{x}$ = 4.8 sec., s = 10.5 sec.).

Given these values it is evident that both computing power and environmental noise have a significant effect on speech recognition. If the Null Hypothesis is that computing power does not have an effect on speech recognition performance then the Z statistic for the comparison of means should be less than 1.645 at the a = 0.05 level of significance. Given the results above for the tests in a quite environment a Z value of 5.0 is calculated indicating that the means are significantly different at the a = 0.01 level of significance also

(H_0 if $Z < 2.326$ else H_1). If noise alone is considered a Z statistic of 6.3 is calculated from the results obtained for the tests of the wearable computer in both the quite and noisy environments.

7.3. Positional Accuracy

Various modes of transportation were used while testing positional accuracy so as to be able to match the application to different data acquisition requirements. Four transportation methods were analyzed: standing on a feature[50] to be captured, and capturing features while walking, riding a bike at approximately 10km/h and driving a car at approximately 20km/h over it. During the walking test the command to capture a defect was issued when passing directly over the control mark. With regards to the cycling test the GPS was mounted on the bike carrier directly behind the seat. The command to capture a defect was issued when the cyclist was estimated to be directly over the control mark. During the driving test the GPS was mounted on the roof of the car above the driver and the car was driven over the control mark so that the driver passed as close to the mark as possible. The command to capture a defect was issued when it was thought that the driver was directly over the control mark. Each of the tests consisted of 30 observations. The tests were completed on May 26, 2002 and June 2, 2002 in University of Calgary Car Park #10. On both days the car park was nearly empty (it contained between 20 and 30 cars) and the conditions were clear and warm with light cloud cover developing around mid afternoon on both days. On May 26, 2002, there was a light north-westerly breeze causing additional environmental noise when cycling into the wind. This is evident in the results of the cycling test as the mean observed position is in the northwest quadrant and nearly twice as far away from the control mark as the other test. This apparent shift is assumed to be a result of the increased background noise causing the speech recognition engine to recognize commands more slowly. During the observation sessions GPS PDOP values were between 1.4 and 2.7, indicating that satellite geometry was good throughout each of the sessions. Multipath errors were considered moderate through out the observation periods as there were few reflective

[50] For the testing of positional accuracy the feature captured in each instance was a control mark that was independently surveyed (two 30 minute static surveys) relative to Pillar S2. A positional accuracy of ±0.004m at the 99% Confidence Level (n = 40) was obtained.

surfaces (some cars parked at the south end of the car park) within 150m of the observation site. A zenith mask of 15 degrees and a PDOP mask of six were set on both the Base GPS and the Rover so as to minimize atmospheric effects. All GPS positions were determined using Real-Time Kinematic Positioning, i.e., carrier phase DGPS.

The model for this analysis is therefore described by four categorical explanatory variables (the transportation modes) and a response variable that is essentially a continuous measurement. As the explanatory variables are categorical one of the simplest graphical views is the Box plot as shown in Figure 7-6. The white horizontal line shows the median response for each mode. The bottom and top of the box indicate the 25 (Q1) and 75 (Q3) percentiles respectively (i.e., the location of the middle 50% of the data). The horizontal lines joined to the boxes by the dashed lines (the whiskers) indicates observations that are within 1.5 times the inter-quartile range (IQR: Q3 – Q1) of the first and third quartile. These lines coincide with the closest observation that is less than or equal to Q3 + 1.5 IQR

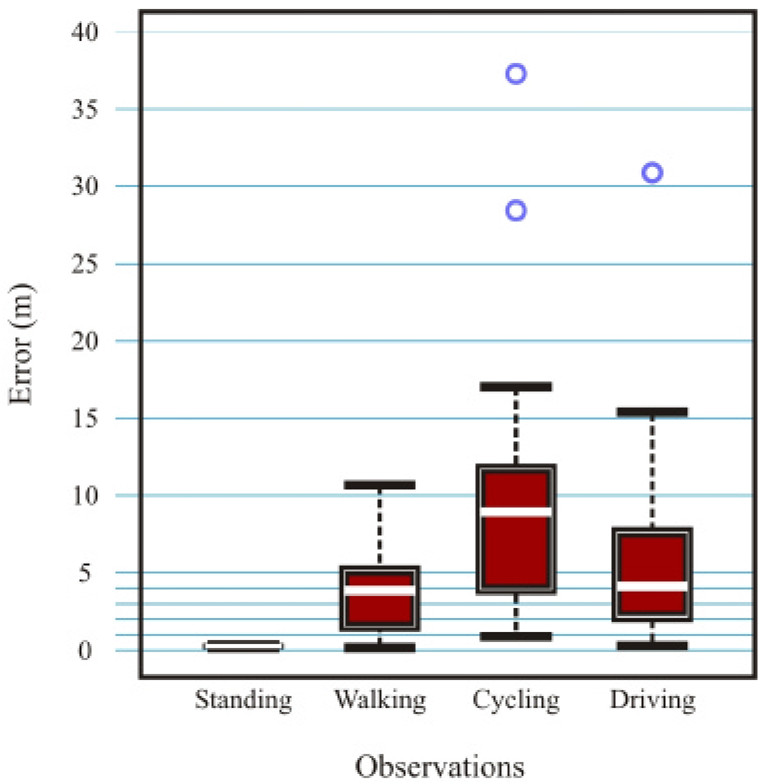

Figure 7-6: Box Plot of Observations for each Mode of Transportation

for the upper line and greater than or equal to Q1 – 1.5IQR for the lower line. Points beyond these lines (outliers) are drawn as individual circles. What is evident from Figure 7-6 is that the range of observations is substantial for the Walking, Cycling and Driving samples in comparison to the Standing sample, and because of the asymmetry in the sizes of the upper and lower parts of their boxes the observations for each of these samples are likely to be somewhat skewed.

Table 7-6 summarizes traditional features such as the mean, standard deviation, variance, etc. Because Figure 7-6 indicates that outliers exist in the observations some additional features have also been calculated. For example the Median Absolute Deviation (MAD) is a more robust measure of variance as it is not sensitive to outliers because the data in the tails have less influence on the calculation of the median than they do on the mean (NIST/SEMATECH, 2002). It is common practice (Crawley, 2002) to compare both MAD and the standard deviation as an alternative means of predicting if outliers exist. Typically, if standard deviation is greater than three to four times MAD then it can be assumed that outliers exist. With respect to this data the comparison implies that no outliers are present.

Table 7-6: Position Summary Statistics

	Standing	*Walking*	*Cycling*	*Driving*
Mean (m)	*0.250*	*3.846*	*9.512*	*5.622*
Standard Deviation (m)	*0.057*	*2.632*	*7.988*	*5.959*
Median Absolute Deviation (m)	*0.049*	*2.058*	*5.641*	*3.886*
Confidence Level (95%) (m)	*0.021*	*0.983*	*2.983*	*2.225*
Count	*30*	*30*	*30*	*30*

For this data the Walking, Cycling and Driving observations are positively skewed (skewness equals 0.6, 1.8 and 2.9 respectively), while the Standing observations are normally distributed (skewness equals 0.1). The Kurtosis measure indicates that the Standing sample is platykurtotic (flat topped distribution, Kurtosis equals -1.1), the Walking sample is normally distributed (Kurtosis equals 0.1) and the Cycling and Driving samples are leptokurtotic (pointy top distribution, Kurtosis equals 4.5 and 10.8

respectively). When comparing the means of two samples the distribution of a sample helps to determine which test should be utilized. If samples are normally distributed then it is possible to use the Student's *t* Test. However, if they are not, as in these cases, the Wilcoxon Rank Sum Test is more appropriate (Crawley, 2002), the results of which are depicted in Table 7-7.

Table 7-7: Wilcoxon Rank-Sum Test for Sample Pairs

	Walking	*Cycling*	*Driving*
Standing	$Z = -6.210, P = 0.000$	$Z = -6.646, P = 0.000$	$Z = -6.483, P = 0.000$
Walking	-	$W = 680, P = 0.000$	$W = 848, P = 0.328$
Cycling	-	-	$W = 1084, P = 0.012$

$n = 30, m = 30$

In all tests comparing the Standing sample with the others an exact *P* could not be computed, hence the Wilcoxon Rank-Sum Test uses a Normal approximation to determine the Z value and from this a *P* value for the hypothesis that the means are the same. For these tests the Null Hypothesis is that the means of each observation pair, Standing/Walking, Standing/Cycling, Standing/Driving, etc., are not significantly different.

In order to minimize the chance of a type one error, that is the error of incorrectly declaring a difference to be true due to chance producing a particular state of events, the Bonferroni adjustment ensures that the overall risk for a number of tests remains at a = 0.05. For example, in five tests the chance of finding at least one difference due to chance is 0.22, or one in five (SISA, 2002). Therefore to ensure that the overall a remains at 0.05, a must be lowered to 0.008 for each of the six tests described in Table 7-7.

For these observations *P* values of 0.00 for all comparisons, except the Walking/Driving and Cycling/Driving cases, are less than 0.008, therefore the Null Hypothesis is rejected. For the Walking/Driving and Cycling/Driving cases *P* is greater than 0.008 therefore the Null Hypothesis, that the means from these observations are not significantly different, is not rejected.

The standard test for comparing whether sample variances are significantly different is Fisher's *F* Test, the results of which are listed in Table 7-8. For these tests the Null

Hypothesis is that the variances of each pair are not significantly different. If we apply the Bonferroni adjustment then if the calculated variance ration (*F*) is greater than or equal to 2.50 (d.f. num = 29, d.f. denom. =29, a = 0.008) then we can conclude that the two variances are significantly different at a = 0.05.

Table 7-8: Fisher's F Test for Variance Equality

	Walking	Cycling	Driving
Standing	*F = 2164.8, ? = 0.000*	*F = 19940.9, ? = 0.000*	*F = 11096.8, ? = 0.000*
Walking	-	*F = 9.211, ? = 0.000*	*F = 5.126, ? = 0.000*
Cycling	-	-	*F = 0.557, ? = 0.120*

F Critical = 1.86; Degrees of Freedom: Numerator 29; Denominator 29

For the pairs Standing/Walking, Standing/Cycling, Standing/Driving, Walking/Cycling and Walking/Driving we can reject the Null Hypothesis as *P* equals 0.000 in each case and *F* is greater than 2.50. For the Cycling/Driving case the Null Hypothesis is accepted as *F* is less than 2.50 and *P* = 0.120. That is to say the variances observed in the Cycling and Driving tests were not significantly different, whereas the variances between the other tests were.

Below are two figures showing different views of the sample data. Figure 7-7 plots residuals versus fitted means (the fitted values in this instance are the mean error for each

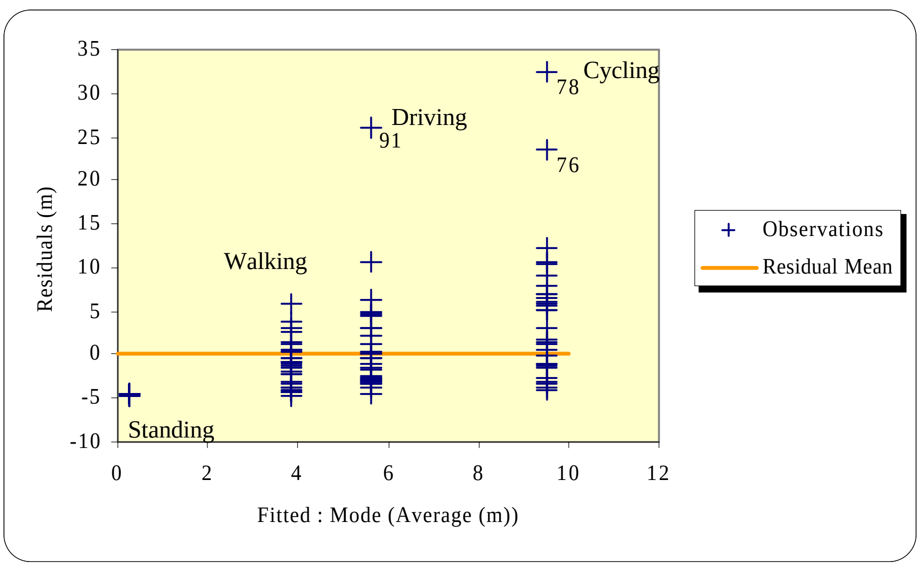

Figure 7-7: Residuals versus Transportation Mode

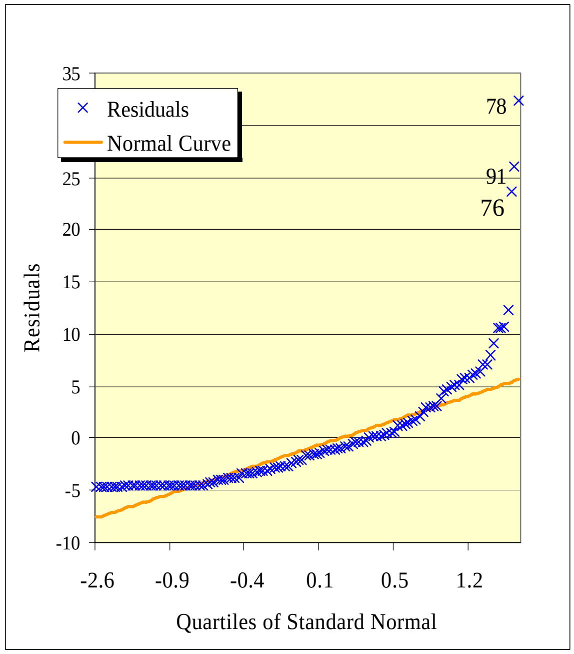

Figure 7-8: Normal Probability Plot

mode of transportation). If the observations are normally distributed then the residuals should form a rectangular shape parallel to the x axis in the plot. They do not. In this case the plot indicates that the data is not linear and that variance increases as the mean increases. Figure 7-7 also displays some asymmetry between the size of the positive and negative residuals. Figure 7-8 is U-shaped which indicates strong signs of non-linearity. In both figures data points numbered 76, 78 and 91 would appear to have the largest influence on the model.

These results indicate that for speeds up to 20km/h background noise and speech recognition have a greater effect on positional error than does speed alone as the mean positional error for cycling was 9.51m as opposed to 5.62m when driving. This result reflects the findings of the speech recognition testing in that the noise environment when driving is more constant, therefore enabling the speech recognition engine to perform more efficiently and hence recognize commands more rapidly. When cycling in a relatively constant noise environment, even the effect of “wind noise”, in particular, noticeably degrades positional accuracy. If background noise can be minimized, or made constant, then for speeds up to 20 km/h, positional accuracy can be improved, whereas, if back ground noise is relatively constant between different modes of transportation then the expectation that positional accuracy improves with speed is realized. It is anticipated that at some point increasing speed, while in a constant noise environment, will increase positional errors to a level greater than that observed while cycling. However, further driving tests at higher speeds are required to determine this threshold.

As an aside, additional environmental noise resulting from walking through snow also had an effect on the speech engines ability to recognize commands. While positional accuracy indicators are not available, recognition rates decreased noticeably, and the time to recognize a command increased, providing further antidotal evidence that current speech recognition engines can not yet perform adequately in outdoor environments.

7.4. SUMMARY

A web-based questionnaire was answered by 80 GIS project managers, who indicated that they are unhappy with the accuracy and quality of their data, although they do not require the data in real-time. Speech recognition testing has been carried out in three different noise environments. Both technical and overall accuracy exceeded 95% in environments that were quiet or constantly loud. However, for tests while walking along a busy road during which the noise level varied, the accuracy of the speech recognition plummeted to 58%. Four positional accuracy tests designed to determine the positional accuracy of defects were undertaken by standing on a defect, and walking, cycling and driving a car over it. Each test consisted of 30 observations and resulted in positional errors of 0.27m, 4.83m, 12.50m and 7.85m respectively at the 95% confidence interval. Of the four methods of transportation the mean positional errors observed while walking and driving, and cycling and driving were equivalent. All other combinations can be considered unrelated and therefore different qualities of positional data can be expected.

CHAPTER 8

8. CONCLUDING REMARKS

This final chapter serves to link the earlier chapters and analyzes the research that has been undertaken in the development of a speech aware mobile GIS application. The analysis is followed by a collection of secondary findings. Finally, it makes recommendations for the future development of a mobile GIS application with particular reference to speech and technical capabilities.

8.1. THE ANALYSIS

This thesis began with the comprehensive analysis of Mobile Geographic Information Systems and the many issues affecting development of such systems. The literature shows that there a number of factors (wireless communication and mobile device limitations) that have a detrimental effect on development in a mobile environment.

One of the primary tasks of this research has been to look at an alternative method of data acquisition, the objective being to improve spatial accuracy, improve attribute accuracy, minimize acquisition time frames, and remove intermediate processes that are typically required to get Geographic Information from the field into an application. The fundamental question that this research has addressed is whether or not a mobile GIS, that includes speech recognition and wireless connectivity for real time access to spatial data, is a viable tool for data acquisition.

In order to simplify this question it was broken down into a series of objectives that have been addressed by various chapters. The first of those objectives was to develop an architecture for mobile GIS, using a wearable computer, based on the principles of interoperability. To this end, a data acquisition tool has been developed, based around the Xybernaut MA IV wearable computer, using ESRI's MapObjects 2.1 ActiveX control.

The hardware architecture consisted of a Real-Time Kinematic GPS configuration, a wearable computer, a PCMCIA wireless network card that provided a connection to the Internet, a server that processed data captured by the mobile GIS and the wired and wireless networks through which data was transmitted. From a software perspective the architecture is similar to that of a traditional client/server application, with two additions. The most significant of these additions was the need for a wireless Gateway, which is the interface between the wired and wireless networks. The interface was used to convert information transmitted by a mobile device from the carrier format used by the wireless network to an Internet protocol such as HTTP, and vice versa. Because of the desire to simplify the implementation process, a publicly provided Gateway (from Telus Mobility) was utilized for this research. The second addition was the inclusion of a Session Work Queue. The purpose of the Session Work Queue was to maintain a record of the data that was to be sent across the networks. If a connection could not be established between the mobile device and the server due to a lack of wireless connectivity, then the Session Work Queue ensured that once a connection was re-established, data that was meant to be sent from the mobile device to the server, or vice versa, was transmitted accordingly.

The most important element making up the mobile GIS, is the wearable computer. The reason for selecting a wearable computer over other mobile computing devices was, and continues to be, the greater capability of the wearable computer in terms of processing power. However, given the poor performance of speech recognition in a variable noise environment, and the inability of the computer to process GPS data at rate greater than 0.5Hz, it is clearly evident from this research that the Xybernaut MA IV is not sufficiently powerful to cope with processing requirements of this application. Speech recognition on its own typically requires a Pentium 600 to 800 MHz computer with a minimum of 384 MB of RAM, and an operating system such as Windows 2000 Professional. The Xybernaut MA IV used for this research is a Pentium 233 MHz computer with 128 MB of RAM, running Windows 98. While a lower performing computer does not necessarily mean poorer speech recognition results, it does result in slower response times as the CPU can not process data as quickly. A comparison of tests between the Xybernaut MA IV and a desktop PC (450 MHz, 384 MB RAM) indicate that the Xybernaut MA IV recognizes speech commands approximately one second later than does the PC in a quiet environment, this increases to approximately three seconds in a noisy environment. Poor performance is also evidenced by the time it takes the speech engine to initialize. The average time observed over ten tests on the PC was 24 seconds, whereas on the Xybernaut MA IV it was 60 seconds. As such, slower computing performance is particularly noticeable in noisier environments. All speech recognition tests undertaken for this research lasted for approximately 45 minutes. When testing in a quiet environment some 500 commands could be issued and processed during that time, while in noisier environments the number of commands that were issued was less than half of that (234). To compound the performance issue, speech recognition rates plummeted by some 40 percent. While this drop in performance was not solely a consequence of the processing power of the Xybernaut MA IV, as environmental noise also played a role, processor speed does have an effect on the performance of speech recognition in situations where the Central Processing Unit (CPU) was working at, or near, 100%, as was often the case in the variable noise environment. When the CPU was working at maximum capacity, the sound card was unable to record commands issued by the user, thereby reducing speech recognition performance.

In addition to the performance issue related to speech recognition it was found that if the roving GPS provided positional data at a rate greater than or equal to 0.5 Hz the processing required to extract the location information from the NEMA sentences was in excess of what the Xybernaut MA IV was capable of handling. After testing a number of acquisition rates it was found that 0.5 Hz was the highest rate that the computer could process and still adequately service other processing demands, i.e., speech recognition. It should be noted that since commencing this research Xybernaut has developed the MA V, which is based on a 500 MHz Celeron processor that can be expected to outperform the MA IV. By being able to perform approximately twice as many operations in the same amount of time it is anticipated that speech recognition in a variable noise environment would improve because the computer would be able to process the sounds that it captures more rapidly and not miss as many commands because the CPU would not be required to perform to its maximum capacity as often.

Authors such as Goodchild *et al.*, (1998) and Bishr (1998) have decomposed interoperability into a number of elements relating to technical, semantic and institutional components. This research has approached interoperability primarily from a technical standpoint with some emphasis on semantics. As promoted by the OpenGIS Consortium, the Geography Markup Language (GML) has been implemented as a means of ensuring interoperability by standardization. While GML does not specifically address the semantics of geospatial information in the global sense, it does provide a means of strictly defining spatial information, which allows others to more accurately interpret the contents of a spatial data set, regardless of their world-view, and make an informed decision as to its fitness for use. GML provides three XML Schemas that provide base geospatial types (0, 1, and 2 dimensional geometry types) and structures which may be used by an application Schema to create application-specific features.

The second objective of this research was to investigate whether speech recognition is an effective method for capturing spatial features and their attributes, by determining if speech recognition responds with sufficient accuracy, and in a timely manner so as to ensure accuracy of position. As reported by Oviatt (2000), and confirmed by the speech recognition analysis carried out for this research, there is up to a 50 percent decrease in recognition rates when speech recognition is implemented in an actual field environment. The reduction in recognition rates when in a field environment (uncontrolled) is primarily related to the inability of the system to adequately remove background noise, which is difficult, if not impossible, to predict and therefore model. The reason that variable background noise is difficult to remove is related to the way in which speech recognition processing works. At the start of a speech recognition session the user typically performs an initialization test in which the user reads a paragraph of text to the speech engine. During the test the speech engine compares the speech signal that is recorded against a standard signal for the same text. From this comparison of signals the speech engine determines the background, or environmental noise, from which a filter is created to suppress environmental noise during the remainder of the speech session. As such the filter is most effective when the background noise is constant. In a controlled environment it is evident that speech recognition is sufficiently fast, and accurate, however in an uncontrolled environment it is not. Thus given the current state of speech recognition technology it is not a viable option to the current methods used for field data acquisition. While the development of a speech recognition engine for a throat microphone is feasible, it is a substantial task in itself. A typical speech engine is developed by recording the speech of a substantial number of people (typically about 100 people from which 7,000 to 10,000 utterances are recorded (Huang *et al.*, 1994)), extracting the phonemes from each speaker, and averaging them to create a database of phonemes.

As reported by Askenfelt *et al.* (1980) and observed during this research, alternative microphones such as a throat microphone, while efficient with respect to minimization of background noise, produced a lower fundamental frequency than that which is expected by the speech recognition engine. This divergence from the training model upon which the speech engine has been designed results in almost total recognition failure. However, if a speech engine could be developed using only speech recorded with a throat microphone then it is anticipated that recognition rates could be improved to match those obtained using a traditional boom microphone in a quiet environment.

Jones *et al.* (1992) identified a number of criteria designed to improve recognition accuracy. The most important criterion was the implementation of a grammar hierarchy, the purpose of which is to minimize the number of commands that need to be recognized at any one time. It was found during this research that it was necessary to display which grammar is active at any one time. It was not uncommon, once or twice per session during the initial training/development phase, to forget which was the active grammar and start issuing commands that the computer was not expecting to hear. This generally led to the repeated issue of incorrect commands, and the effective stalling of the application if the visual or sound aids were not available. As reported by Murray *et al.* (1996) this problem is partially exacerbated by the user persisting with speech commands when an alternative means of interaction with the computer, i.e., via the keyboard or mouse, could break the non-recognition loop.

This is essentially a training issue. For example, when we converse with another person, and say something incorrectly or are not understood, we will generally rephrase what it was we wanted to say. It is apparent that when speaking to a computer we want to continue to use speech when we have made an error, as we would in normal conversation, rather than adopting an alternative mode of interaction that would ensure that the computer responds appropriately. However, as the application was used more regularly, it was noted that this habit was gradually broken. As observed by Shneiderman (2000), emotive content of spoken language, while important to human-human interaction, can be disruptive to human-computer interaction. For instance, while stuck within a non-recognition loop it was not unusual for the user to issue commands in a more aggressive manner, compounding recognition problems because the commands issued diverged from the speech engine's training model.

One of the objectives of wearable computers is to ensure that the user maintains an awareness of the local environment. This research indicates that when speech recognition is working correctly this goal is met, and as such speech recognition can find a useful role in hands-busy, eyes-busy situations as reported by Murray *et al.* (1996). However, when speech recognition fails to perform adequately the user’s total concentration is restricted entirely to the software application, thus returning the user to the traditional human computer relationship where the computer demands the user’s full attention in order to complete a task. This is not satisfactory when working in a mobile environment. To highlight the consequences of this predicament, while testing the application on the bicycle, speech recognition tended to perform more poorly when riding into the wind (due to increased background noise) and therefore required additional attention to ensure that the correct process was being activated. Although testing was undertaken in an empty car park on the weekend an accident with another cyclist was only averted due to the vigilance of the other cyclist, that is, I was sufficiently distracted by the software application that I was unaware of what was happening in my local environment. This tends to support the findings of authors such as Shneiderman (2000) and Strayer *et al.* (2001), who have reported reduced cognitive abilities of speech recognition users who are attempting to multitask, and cell phone users who are driving, respectively. Speech recognition is statistical by nature, so it is highly probable that recognition errors will occur. Unless recognition accuracy can be further improved, user safety, and the safety of others, must be considered when determining the suitability of this application for a particular process.

Given the limitations of the NMEA-0183 standard, in that data can only be sent from the GPS units to the wearable computer, and the limitation of the wearable computer to only be able to process GPS positions at rates of less than 0.5 Hz, it was necessary to interpolate the position of a feature. Interpolation was based on the time that the command to capture a feature was issued and the time difference between GPS positions received just before and just after the issued of an “Add Defect” command. By its nature, the interpolation process will add to the positional error of a captured feature. The magnitude of the positional error will be a function of the time difference between consecutive GPS positions, the speed at which the user is moving, and the users dynamics, i.e., are they smooth. The positional error is also compounded by the speed with which the speech engine recognizes the command to capture a defect. During the initial design it was intended that the application would make use of a function within Microsoft's Speech Application Programming Interface that records the time that the microphone senses the commencement of an utterance, so as to be able to minimize the error of interpolated positions. However, it was found that background noise made it extremely difficult to determine exactly when the command to capture a feature was issued. In the final application it was determined that the most practical time to use for the interpolation of a position was when the speech recognition engine actually recognized the command to capture a feature, even though it is accepted that this process will degrade the position of captured features to some extent due to the delay between the issue of the “Add Defect” command by the user and the recognition of the command by the speech recognition engine.

After having used this multimodal environment for a period of time, in both an office and field based environment, it is apparent that speech recognition in an office environment, where environmental noise can be controlled, can enhance human-computer interaction. By incorporating additional modes of interaction, such as sound and speech, between a user and the computer, information can be presented in a more efficient and accurate manner. This is particularly evident with respect to the verification of captured data. A speech recognition engines ability to convert text to speech and “vocalize” what it has recognized provides a quick and easy means of data verification. During testing of the system missed recognitions were easily identified, primarily, it is assumed, because a different mode of interaction with the computer was invoked. Unfortunately, aside from the verification of data acquisition, these enhancements to human-computer interaction do not translate well in to a field environment.

The third objective of this research has been to investigate the positional accuracy of captured features using different modes of transportation. The goal was to determine the capabilities of such a tool in light of end-user requirements regarding positional accuracy. A significant portion of geospatial data users would like to have spatial data with an accuracy of better than one meter (?^{2} test with Yates' Continuity Correction = 20.527, P = 0.000). Given the test results obtained, this accuracy criterion can be met if the user stands upon the features to be captured for at least the time taken for the GPS unit to provide the application with two positions. If the GPS unit provides a position at the rate of 0.5 Hz, then the user need only stand in one place for 4 seconds at most. This will ensure that the application will have received sufficient information from the GPS unit to accurately interpolate the position of a captured feature. Given that the accuracy of the other modes of transportation tested - walking, cycling and driving - were significantly greater than one meter, these modes of transportation will not meet the requirements of most geospatial users within a local government environment. However, these modes of transport may be adequate for the capture of spatial data for other types of applications, for example environmental, natural resources, or market research, where the accuracy requirements are not as stringent (Montgomery *et al.*, 1993).

The fourth objective has been to explore real-time access and transmission of spatial information over a wireless communication interface. The wireless network is the enabling component of a mobile application in that the user has access to data without being restricted by place. Users are restricted to public cellular networks via a CDPD modem, to GSM or CDMA wireless networks via a wireless hand phone connected to a computer with a serial cable, or to a private wireless network. With all networks there are limitations with respect to coverage (the area within which the network can be accessed) and cost to access the network. At present the cellular networks have the greatest coverage followed by the wireless networks.

Because of the limited number of ports available on the wearable computer, the simplicity of using a PCMCIA card, the greater coverage, and the better pricing structure (flat monthly rate as opposed to a time based or packet based rates), a CDPD modem was selected as the most appropriate wireless option. While the published data transmission rate for a CDPD network is 19.2 kbps, observed transmission rates were generally in the range of 10 to 12 kbps. However, by minimizing the amount of data that was required to be transmitted, these data rates did not appear to significantly delay the transmission of data. A typical GML file for one road defect required approximately 1860 bytes, which were usually broken up into three packets of 536 bytes plus the remainder of around 250 bytes. For GML files containing more points, for example 30, the generally packet size was again 536 bytes, although packet sizes ranging from 2000 to nearly 8000 bytes were also observed. In general, one defect was processed by the server within two to three seconds of being sent by the mobile device. However, with the larger files (10 points) this often extended to approximately 15 seconds with the majority of the time being taken up by the transmission of data over the wireless network.

The wireless component provides the greatest opportunity for time and cost savings. By providing the mobile user with a connection to corporate databases in this manner, the user is able to add, modify and delete data in near real-time, thus removing the need to spend additional time, or employ additional personnel to process the data in the office.

One of the principal goals of this research has been the simplification of data acquisition; simplification in terms of the process that must be undertaken to acquire data and simplification in terms of what is required of the field user. The current system does not respond as instructed, nor in a time frame that is acceptable when working in a typical outdoor urban environment. However, if the background noise issued can be resolved, through improvement of speech recognition algorithms that extract speech in a field environment, or the development of a speech engine for throat microphones, it is apparent that this mode of interaction with the computer shows promise, but user safety must also be considered. Speech recognition's ability to interact with the user via text-to-speech capabilities (playback to the use of what is recognized) provides an opportunity to verify captured data on-the-fly, rather than having somebody else verify it later.

If head mounted displays are to be utilized, speech recognition is a necessity. Without it, interaction with the computer by mouse and miniature keyboard is difficult. In short, experience indicates that when the system works, it works well and is easy to work with, but when the system fails to respond appropriately, human-computer interaction becomes difficult and user frustration quickly escalates.

Therefore to answer the principal question of this research, "Is a mobile GIS, that includes speech recognition and wireless connectivity for real-time access to spatial data, a viable tool for data acquisition?”, I must say “no”.

8.2. Factors to Consider

Aside from the technical issues encountered in the previous section there are a number of other issues that warrant discussion. The first of these relate to the use of ActiveX components when developing software. While the benefits of ActiveX components far outweigh the problems encountered with their use, a number of unforeseen issues were encountered with ESRI's MapObjects and both Microsoft's and Dragon NaturallySpeaking's Speech Application Programming Interfaces. With respect to MapObjects, two major issues were encountered during development. The first related to database access. For some time now Microsoft's Active Data Objects (ADO) Control has been the standard interface used for database access; however, until the release of MapObjects 2.1, ESRI did not support the use of the ADO control. This would not have been a great problem had it been mentioned in the documentation provided by ESRI. Each of the major components were first developed individually to remove as many bugs as possible while the software was in a simplified form and integrated when the components were working satisfactorily. It was at this time that the database access issue was realized. The second issue with MapObjects is that it is not possible to add a defect to the Road Defect layer or theme more than once without first removing the layer from the project and then reloading it. This issue has been discussed at length on ESRI’s MapObjects discussion forum, but has yet to be resolved.

With regards to the Speech Application Programming Interfaces, it was found that a number of functions described by Microsoft and implemented by Dragon were only accessible using C++. As this application was developed in VB 6.0 it required additional libraries to be developed in C++ that would allow VB access to these functions. To compound this problem it was also found that some functions described by Microsoft have yet to be implemented. As the Microsoft Speech Application Programming Interface is a free development kit, Microsoft does not provide user support, making it difficult to access some of its functionality. In the end, speech recognition user groups were relied upon heavily in order to implement the speech recognition component of this project.

The world of XML is expanding daily, as is the XML standard. When commencing the development of the GML data model it was found that the base GML Schemas did not conform to the current XML standard. This required a considerable amount of time updating both the Feature and Geometry Schemas so that they did conform to the current XML schema and could then be parsed by an XML validator such as XML Spy. It would appear that the Open GIS Consortium lags the World Wide Web Consortium by approximately six months - being the time it took the OGC to update their GML schema to conform to the latest XML schema - with respect to their schema documentation.

Geographic Information Systems work because spatial data can be georeferenced. Without this capability, or if data is incorrectly georeferenced, a GIS is no more than a glorified database. Having determined that users would like data accuracy to be less than one metre it was decided that Alberta Survey Control Monuments (ASCM) around the University of Calgary would make appropriate control marks. However, after undertaking a small Control Survey between the pillars on top of the Engineering Faculties F Block and two ASCM's, one on the northeast corner of 32nd Avenue and 33rd Street NW (ASCM 263079), and the other halfway along the western side of 31st Street NW, opposite the parking lot of TR*Labs* (ASCM 156596), it was found that there was a discrepancy between the WGS 84 coordinates of the pillars and the ASCM's (3TM). The survey consisted of two static occupations of 30 minutes each, on each ASCM, with a second receiver on pillar S2 throughout the survey. The survey was adjusted using Trimble's Geomatics Office software. When adjusting the network with minimal constraints, i.e., only pillar S2 was fixed, the maximum error in either the Northings or the Eastings was 0.003m at a Confidence Interval of 95%. When the two ASCM were held fixed errors in the order of ±0.04m were observed. While these errors are larger than was anticipated for a baseline of 400m, insufficient ASCM’s were surveyed to determine the cause of these errors. It could be that one of the ACSM's was disturbed, although they were not visibly so, or there is a rotation between the WGS84 framework and the Provincial spatial reference system.

However, when all three control marks were constrained, using their WGS84 coordinates, an error of approximately -1.00m in the Eastings and +0.70m in the Northings (observed - control) was calculated[51]. The University of Calgary pillars were surveyed during the City of Calgary's initial Highly Precision Network campaign; however they were never integrated because they did not meet the Alberta Sustainable Resource Development Geodetic Control Units criteria for integration. It was therefore concluded that the errors discovered during this control survey are probably due to a bias caused by a lack of integration of the pillars with the surrounding ASCM's. While errors of this magnitude may be adequate for certain GIS data acquisition projects, if high precision data is a requirement, then it is imperative that the Base Station used as part of the Real-Time Kinematic GPS system is adequately referenced to the map projection within which the data to be captured must sit. As such, the coordinates used for Pillar S2 were determined by least squares adjustment of the GPS observations with the ACSM's held fixed.

There are two final observations. For a substantial period of time, some four weeks of trying to test the system, it appeared that it was not possible to run the GPS unit at the same time as the wireless modem. Either the wireless modem could not register with Telus Mobility, or the GPS could not lock on to any satellites. During this testing period it was found, by trial and error, that if the GPS unit was activated prior to connecting it to the wearable computer (the computer had to be turned off when starting the GPS unit) then everything functioned as was anticipated. The reason for having to power up the system in this particular order has yet to be determined, although it is speculated that the system is either generating noise which is interfering with the wireless components or there is interference between the GPS radio antenna and the wireless antenna, which is not allowing the system to register with Telus Mobility. The final observation is in respect to the number of satellites in view. Throughout the testing of the system no consideration had

---

51 Following a discussion with Mr. Geoff Banham of the Geodetic Control Unit, Director of Surveys and Technical Services, Alberta Sustainable Resource Development, it was determined that the University of Calgary pillars have yet to be integrated into the Provincial Spatial Referencing System.

been given to the satellite constellation. However, while testing it was noted that on occasion differential positions could not be determined (autonomous positioning only). Upon further investigation it was found that there were not four common satellites within view of both the Base Station and Rover. While both of these observations are relatively simplistic, it was the intention of this research to test the application in an environment that is as close to real-life as was possible. Hence, if such a system is to be implemented within the commercial world these issues must be factored in to the training of users so that they have an adequate understanding of the workings of GPS so as to be able to monitor quality indicators such as the number of satellites in use, fix status (autonomous, DGPS, etc.), PDOP, etc. and assess whether or not they are meeting their accuracy requirements.

8.3. Future Investigations

I suggest that this application can become a viable data acquisition tool if appropriate research is carried out in a number of areas. Because the primary component affecting the usefulness of the mobile GIS is a speech recognition component, any effort to improve the system must begin with this component. Specific areas for improvement are described below.

While the throat microphone was unable to improve recognition rates, inspection of its signal does indicate that if a speech engine could be adapted to recognize the signal recorded by a throat microphone then this may provide the best solution to the removal of background noise and therefore improves recognition rates. It is anticipated that the simplest method of adapting current speech recognition engines would be to observe the differences between speech recorded by a throat microphone and a boom microphone over a range of utterances and develop a transformation filter that can convert the signal recorded by the throat microphone to match that of the boom microphone so as to recreate the high frequency components of speech, prior to the signal being processed by the speech engine.

One of the handicaps of the current system is computing performance. While the performance of wearable computers has nearly doubled over the last 24 months during which this research was undertaken, investigation into the development of a computer specific to mobile data acquisition using speech recognition would be beneficial. The Xybernaut MA IV does not currently take advantage of current speech recognition technology provided by the more advanced Pentium III and IV chip sets. For example, Dragon NaturallySpeaking's version 4 speech engine is the latest engine that can be installed on the MA IV. The speech engine is now four years old, having been released in August of 1998, and as of October 2001 Dragon NaturallySpeaking released version 6. Hence it is not possible to make use of the most up-to-date technology.

Current speech recognition engines must operate continuously in the background of the computer's operating system. There is a noticeable lag in the operation and function of a computer when speech recognition is enabled. Hence, investigation into the integration of a speech recognition chip set with a miniaturized computer built around a PC/104 computer module, for example, is warranted. By moving the speech recognition processing demands from central processing unit to the speech recognition chip it is expected that all round performance can be improved. The Xybernaut MA IV is essentially a laptop reorganized to fit in a box 19 cm by 6 cm by 12 cm. If current laptop technology can include Pentium IV 1.5 - 1.7 GHz CPUs then it is evident that computing performance can be improved substantially over that of the Xybernaut MA IV.

Significant cognitive based testing is also required to ensure that the system is simple and intuitive to use. Cognitive testing should also investigate the user safety aspect of this application as there is significant evidence of higher accident rates for cell phone users when driving a vehicle, an environment that is not dissimilar from that which is encountered when using this application. Testing during this research has clearly indicated that feedback and error correction are important aspects of system design, as is training of the user to make use of the most appropriate mode of interaction with the computer under different situations. It is these human factor considerations that will be crucial in determining the success of a mobile GIS data acquisition tool based around speech recognition.

If the application can be improved so that it is a viable data acquisition tool then research in the domain of cost benefit analysis is also warranted. Given that the City of Calgary Streets Department spends approximately 110 man days per year entering and/or editing road and sidewalk defect information. By removing the need for this additional work, cost savings in the order of $18,000 and $19,000 per year are possible assuming an hourly rate of $13, an overhead factor of approximately 1.8 to 1.9, and a 7 hour working day. Field testing to determine if the application provides other tangible and non-tangible benefits, possibly as a result of improved efficiency in the acquisition of field data, or improved accuracy of data should also be investigated. Tangible and non-tangible benefits could then be assessed against procurement costs required to implement the system.

Last, with regards to the wireless/Internet component, further investigation with respect to reliability, data transfer capabilities and latency are warranted. At present the system only sends data to the server, and requires that all base data sets are maintained on the mobile device. However, a logical improvement to this would be to implement a system that allows the download of data sets within a certain distance of the field operator, which can be updated dynamically as the user moves from location to location. This would minimize the data storage demands on the mobile device and allow more flexible access to corporate data sets.

B i b l i o g r a p h y

AltaLIS Ltd., (2002), Property: Title Mapping, (accessed 2 July, 2002), http://www.altalis.com/downloadables/April%202002%20Updates/title%20mapping%20factsheet.pdf

Alwang, G., (2000), Speak Easier with L & H 5, PC Magazine, (accessed 4 May, 2002), http:/www.pcmag.com/article/0,2997,s=1671&a=4907,00.asp

Alwang, G., (2002a), IBM ViaVoice for Windows Pro USB Edition Release 9, PC Magazine, (accessed 4 May, 2002), http:/www.pcmag.com/article/0,2997,s=1671&a=22048,00.asp

Alwang, G., (2002b), Dragon Naturally Speaking Professional Solutions 6.0, PC Magazine, (accessed 4 May, 2002), http:/www.pcmag.com/article/0,2997,s=1671&a=22047,00.asp

Ancona, M., Dodero, G., Gianuzzi, V., (1999), Mobile Computing in Outdoor Environments (Extended Abstract), In Proceedings of the 1999 ACM Symposium on Applied Computing (SAC '99), San Antonio, Texas, USA, pg. 413 to 415

Arciniegas, F. A., (2001), *XML Developers Guide*, McGraw-Hill, New York

Aris, J., (2000), Inventing Systems Engineering, IEEE Annals of the History of Computing, Vol. 22, Issue 3, pg. 4 to 15

Aronoff, S., (1989), *Geographic Information Systems: A Management Perspective*, WDL Publications

Askenfelt, A., Gauffin, J., Sundberg, J., (1980), A comparison of Contact Microphone and Electroglottograph for the Measurement of Vocal Fundamental Frequency, Journal of Speech and Hearing Research, Vol. 23, Issue2, pg. 258 to 273

Autodesk Corporation, (2001), *Autodesk OnSite*, Autodesk - Autodesk® *OnSite*, (accessed June 9, 2001), http://www3.autodesk.com/adsk/section/0,,135815-123112,00.html

Babbie, E., (1990), *Survey Research Methods*, Wadsworth, Belmont, CA

Babin, D. N., Mazurenko, I. L., Urantsev, A. V., (1999), Use of Additional Information Sources in Identification and Recognition of Speech Response through Noise, Pattern Recognition and Image Analysis, Vol. 9, Issue 1, pg. 19 to 20

Bélanger, J., (2000), *The Influence of Employee Involvement on Productivity: A Review of Research*, Applied Research Branch of Strategic Policy, Human Resources Development Canada, R-00-4E

Bennet, P., (2000), The NMEA FAQ, (accessed 25 January, 2000), http://vancouver-webpages.com/peter/nmeafaq.txt

Bishr, Y., (1998), Overcoming the Semantic and Other Barriers to GIS Interoperability, International Journal of Geographical Information Science, Vol. 12, Issue 4, pg. 299 to 314

Blum, B. I., (1992), *Software Engineering: A Holistic View*, Oxford University Press

Brewster, S. A., (1994), Providing a Structured Method for Integrating Non-Speech Audio into Human-Computer Interfaces, PhD Thesis, University of York, U.K.

Buckingham, S., (2000), What is General Packet Radio Service?, (accessed 13 May, 2002), http://www.gsmworld.com/technology/gprs/intro.shtml

Budka, K. C., Jiang, H., Simmars, S. E., (1997), Cellular Digital Packet Data Networks, Bell Labs Technical Journal, Summer 1997, pg. 164 to 181

Burrough, P. A., McDonnell, R. A., (1998), *Principals of Geographic Information Systems*, Oxford University Press

CDG.org, (2002), What is CDMA? (accessed 13 May, 2002), http://www.cdg.org/tech/about_cdma.asp

Clampitt, P. G., DeKoch, R. J., Cashman. T., (2000), A Strategy for Communicating about Uncertainty, The Academy of Management Executive, Vol. 14, Issue 4, pg. 41 to 57

Córcoles, J. E., González, P., (2001), A Spatial Query Language over XML Documents, In Proceedings of the IASTED International Conference on Software Engineering and Applications, Anaheim, California, USA, pg. 1 to 6

Crawford, S.D., Couper, M.P., Lamias, M.J., (2001), Web Surveys: Perception of Burden, Social Science Computer Review, Vol. 19, pg. 146 to 162

Crawley, M., J., (2002), *Statistical Computing: An Introduction to Data Analysis and S-Plus*, John Wiley & Sons, Ltd, West Sussex, England

Dana, P. H., (1999), Global Positioning System Overview, The Geographers Craft, (accessed 14 May, 2002), http://www.colorado.edu/geography/gcraft/notes/gps/gps_f.html

Date, C. J., (2000), *An Introduction to Database Systems*, 7th Edition, Addison Wesley Longman, Inc.

Dàtria Systems, (2001), The Value of Speech Recognition, Speech Recognition Software for Mobile Workforces, (accessed April 23, 2001), http://www.datria.com/index.htm

Dàtria Systems, Inc., (2001), Voice Empowered Mobile Systems, Dàtria System's VoCarta Field, (accessed June 9, 2001), http://www.datria.com/products/field/index.htm

DB Network Technologies, Inc., (2001), Facilities, Inspection, Maintenance, Utilities Page, (accessed June 9, 2001), http://www.dbnt.com/utility/left_right.htm

de Knecht, J. P., Schavemaker, J. G. M., Reinders, M. J. T., Vossepoel, A. M., (2001), Utility Map Reconstruction, International Journal of Geographical Information Science, Vol. 15, Issue 1, pg. 7 to 26

Denys, P., (2002), Receiver Capabilities and Positioning Methods used by Current Day GPS Technology, New Zealand Surveyor, No. 292, pg. 27 to 34

Dhawan, C., (1997), *Mobile Computing: A System Integrator's Handbook*, McGraw-Hill

Director of Surveys, (2000), Standards, Specifications and Guidelines for Establishment and Maintenance of Alberta Survey Control Using GPS, Alberta Environment

Drashansky, T. T., Weerawarana, S., Joshi, A., Weerasinghe, R. A., Houstis, E. N., (1996), Software Architecture of Ubiquitous Scientific Computing Environments for Mobile Platforms, Mobile Networks and Applications, Vol. 1, Issue 4, pg. 421 to 432

Dragon Systems, (1999), NaturallySpeaking SDK Guide and Reference, Dragon Systems, Inc.

Drinnan, P. W., (2002) Field is Hooked . . . Cannot Live Without It!!, In The 2002 Gold Paper Collection: A Collection of Speaker Award Winning Papers from the Annual Geospatial Information and Technology Association Conference XXV, Tampa, FL, USA

Duran, S., (1997), *Keyword Spotting using Hidden Markov Models*, MSc Thesis, Bogaziçi University, (accessed July 4, 2002), Istanbul, Turkey http://www.busim.ee.boun.edu.tr/~speech/thesis/sevket_thesis.pdf

Egenhofer, M. J., (1999), Chapter 1 – Introduction: Theory and Concepts of *Interoperating Geographic Information Systems*, Edited by Goodchild, M. F., Egenhofer, M. J., Fegeas, R., and Kottman, C. A., Kluwer Academic Publishers

Egenhofer, M. J., Glasgow, J., Günther, O., Herring, J. R., Peuquet, D. J., (1999), Progress in Computational Methods for Representing Geographical Concepts, International Journal of Geographical Information Science, Vol. 13, Issue 8, pg. 775 to 796

Elsea, P., (1996), Microphones, University of California, Santa Cruz, Electronic Music Studio – Technical Writings, (accessed 7 May, 2002), http://arts.ucsc.edu/EMS/Music/tech_background/TE-20/teces_20.html

ESRI Inc., (2001), *ArcPad*, ArcPad, (accessed June 9, 2001), http://www.esri.com/software/arcpad/index.html

eMicrophones.com, (2002), Keys to High Speech Recognition Accuracy, (accessed 6 May, 2002), http://www.emicrophones.com/articles/Keys_to_dictation.asp

Falzon, K., Williamson, I., (2001), Digital Lodgement of Cadastral Survey Data in Australia – User Needs, Trans Tasman Surveyor, Vol. 1, No. 4, pg. 8 to 17

Fisher. P. F., (1999), Chapter 13 – Models of Uncertainty in Spatial Data in *Geographical Information Systems: Principles, Techniques, Management and Applications*, Edited by Longley, P. A., Goodchild, M. F., Maguire, D. J., Rhind, D. W., John Wiley & Sons, New York

Francica, J., (2000), Wireless Meets GIS, GEOWorld, Vol. 13, Issue 12, pg. 38 to 39

Francis. L., (1997), Mobile Computing - A Fact in Your Future, In Proceedings of the 15th Annual International Conference on Computer Documentation (SIGDOC '97), Snowbird, Utah, USA, pg. 63 to 67

Freedman, D., Pisani, R., Purves, R., (1998), *Statistics*, 3rd Edition, W.W. Norton & Company, New York

GE Smallworld, (2001), GE Smallworld Scout, GE Smallworld | Products | Spatial Technologies | Query, View & Print | GE Smallworld Scout, (accessed June 11, 2001), http://www.smallworld.co.uk/english/products/spatial/smallworldscout.asp

Geospatial Technology and Information Association, (2000), *The Geospatial Technology Report 2000 - A Survey of Organizations Implementing Geospatial Technologies*, The Information Technology Group of the Geospatial Technology and Information Association

Gong, Y., (1995), Speech Recognition in Noisy Environments, Speech Communication, Vol. 16, Issue 3, pg. 261 to 291

Goodchild, M. F., (1992), Geographic Data Modelling, Computers and Geosciences, Vol. 18, Issue ?, pg. 401 to 408

Goodchild, M. F., Egenhofer, M. J., Fegeas, R., (1998), Interoperating GIS: Technical Report of a Specialist Meeting held under the Auspices of the Varenius Project, National Center for Geographic Information and Analysis, University of California, Santa Barbara, USA

Graham, L. A., (2000), Life in the Fast Lane, GEOWorld, Vol. 13, Issue 7, pg. 30 to 35

GSM World, (2002), History of GSM, GSM Association, (accessed 13 May, 2002), http://www.gsmworld.com/about/history/index.shtml

GSM World, (2002a), High Speed Circuit Switched Data, GSM Association, (accessed 13 May, 2002), http://www.gsmworld.com/technology/hscsd/index.shtml

Haller, R., Mutschler, H., Voss, M., (1984), Comparison of Input Devices for Correction of Typing Errors in Office Systems, In B. Shackel, Ed. Proceedings of INTERACT 84, 1st IFIP Conference on Human Computer Interaction, London, UK, Vol. 2 pg. 218 to 223

Hamil, D. L., (2002), Your Mission Should You Choose to Accept It: Project management Excellence, (accessed April 9, 2002), http://spatialnews.geocomm.com/features/mesa1/

Harrington, A., (2000a), Consider the Differences among GPS Integration Technologies, GEOWorld, Vol. 13, Issue 2, pg. 26

Harrington, A., (2000b), Lessen Atmospheric and Multipath GPS Errors, GEOWorld, Vol. 13, Issue 12, pg. 26 to 27

Hofmann-Wellenhof, B., Lichteneggar, H., Collins, J., (1997), *Global Positioning System: Theory and Practice*, Springer-Verlag Wien, New York

Huang, X., Acero, A., Alleva, F., Beeferman, D., Hwang, M., Mahajan, M., (1994), From CMU Sphinx-II to Microsoft Whisper – Making Speech Recognition Usable, Microsoft Research, Redmond, WA, USA, MSR-TR-94-20

Hunt, J. J., Finkle, R. W., (2001), Work Orders and GIS in the Field; Quantifying Cost Savings at the City of Arlington, Texas, AM/FM/GIS Networks, Vol. 17, Issue 2, pg. 5 to 8

Hunter, A. J. S., (2000), The Road to Ubiquitous Geographic Information Systems Roam Anywhere - Remain Connected, Editor: Wigham, P.A., In Proceedings of the 12th Annual Colloquium of the Spatial Information Research Centre, Dunedin, New Zealand, pg. 47 to 56

Huxhold, W.E., Levinsohn, A. G., (1995), *Managing Geographic Information System Projects*, Oxford University Press

IBM, (2002), System Requirements, IBM ViaVoice for Windows, (accessed 4 May, 2002), http://www-3.ibm.com/software/speech/desktop/w9-adv-sr.html

Intel Corporation, (2001), The Benefits of Mobile Computing, The Benefits of Mobile Computing, (accessed May 24, 2001), http://www.intel.com/mobile/tbo/migration/benefits.htm

iMedeon, Inc., (2001), Improving The Connectivity of your Mobile Work Force for Greater Productivity, iMedeon.com > Solutions > Products > iM:Work > iM:Field, (accessed June 9, 2001), http://www.imedeon.com/solutions/products/iMField.html

Indian Towing Co. v. United States, 350 U.S. 61 (1955)

Intergraph Corporation, (2001), InService Suite Redefining Distribution and Workforce Management, Intergraph Energy - Integrated Outage, Dispatch, and Mobile Data Solutions, (accessed June 9, 2001), http://www.ingr.com/electric/is.asp

Jacobson, I., Booch, G., Rumbaugh, J., (1999), *The Unified Software Development Process*, Addison-Wesley

Jain. S., (2001), Introduction to Mobile Computing: The way the world is changing! Introduction to Mobile Computing: ACM Crossroad's Student Magazine (the ACM's first electronic publication), (accessed May 18, 2001), http://www.acm.org/crossroads/xrds7-2/intro72.html

Jecker, D., (1999), Benchmark Tests: Speech Recognition, PC Magazine, November 11

Johnson, R. A., (2000), *Miller & Freund's Probability and Statistics for Engineer*, 6th Edition, Prentice Hall, New Jersey

Jonas, M., Hillman, B., (2000), Take your Computer to the Field, GEOWorld, Vol. 13, Issue 11, pg. 32 to 34

Jones, D. M., Frankish, C. R., Hapeshi, K., (1992), Automatic Speech Recognition in Practice, Behaviour and Information Technology, Vol. 11, Issue 2, pg. 109 to 122

Junqua, J., (1993), The Lombard Reflex and its Role on Human Listeners and Automatic Speech Recognition, Journal of the Acoustical Society of America, Vol. 93, Issue 1, pg. 510 to 524

Karimi, H. A., Krishnamurthy, P., Banerjee, S., Chrysantheis, P. K., (2000), Distributed Mobile GIS, GIM International, Vol. 14, Issue 9, pg. 80 to 83

Kaye B.K., Johnson T.J. (1999), Research Methodology: Taming the Cyber Frontier, Social Science Computer Review, Vol. 17, pg. 323 to 337

Keizer, G., (2002), CNet Review: Dragon Naturally Speaking Preferred 6.0, CNet.com, (accessed 5 May, 2002), http://cnet.com/software/0-3227838-1232-9090735.html?tag=subnav

Kottman, C., (1999), Chapter 4 - The Open GIS Consortium and Progress toward Interoperability in GIS of *Interoperating Geographic Information Systems*, Edited by Goodchild, M. F., Egenhofer, M. J., Fegeas, R., and Kottman, C. A., Kluwer Academic Publishers

LaForge, P. M., (2001), The 3G Evolution at High Speed, Guest Column, CDG.org, (accessed 13 May, 2002), http://www.cdg.org/resource_center/GuestCol/laforge_april_01.asp

Lake, R., (2000), Introduction to GML Geography Markup Language – Part 1 in the GML Series, (accessed 28 May, 2002), http://www.jlocationservices.com/company/galdos/articles/introduction_to_gml.htm

Landgraf, G., (1999), Evolution of EO/GIS Interoperability towards an Integrated Application Infrastructure, in Proceedings of Interoperating Geographic Information Systems: Second International Workshop, INTEROP '99, Zurich, Switzerland, pg. 27 to 40

Land Information New Zealand (2002a), Data Characteristics of DCDB, (accessed 2 July, 2002), http://ww2.linz.govt.nz/newinternet/dcdb/dataquality.htm

Lathi, B. P., (1992), *Linear Systems and Signals*, Cambridge Press, Berkeley

Leick, A., (1995), *GPS Satellite Surveying*, John Wiley & Sons, Inc. New York

Leclercq, E., Benslimane, D., Yétongnon, K., (1999), ISS: A Semantic Mediation Model and an Agent Based Architecture for GIS Interoperability, in Proceedings of the International Database Engineering and Applications Symposium, Montreal, Canada, pg. 87 to 91

Levinsohn, A., (2000), *Geospatial Interoperability: The Holy Grail of GIS*, GEOWorld, Vol. 13, Issue 10, pg. 28 to 29

Licklider, J.C.R., (1960), Man-Computer Symbiosis, IRE Transactions of Human Factors in Electronics, Vol. HFE-1, pg. 4 to 11

Liu, G., Marlevi, A., Maguire, G. Q. Jr., (1995), A Mobile Virtual Distributed System Architecture for Supporting Wireless Mobile Computing and Communications, In Proceedings of the First Annual International Conference on Mobile Computing and Networking (MOBICOM '95), Berkeley, California, USA, pg. 111 to 118

Lo, Y., Hara, T., Tsukamoto, M., Nishio, S., (2000), A Hybrid Method for Concurrent Updates on Disconnected Databases in Mobile Computing Environments, In Proceedings of the 2000 ACM Symposium on Applied Computing 2000 (Volume 2) (SAC '00), Como, Italy, pg. 563 to 565

Locke, J., (1689), *An Essay Concerning Human Understanding*, 1st Edition, Oxford University Press

Longley, P. A., Brooks, S. M., McDonnell, R. MacMillan, B., (1998), *Geocomputation: A Primer*, John Wiley and Sons Ltd.

Longley, P. A., Goodchild M. F., Maguire, D. J., Rhind, D. W. (Editors), (1999), *Geographical Information Systems: Principles, Techniques, Management, and Applications*, John Wiley & Sons Ltd.

Lucent Technologies, (2002), Bell Labs Technology: Early mobile telephone service, (accessed 12 May, 2002), http://www.bell-labs.com/technology/wireless/earlyservice.html

M3i Systems Inc., (2001), Geobase, Products - PragmaCAD – Geobase, (accessed June 9, 2001), http://www.m3isystems.com/m3iweb/Products/WorkforceManagement/geobase.htm

McIlroy, M. D., Pinson, E. N., Tague, B. A., (1978), UNIX Time-Sharing System: Foreword, The Bell System Technical Journal, Vol. 57, Issue 6, pg. 1899 to 1904

MapInfo Corporation, (2001), MapInfo MapInHand, MapInfo Software, (accessed June 9, 2001), http://dynamo.mapinfo.com/products/web/Overview.cfm?ProductID=42

MapFrame Corporation, (1998), Mobile Computing: Breaking Barriers to Field Automation, MapFrame Corporation Whitepaper

MapFrame Corporation, (2001), FieldSmart Specifications, (accessed June 9, 2001), Specifications, http://www.mapframe.com/specs.htm

Mann, S., (1998), Wearable Computing as means for Personal Empowerment, Keynote Address for The First International Conference on Wearable Computing (ICWC-98), May 12 to 13, Fairfax, Vaginia, USA, (accessed May 29, 2001), http://wearcam.org/icwckeynote.html

Mark, D. M., (1999), Chapter 7 - Spatial Representations: A Cognitive View in *Geographical Information Systems: Principles, Techniques, Management and Applications*, Edited by Longley, P. A., Goodchild, M. F., Maguire, D. J., Rhind, D. W., John Wiley & Sons, New York

Martin, D. J., (1999), Chapter 6 - Spatial Representations: The Social Scientist's Perspective in *Geographical Information Systems: Principles, Techniques, Management and Applications*, Edited by Longley, P. A., Goodchild, M. F., Maguire, D. J., Rhind, D. W., John Wiley & Sons, New York

Medin, C., Roy, S., Ann, T., (1999), World Wide Web versus mail surveys: A comparison and report, Paper presentation at ANZMAC99 Conference, Marketing in the Third Millennium, Sydney, Australia, (Accessed May 28,2001), http://www.anzmac99.unsw.edu.au/anzmacfiles/papers.htm

Meneghello, M., (2001), XML (eXtensible Markup Language) – The New Language of Data Exchange, Cartography, Vol. 30, Issue 1, pg. 51 to 57

MGS-Mobile GIS Systems, (2001), Skypower: The Ultimate, Portable Positioning and Video Mapping System, mgs - mobile gis systems, (accessed June 9, 2001), http://www.mgs-mobile.com/index.htm

Microsoft Corporation, (1999), A Beginners Guide to the XML DOM, Microsoft Development Network, (accessed 30 May, 2002), http://msdn.microsoft.com/library/default.asp?url=/nhp/Default.asp?contentid=28000438

Microsoft Corporation, (1999a), *White Paper: Virtual Private Networking in Windows 2000: An Overview*, Microsoft Corporation

Microsoft Corporation, (2000), Microsoft Speech SDK Version 5.0 User Manual, Microsoft Corporation

Microsoft Corporation, (2002a), Design Guidelines for Voice User Interfaces, Microsoft .Net Speech Technologies, (accessed 3 May, 2002), http://www.microsoft.com/speech/techInfo/whitepapers/papers/designguide.asp

Microsoft Corporation, (2002b), Programming Approached to Speech-Enabled Applications, Microsoft .Net Speech Technologies, (accessed 3 May, 2002), http://www.microsoft.com/speech/techInfo/whitepapers/papers/programming.asp

Microsoft Corporation, (2002c), About the W3C Document Object Model, Microsoft Development Network, (accessed 30 May, 2002), http://msdn.microsoft.com/library/default.asp?url=/workshop/author/dom/domoverview.asp

Microsoft Corporation, (2002d), SOAP, Microsoft Development Network, (accessed 30 May, 2002), http://msdn.microsoft.com/library/default.asp?url=/nhp/Default.asp?contentid=28000523

Microspeech.com, (2002), Ask Tech Guy, (accessed 4 May, 2002), http://www.microspeech.com/asktechguy/tips.shtml

MobileInfo.com, (2001a), Handheld Topics: Unique Application Integration Issues, MobileInfo PDA, Palm, Win CE, Smart Telephone Unique Application Issues, (accessed May 15, 2001a), http://www.mobileinfo.com/Handheld/unique_issues.htm

MobileInfo.com, (2001b), Mobile Computing Hardware & Software Components, MobileInfo Hardware & Software Components, (accessed May 15, 2001b), http://www.mobileinfo.com/hwsw_components.htm

MobileInfo.com, (2001c) Mobile Computing Solution Benefits, MobileInfo Solution Benefits, (accessed May 15, 2001c), http://www.mobileinfo.com/solution_benefits.htm

Montgomery, G. E., Schuch, H. C., (1993), *GIS Data Conversion Handbook*, GIS World Books

Murray, A. C., Jones, D. M., Frankish, C. R., (1996) Dialogue Design in Speech-mediated Data-entry: The Role of Syntactic Constraints and Feedback, International Journal of Human-Computer Studies, Vol. 45, pg. 263 to 286

National Inventors Hall of Fame, (2001), Guglielmo Marconi: Transmitting Electrical Signals, Guglielmo Marconi, (accessed June 13, 2001), http://www.invent.org/search/71.html

Nelson, M. G., (2000), Fast Is No Longer Fast Enough, Information Week Online, (accessed 2 July, 2002), http://www.informationweek.com/789/web.htm

Neumann, K., Maskarinec, M., (1997), Mobile Computing within a Distributed Deductive Database, In Proceedings of the 1997 ACM Symposium on Applied Computing (SAC '97), San Jose, California, USA, pg. 318 to 322

NIST/SEMATECH, (2002), *e-Handbook of Statistical Methods*, (accessed 20 September, 2002), http://www.itl.nist.gov/div898/handbook/

NMT Corporation, (2001), FAAR, NMT.com, (accessed June 9, 2001), http://www.nmt.com/nmtgen.asp?var=faar

Onsrud, H.J., (1999), Liability in the Use of GIS and Geographical Datasets, in *Geographical Information Systems: Principles, Techniques, Management and Applications*, Edited by Longley, P. A., Goodchild, M. F., Maguire, D. J., Rhind, D. W., John Wiley & Sons, New York

OpenGIS Consortium Inc., (1998), *The OpenGIS™ Guide (Draft) Third Edition*, (accessed 28 May, 2002), http://www.opengis.org/techno/guide/guide980615/toc.htm

OpenGIS Consortium Inc., (1999a), *The OpenGIS™ Abstract Specification Topic 0: Abstract Specification Overview, Version 4*, OpenGIS Project Document Number 99-100r1, Open GIS Consortium, Inc.

OpenGIS Consortium Inc., (1999b), *The Open GIS™ Abstract Specification, Topic 1: Feature Geometry, Version 4*, OpenGIS Project Document Number 99-101, Open GIS Consortium, Inc.

OpenGIS Consortium Inc., (1999c), *The Open GIS™ Simple Features Specification for OLE/COM Revision 1.1*, OpenGIS Project Document Number 99-050, Open GIS Consortium, Inc.

OpenGIS Consortium Inc., (1999d), *The Open GIS™ Abstract Specification Topic 2: Spatial Reference Systems, Version 4*, OpenGIS Project Document Number 99-102r1, Open GIS Consortium, Inc.

OpenGIS Consortium Inc., (2001), *The Open GIS™ Abstract Specification, Topic 1: Feature Geometry, (ISO 19107 Spatial Schema), Version 5*, OpenGIS Project Document Number 01-101, Open GIS Consortium, Inc.

OpenGIS Consortium Inc., (2002a), *OpenGIS® Geography Markup Language (GML) Implementation Specification, Version 2.1.1*, OpenGIS Project Document Number 02-009, Open GIS Consortium, Inc.

OpenGIS Consortium Inc., (2002b), *OpenGIS Simple Features Specification*, (accessed 28 May, 2002), http://www.opengis.org/pressrm/summaries/20010219.TS.SimpFeat.htm

Out-loud.com, (2002), Computing Out Loud, (accessed 7 May, 2002), http://www.out-loud.com/index.html

Ostroff, J. S., Paige, R. F., (2000), The Logic of Software Design, IEE Proceedings – Software, Vol. 147, Issue 3, pg. 73 to 79

Oviatt, S., MacEachern, M., Levow, G., (1998), Predicting Hyperarticulate Speech during Human-Computer Error Resolution, Speech Communication, Vol. 24, Issue 2, pg. 87 to 110

Oviatt, S., (2000), Taming Recognition Errors with a Multimodal Interface, Communications of the ACM, Vol. 43, Issue 9, pg. 45 to 51

Peuquet, D. J., (1999), Chapter 8 – Time in GIS and Geographical Databases in *Geographical Information Systems: Principles, Techniques, Management and*

Applications, Edited by Longley, P. A., Goodchild, M. F., Maguire, D. J., Rhind, D. W., John Wiley & Sons, New York

Pocket Systems Ltd., (2001), Welcome to Pocket Systems…, Pocket Systems Ltd – PocketGIS, (accessed June 11, 2001), http://www.pocket.co.uk/

PointBase, Inc., (2001), *PointBase Micro*, PointBase, (accessed June 9, 2001), http://www.pointbase.com/node.shtml?navHier=Products/PointBase+Micro&CF=products/micro.html

Pray, M., Bourgeois, S., Lawless, D., (1998), Collecting Data at the Speed of Sound, In the 18th Annual ESRI International User Conference, San Diego, California, (accessed 2 May 2002), http://gis.esri.com/library/userconf/proc98/PROCEED/TO650/PAP607/P607.HTM

Qualcomm, (2002), About CDMA, (accessed 13 May, 2002), http://www.qualcomm.com/cdma/

Radovanovic, R.S. (2000). High Precision Deformation Monitoring by Multipath Mitigation via Day-To-Day Correlation Analysis. Proceedings of ION-GPS 2000, Salt Lake City, USA. (accessed September 19,2002), http://www.ucalgary.ca/~rsradova/IONmod.pdf

Raper, J. F., (1999), Chapter 5 – Spatial Representation: The Scientist's Perspective in *Geographical Information Systems: Principles, Techniques, Management and Applications*, Edited by Longley, P. A., Goodchild, M. F., Maguire, D. J., Rhind, D. W., John Wiley & Sons, New York

Reminga v. United States, 695 F. 2d 1000 (6th Cir. 1982)

Reichardt, M., (2001), OGC's GML 2.0: A New Wave of Open Geoprocessing on the Web, GeoInformatics, July/August Issue 4, pg. 18 to 21

Reid, H., Francica, J., (2000), Thin is In, GEOWorld, Vol. 13, Issue 6, pg. 38 to 42

Rhodes, B. J., (1997), The Wearable Remembrance Agent: A System for Augmented Memory, Personal Technologies, Issue 1, pg. 218 to 224

River Run Software Group, (2001), *White Paper: Field Inspection Application*, River Run Software Group: Field Inspection White Paper, (accessed June 4, 2001), http://www.riverrun.com/whitepapers/whitepaper-field.html

Robinson, A. H., Sale, R. D., Morrison, J. L., Muehrcke, P. C., (1984), *Elements of Cartography*, 5th Edition, John Wiley & Sons Ltd.

Rysavy, R., (1999), The Evolution of Cellular Data: On the Road to 3G, Rysavy Reseach, (accessed 12 May 12, 2002), http://www.rysavy.com/Articles/3G/3g.htm

SISA, (2002), Bonferroni Correction Online. Adjustment for Multiple Comparisons, (accessed 20 September, 2002), http://home.clara.net/sisa/bonhlp.htm

SSH Communications Security Corp., (2001), *White Paper: SSH Secure Shell, Version 1*, SSH Communications Security Corp

Satyanarayanan, M., (1996), Fundamental Challenges in Mobile Computing, In Proceedings of the Fifteenth Annual ACM Symposium on Principles of Distributed Computing (PODC '96), Philadelphia, PA, USA, pg. 1 to 7

ScanSoft, (2002), Dragon Naturally Speaking 6.0, Professional Solutions: The Professional Choice for Voice, ScanSoft Inc.

Shneiderman, B., (2000), The Limits of Speech Recognition, Communications of the ACM, Vol. 43, Issue 9, pg. 63 to 65

Sollenberger, N., R., Seshadri, N., Cox, R., (1999), The Evolution of IS-136 TDMA for Third Generation Wireless Services, IEEE Personal Communications, Vol. 6, Issue 3, pg. 8 to 18

Sondheim, M., Gardels, K., Bueheler, K., (1999), Chapter 24 – GIS Interoperability in *Geographical Information Systems: Principles, Techniques, Management and Applications*, Edited by Longley, P. A., Goodchild, M. F., Maguire, D. J., Rhind, D. W., John Wiley & Sons, New York

SoundBlaster.com, (2002), Detailed Product Comparison, Creative Technologies Ltd., (accessed 8 May 2002), http://www.soundblaster.com/compare/DetailedCompare.asp

Spencer, A. L. Jr., (2001), Benefits of Geospatial Operations Support Systems for the Communications Industry, In The 2001 Gold Paper Collection: A Collection of Speaker Award Winning Papers from the Annual Geospatial Information and Technology Association Conference XXIV, San Diego, CA, USA

Strayer, D., Drews, F., Albert, R. and Johnston, W. (2001) *Does Cell Phone Conversation Impair Driving Performance?* (9/10/01), National Safety Council | Cellphone Article, http://www.nsc.org/library/shelf/inincell.htm

Survey Quarterly, (2002), Landonline Makes its Mark, Issue No. 29, March, pg. 25

Tadpole-Cartesia, (2001), Products – Conic Solutions, Tadpole-Cartesia - Conic Solutions, (accessed June 9, 2001), http://www.tadpole.com/car/index.htm

Taylor, J., Henderson, R., (2001), Utilities Turning Off Radios, Picking Up Phones, Energy IT, Vol. 6, Issue 3, pg. 66 to 69

TDS GIS Solutions Inc., (2001), Start Collecting Field Data the Smart Way, Fieldnotes 32, (accessed June 11, 2001), http://www.penmetrics.com/products.htm

Telus Mobility, (2002), Product & Services / Wireless Web / Mobile Computing / Velocity Wireless - TELUS Mobility – Alberta, (accessed 5 July, 2002), http://www.telusmobility.com/ab/wweb/velocity_wireless.shtml#b

The City of Calgary, (2001), Road Maintenance, City of Calgary - Engineering and Environmental Services - Street Smarts, (accessed April 24, 2001), http://www.gov.calgary.ab.ca/eesd/streets/mtce_mission.html

Thierrin, R. D., (1999), Infrastructure and Asset Management using Mobile Technology, Geospatial Information Technology Association Conference XXII, Charlotte, NC, USA, pg. 112 to 119

Trimble Navigation Ltd, (2002), Differential GPS, How does Differential GPS Work?, (accessed 15 May, 2002), http://www.trimble.com/gps/how2.html

Trimble Navigation Ltd, (2002a), Summary of GPS Error Sources, (accessed 17 May, 2002), http://www.trimble.com/gps/errorsources.html

Trubridge Callender Beach Ltd., (1999), *Data Characteristics: Data Conversion of Water Network Record – Johnsonville Portion*, Report to Wellington City Council

Tyfa, D. A., Howes, M., (2000), Speech Recognition for Command Entry in Multimodal Interaction, International Journal of Human-Computer Studies, Vol. 52, pg. 637 to 667

Vassiliou, M. S., Sundareswaran, V., Chen, S., Behringer, R., Tam, C., Chan, P., Bangayan, P., McGee, J., (2000), Integrated Multimodal Human-Computer Interface and Augmented Reality for Interactive Display Applications, In Proceedings of SPIE: Cockpit Displays VII: Displays for Defence Applications – The International Society of Optical Engineering, Orlando, FL, USA, Vol. 4022, pg. 106 to 115

van der Vlugt, M., (1999), OpenGIS and Interoperability in Europe: The GIPSIE Project, In the Proceedings of Geographical and Spatial Information Management; Improving Access to Better Information, London, UK, pg. 8/11/1 to 8/11/3

Varshney, U., Vetter, R., (2000), Emerging Mobile and Wireless Networks, Communications of the ACM, Vol. 43, Issue 6, pg. 73 to 81

Walter, V., Fritsch, D., (1999), Matching Spatial Data Sets: A Statistical Approach, International Journal of Geographical Information Science, Vol. 13, Issue 5, pg. 445 to 473

World Wide Web Consortium, (2000), Document Object Model (DOM) Level 1(Second Edition) Specification, (29 September, 2000), Version 1.0, (accessed 30 May, 2002), http://www.w3.org/TR/2000/WD-DOM-Level-1-20000929/

World Wide Web Consortium, (1999), Namespaces in XML (14 January 1999), (accessed 28 May, 2002), http://www.w3.org/TR/1999/REC-xml-names-19990114/

World Wide Web Consortium, (2001a), XML Schema Part 1: Structures. W3C Candidate Recommendation (2 May 2001), (accessed 28 May, 2002), http://www.w3.org/TR/xmlschema-1/

World Wide Web Consortium, (2001b), *XML Schema Part 2: Datatypes. W3C Candidate Recommendation (2 May 2001)*, (accessed 28 May, 2002), http://www.w3.org/TR/xmlschema-2/

Weber, S. P., (2000), The Last Frontier for GIS Data, In The 2000 Gold Paper Collection: A Collection of Speaker Award Winning Papers from the Annual Geospatial Information and Technology Association Conference XXIII, Denver, CO, USA, pg. 50 to 54

Weintraub, M., Taussig, K., Hunicke, K., Snodgrass, A., (1996), Effect of Speaking Style on LVCSR Performance, In Proceedings of the International Conference on Spoken Language Processing, Philadelphia, USA, Vol. 1 pg. 16 to 19

Weiser, M., (2000a), Ubiquitous Computing, Ubiquitous Computing, (accessed Aug. 26, 2000a), http://www.ubiq.com/hypertext/weiser/UbiHome.html

Weiser, M., (2000b), Open House, Interactive Telecommunications Program of New York University, ITP Review 2.0., (accessed Aug. 9, 2000b), (http://www.itp.tsoa.nyu.edu/~review/) http://www.ubiq.com/hypertext/weiser/WeiserPapers.html

Weiser, M., (1991), The Computer for the 21st Century, Scientific American, September, pg. 94 to 104.

Wiederhold, G., (1999), Mediation to Deal with Heterogeneous Data Sources, in Proceedings of Interoperating Geographic Information Systems: Second International Workshop, INTEROP '99, Zurich, Switzerland, pg. 1 to 16

Wilson, J. D., (1998), GIS Goes Mobile, GIS World, Vol. 11, Issue 12, pg. 54 to 57.

Wilson, J. D., (2000), Mobile Technology takes GIS to the Field, GeoWorld, Vol. 13, Issue 6, pg. 32 to 36

World Wide Web Consortium, (2000), About the World Wide Web, (accessed 28 May, 2002), http://www.w3.org/Consortium/

Xybernaut Corporation, (2001), Product Information: Mobile Assistant IV, (accessed June 4, 2001), http://www.xybernaut.com/product/prod_des.htm

Yacoub, M. D., (1993), *Foundations of Mobile Radio Engineering*, CRC Press LLC

Zimmerman, J. B., (2001), Mobile Computing: Characteristics, Business Benefits, and the Mobile Framework, Mobile Computing: Characteristics, Business Benefits, and the Mobile Framework, (accessed May 24, 2001), http://faculty.ed.umuc.edu/~meinkej/inss690/zimmerman/INSS 690 CC - Mobile Computing.htm

Appendix A – Data Accuracy Requirements Survey

Data Accuracy Requirements Survey

UNIVERSITY OF CALGARY

GEOMATICS ENGINEERING

About this Survey

As partial fulfilment towards a Master of Science Degree at the University of Calgary, I am investigating the use of Mobile GIS as a data acquisition tool. One of the research components is to investigate the ability of a Mobile GIS platform to meet the spatial accuracy needs of end users. **Therefore, as a user your assistance is very important**.

It is recognized that organizations may use a wide variety of data, which encompass a number of accuracy specifications. When completing this survey please use typical examples from your organization.

Confidentiality

Participation in the survey is ***voluntary***, and responses to this survey are ***strictly confidential***. Contact information has been requested in order that further consultation may be entered into if the participant indicates that they are willing to do so.

It is the Universities policy that all individuals participating in a survey be informed of the purpose and use of solicited information. The following is furnished to fulfill this requirement:

Informed Consent

This Consent form is only part of the process of informed consent. It should give you the basic idea of what the research is about and what your participation will involve. If you would like more detail about something mentioned here, or information not included here, you should feel free to ask. Please take the time to read this carefully and to understand any accompanying information.

As mentioned above the purpose of this research is to investigate data accuracy requirements of the GIS user community so as to determine the appropriateness of Mobile GIS for different data acquisition activities.

This Web-based survey is intended for Utility, Local, Provincial and Federal Government GIS project managers. The survey consists of fourteen research questions that are designed to ascertain the difference between the accuracy of data that is used by these sectors of the GIS community, and the accuracy that they desire. The survey should take approximately five (5) minutes to complete.

There are no known harms associated with your participation in this research.

Confidentiality will be respected. Data will be kept within a password protected database to which only the researcher has access. No information that discloses your identity will be released or published without your specific consent to disclose.

You will not benefit directly from participation in this research.

You have the right to refuse to participate or to withdraw from this study at any time.

By clicking on the **Go to Survey** button on this form it indicates that you have understood to your satisfaction the information regarding participation in the research project and agree to participate as a subject. In no way does this waive your legal rights nor release the investigators, sponsors, or involved institutions from their legal and professional responsibilities. You are free to withdraw from the study at any time. Your continued participation should be as informed as your initial consent, so you should feel free to ask for clarification or new information throughout your participation. If you have further questions concerning matters related to this research, please contact:

Andrew Hunter
MSc Candidate
Department of Geomatics Engineering
University of Calgary
Calgary
Canada
Ph:+(403) 220 7106
ahunter@ucalgary.ca

Dr. C. V. Tao
Supervisor
Department of Geomatics Engineering
University of Calgary
Calgary
Canada
Ph:+(403) 220 5826
ctao@ucalgary.ca

Dr. B. Ballantyne
Supervisor
Department of Geomatics Engineering
University of Calgary
Calgary
Canada
Ph:+(403) 220 7105
bballan@ensu.ucalgary.ca

If you have any questions or issues concerning this project that are not related to the specifics of the research, you may also contact the Research Services Office at +(403) 220 3782 and ask for Mrs. Patricia Evans.

To begin the survey, please click here:
If you do not wish to proceed, please click here:

Thank you for your participation!

Andrew Hunter
Andrew Hunter's web page

Created by Andrew Hunter, 11 May 2001

UNIVERSITY OF CALGARY

Data Accuracy Requirements Survey

GEOMATICS ENGINEERING

Your Details (All fields are required):

First Name:

Last Name:

Email Address:
example: scooby@dooby.doo

Company:

Phone Number:
example: xxx-xxx-xxxx for North America and NZ
or xxx-xxxx-xxxx for Australia

Country:
Select one

Are you willing to be contacted? Select one

Position:

Industry: Select one

Purpose of GIS within your organization:

Survey:

1) What is the relative positional accuracy of your spatial data?

2) Is this adequate for your applications?

3) Given the cost of acquiring accurate data, what accuracy specification would be ideal for your GIS applications?

4) Regardless of cost, what accuracy specification would be ideal for your GIS applications?

5) What was the primary data acquisition method used during your conversion process for the spatial component of your GIS?

6) If field surveys were not the primary data acquisition method used during your conversion process what was the source of your data?

7) What is the primary method of updating your spatial data?

8) What type of reference framework was used during the conversion process?

If "Other" was selected above, please identify:

9) How long did it take to complete your initial data conversion process?

10) Was field validation undertaken?

11) Are you considering improving the accuracy of your spatial data?

If "Yes" explain why:

12) Has Conflation (rubber sheeting) been used to align/merge spatial data?

If "Yes", where the results adequate?

If "No" explain why not:

13) How long does it take for amendments to data to be made available for use in your organization?

14) How long would you like it to take for amendments to be made available to your organization?

Submit Reset

Created by Andrew Hunter, 12 June 2001

APPENDIX B - EXISTING MOBILE GIS APPLICATIONS

Company	ESRI Inc., Redlands, CA, USA
Available Since	1998/1999
Technology	ArcPad is promoted as being hardware independent, however it only runs on Windows CE 2.11 or higher, 95/98, NT, and 2000 operating systems. With the release of ArcPad 5.0.1, ESRI's mobile application now supports the downloading of spatial data via a TCP/IP connection (wireless, cellular or LAN). ArcPad 5.0.1 can also act as a client to ESRI's Internet mapping and GIS software, ArcIMS. Standard map navigation, feature identification and editing features are incorporated along with the ability to utilize hyperlinks to external files, query features are also provided to determine areas, distances and directions. Input for data editing is via redlining using a stylus (mobile device requires touch sensitive screen), cursor or GPS (can only use GPS receivers that can output NMEA and TSIP formatted sentences), allowing real-time positioning in the field. The user is also provided with functionality required to generate input forms on the fly. ArcPad also provides integration tools for the import/export of ArcView projects.
Web Address	http://www.esri.com/software/arcpad/index.html
Company	AutoDesk Corp., San Rafael, CA, USA
Available Since	?
Technology	OnSite allows users to capture information in the field by drawing (redlining) on the mobile device screen and adding text to the new features. Synchronization facilities are included so that databases can be updated at a later date based on notes and redlines compiled by the user while in the field.

	Onsite has been designed to run on one of the Palm Vx series mobile devices running Palm OS 3.3, or on a Fujitsu PenCentra 130, or Symbol PPT 2700 for WinCE (v.2.11 and up) users. In order for OnSite to connect to remote databases it requires access to a java based servlet (Jrun) and AutoDesk's MapGuide 4. OnSite is built around Oracles 8i Lite database management system and supports wireless and wired connections to remotes spatial and non-spatial databases, however it does not allow real-time access to remote data. OnSite incorporates synchronization facilities as described above, standard GIS navigation features; feature selection and identification; standard symbol libraries; data acquisition by redlining with notes; coordinate transformation facilities; and data transfer security features.
Web Address	http://www.autodesk.com/profctr/gis/gisindex.htm
Company	Tadpole-Cartesia, Carlsbad, CA, USA
Available Since	1996
Technology	Conic GIS is a field data capture software product designed to function on a pen-based computer that consists of four integrated modules: Conic View; Conic Redline; Conic Query; and Conic Capture. Conic View provides map viewing and navigation functionality of spatial datasets. Conic Redline allows users to interactively annotate digital data with a stylus. Conic Query provides facilities to view, capture and edit feature attributes. Conic Capture includes tools to update or add features to existing spatial datasets. It also includes tools that allow a mobile device to be linked to survey equipment such as GPS, Electronic Total Stations and Laser Rangefinders, from which the location of new features can be acquired. Through the use of a wireless modem, real-time data communications between users in mobile and office environments can be implemented in order that data and job information can be updated on the fly.

Web Address	http://www.tadpole.com/car/index.htm
Company	MapInfo Corporation, Troy, New York, USA
Available Since	August 2000
Technology	MapinHand has been developed for PDA's using Palm Computing 3.0 (or higher) or Windows CE 2.0 (or higher) operating systems. MapinHand uses MapInfo's MapXtreme® Web software to provide real-time access to external databases and spatial information in Oracle and MapInfo formats. MapinHand has been built around Oracle8i™. As with other applications MapinHand provides map navigation, query, data editing (redlining) functionality. Geocoding tools are also provided to assist with the location of features in spatial datasets. MapinHand is primarily a Web based application and therefore requires access to java-based servlets that manage information requests.
Web Address	http://dynamo.mapinfo.com/products/web/Overview.cfm?ProductID=42
Company	Dàtria Systems, Inc., Englewood, CO, USA
Available Since	1997
Technology	VoCarta Field is a speech enabled data acquisition application. Captured features are stored in a standard database file that can be uploaded to any relational or spatial database. VoCarta Field includes development tools that can be used to modify the system to best suit a projects data acquisition requirements. This includes tools for building and editing vocabularies (or sets of words appropriate to the data), quality assurance tools, database administration tools, and tools to connect to other surveying and field inspection technologies such as GPS, laser range finders, digital cameras, Digital Measuring Instrument (DMI) and bar-code readers. When incorporated with VoCarta TeleForms external databases can be accessed via a cellular connection and be edited while on the move.

Web Address	http://www.datria.com/
Company	iMedeon, Inc., Alpharetta, GA, USA
Available Since	iM:Collect – July 2000; iM:Field – March 2001
Technology	iM:Collect has been developed as a mobile data collection solution for use on Palm Operating System devices and Windows 95/98/CE and NT operating systems. iM:Collect works in conjunction with iM:Work, iMedeon's wireless web application. iM:Collect includes an administration tool, which allows users to configure the system to meet data acquisition requirements. Pre-configured templates can be utilised to accelerate the creation of data collection objects. The system also manages version control and push/pull synchronization with external databases. iM:Collect is essentially a mobile database application and does not provide any mapping functionality. iMedeon has also developed iM:Field which is a wireless enabled web base application with GIS capabilities that can query enterprise databases in real-time and can be connected to a GPS unit.
Web Address	http://www.imedeon.com/
Company	PointBase, Inc., Mountain View, CA, USA
Available Since	PointBase Micro: June, 2001; Mobile Edition: 2000
Technology	Mobile Edition is a database management system developed for Internet applications and mobile devices. Mobile Edition is a Pure Java object relational database and is customizable with any of the major Java based application development tools. Mobile Edition does not provide any mapping tools. Mobile Edition can be implemented on any computing device that has a Java Virtual Machine.

	PointBase Micro is specifically designed for mobile computing applications based upon the J2ME or J2SE architectures. It has been specifically designed for running on a PDA but will run on Windows, Windows CE/Pocket PC, Symbian EPOC, Palm OS, Motorola, and iDen.
Web Address	http://www.pointbase.com/home.shtml
Company	Intergraph Corporation, Huntsville, Al, USA
Available Since	March 2000
Technology	InService is a mobile GIS and work management system that can interface with Supervisory Control and Data Acquisition Systems (SCADA); Customer Information Systems (CIS); corporate Work Management Systems (WMS); Automated Field Detection Devices; and Automatic Vehicle Location (AVL) systems. InService runs on Windows based operating systems and requires an Oracle Database for data storage.
Web Address	http://www.ingr.com/electric/is.asp
Company	M3i Systems Inc., Montreal, Quebec, Canada
Available Since	1993
Technology	The Pragma family of products is designed as an outage management system for electrical and telecommunication and organizations. The system includes a mobile GIS unit for maintaining the geographic elements of electrical or communication networks. M3i also caters to various public safety organizations, their initial installation being in 1993 for the Police Department at St-Jean-sur-le Richelieu, Quebec.
Web Address	http://www.m3isystems.com/M3iWeb/Index.htm
Company	MapFrame Corporation, Dallas, TX, USA
Available Since	1998
Technology	FieldSmart, MapFrame’s mobile field solution has been developed for

	Windows 95/98/NT/2000/CE operating systems. FieldSmart provides all basic GIS viewer functions such as map navigation, the ability to handle vector and raster based data, basic distance and area calculations and simple database queries. Editing is performed via redlining and the addition of annotations. FieldSmart Connect provides wireless connection to corporate databases.
Web Address	http://www.mapframe.com/
Company	DB Network Technologies, Inc., Pomona, CA, USA
Available Since	?
Technology	Integrated facilities inspection and maintenance system for delivery of GIS and other data to the field
Web Address	http://www.dbnt.com/index_flash.htm
Company	NMT Corporation, La Crosse, WI, USA
Available Since	1998
Technology	FAAR is NMT's platform-independent GIS viewing software. FARR combines a GIS system with a CIS database. FARR is currently used by Gas, Electric and Telecom organizations for fault detections and repair.
Web Address	http://www.nmt.com/nmthome.htm
Company	GE Smallworld, Cambridge, UK
Available Since	?
Technology	Smallworld Scout is a Windows based GIS that allows query/view and print operations to be performed on a Smallworld database via Smallworld's Mobile Data Server. The Mobile Data Server allows a spatial subset of a database to be replicated on a mobile device for use in the field.
Web Address	http://www.smallworld.co.uk/

Company	TDS GIS Solutions Inc., Portland, OR, USA
Available Since	?
Technology	Fieldnotes for Mobile Mapping has been developed for Windows based pen or laptop computers. Fieldnotes includes map-viewing functions including pan, zoom, coordinated look up, pop-up legends. The primary purpose of Fieldnotes is to update/modify existing digital map information.
Web Address	http://www.penmetrics.com
Company	Pocket Systems Ltd., UK
Available Since	PocketGIS for Newton OS released in 1997 and PocketGIS for Windows CE released in May 1999.
Technology	PocketGIS is a GIS for field data capture which can display and edit map geometry and attributes. PocketGIS runs on Microsoft Windows CE. PocketGIS can be linked to a Global Positioning System (GPS) receiver for real time location tracking and data capture, to a laser rangefinder, or to a Digital Camera.
Web Address	http://www.pocket.co.uk
Company	MGS-Mobile GIS Systems Oy, Helsinki, Finland
Available Since	?
Technology	SkyPower is a GPS based, multimedia GIS system especially suitable for video mapping applications in mobile environment. SkyPower runs on Windows based operating systems.
Web Address	http://www.mgs-mobile.com/index.htm
Company	CTN Data Service, Inc.
Available Since	1992

Technology	FarmSite is a companion tool to FarmWorks, a mostly farm accounting system. It's a home-brew Pocket GIS that relies on SHP formats that was created for their clients interested in getting started with farm mapping. It also integrates to FarmWorks crop records and field cost accounting modules. Farm Site geo-referenced map "layers", which typically include yield maps, soil type maps, soil test maps, and GPS maps.
Web Address	http://www.farmworks.com/products/farmsite/
Company	StarPal, Inc.
Available Since	?
Technology	StarPal's Handheld Geographic Information System (HGIS®) is a field tool that runs on PocketPC computers or ruggedized computers running Microsoft Windows (CE, 2000, 98, 95, ME, XP, or NT). Compatible formats include ESRI Shape SHP, MapInfo MIF, and dBASE DBF. The HGIS connects to a wide range of GPS systems for the determination of position. The HGIS can perform a number of basic GIS functions such as position location in 100 different coordinate systems, area and distance calculation and union/intersection/difference of features, for example, calculate the area of a field minus any ponds within the field.
Web Address	http://starpal.com/index.html

Appendix C – Hardware and Microphone Requirements

A speech application requires certain hardware on the user's computer in order to obtain adequate results. Speech recognition and text-to-speech engine Developer ScanSoft who develop and market Dragon Naturally Speaking recommend a minimum configuration of a Pentium® II 400MHz processor or equivalent, 128MB of RAM, at least 300MB of hard disk space; a sound card such as Creative® Labs Sound Blaster® 16 or the equivalent; Microsoft® Windows® XP, Millennium, 2000, 98, 95C, or Windows NT® 4.0 (with SP-6 or greater); and a microphone, preferably a close-talk microphone with a near-field element so that background noise can be eliminated (ScanSoft, 2002). IBM, the creators of ViaVoice 9.0, set out minimum processor specifications for both Microsoft® Windows® 98 and Microsoft® Windows® Millennium, being a Pentium® 300 MHz processor and a Pentium® III 600 MHz processor respectively. Other requirements include 64MB of RAM, 500+MB of hard drive space and a Windows® compatible 16-bit sound card (IBM, 2002).

It must, however, be remembered that these are minimum requirements specified by software developers. After reviewing some dedicated speech recognition web sites (emicrophones.com, 2002; out-loud.com, 2002; microspeech.com, 2002) and the USENET Group comp.speech.users it is evident that if good recognition results are to be obtained then a minimum configuration should at least include a Pentium® III 600 MHz processor, 384 MB of Ram, Microsoft® Windows® 2000 Professional, a disk swap file set to at least 300 MB and made permanent. There are numerous options with respect to sound cards, however, a sound card from the SoundBlaster Live product range, or USB sound pods from Andrea or Buddy/Emkay are recommended by many users; and lastly, an active noise cancelling microphone.

Microphones

A microphone is a transducer, a device that changes information from one form to another. Sound information exists as patterns of air pressure; the microphone changes this information into patterns of electric current.

Elsa (1996) describes two of the most commonly encountered microphone designs, being the magneto-dynamic design and the variable condenser design. In the magneto-dynamic (commonly referred to as the dynamic) microphone, sound waves cause movement of a thin metallic diaphragm and an attached coil of wire. A magnet produces a magnetic field which surrounds the coil, and motion of the coil within this field causes current to flow. It is important to remember that current is produced by the motion of the diaphragm, and that the amount of current is determined by the speed of that motion. This kind of microphone is known as velocity sensitive.

In a condenser microphone, the diaphragm is mounted close to, but not touching, a rigid back plate. A battery is connected to both the diaphragm and the back plate, which produces an electrical potential, or charge, between them. The amount of charge is determined by the voltage of the battery, the area of the diaphragm and the back plate, and the distance between the two. This distance changes as the diaphragm moves in response to changes in air pressure caused by sound waves. If the distance between the diaphragm and the back plate changes then current flows in the wire, as the battery maintains the correct charge. The amount of current is proportional to the displacement of the diaphragm, and is so small that it must often be electrically amplified before it leaves the microphone.

An important feature that affects sound quality is noise cancellation (emicrophones.com, 2002). With speech recognition, the user wants a microphone to screen out sounds from all directions except the user's voice. A number of screening techniques exist, the most common of which are Cardioid microphones, which only pick up sounds directly in front of the microphone; and Active Noise Cancellation microphones. The Cardioid microphone gets its name from the heart-shaped cross-section of the sensitivity pattern. The microphone is most sensitive to sounds that occur directly in front of it, and then the sensitivity is sharply reduced as the sound source moves around and behind the front end of the microphone. Sounds from directly behind the microphone are almost totally blocked (Elsa, 1996). The Active Noise Cancellation microphone relies on two or more microphones. In a two-microphone configuration, one is used to pick up the speaker's voice, and the other is used to gather the ambient noise in the environment. The ambient noise signals are then subtracted from the speaker's signal.

APPENDIX D - AMERICAN ENGLISH PHONEME REPRESENTATION

This is a brief introduction to the use and implementation of the SAPI phoneme representations[52].

SYMBOLIC AND NUMERICAL REPRESENTATION

Application developers can create pronunciations for words that are not currently in the lexicon by using the English phonemes represented in the following table. The phoneme set is composed of a symbolic phonetic representation (SYM).

The application developer will be able to enter the SYM representation to create the pronunciation using the XML PRON tag, or by creating a new lexicon entry. Each phoneme entry should be space delimited.

Tag	*Description*
PRON SYM	*Tag used to insert a pronunciation using symbolic representation.*

Example: pronunciation for “hello”:

```
<PRON SYM = "h eh l ow"/>
```

For improved accuracy, the primary (1), secondary (2) stress markers, and the syllabic markers (-) can be added to the pronunciation.

Example: pronunciation for “hello” using the primary stress (1) and syllabic (-) markers:

```
<PRON SYM = "h eh - l ow 1"/>
```

AMERICAN ENGLISH PHONEME TABLE

SYM	*Example*	*Phoneme ID*
-	*syllable boundary (hyphen)*	*1*
!	*Sentence terminator (exclamation mark)*	*2*
&	*word boundary*	*3*
,	*Sentence terminator (comma)*	*4*
.	*Sentence terminator (period)*	*5*
?	*Sentence terminator (question mark)*	*6*
_	*Silence (underscore)*	*7*

[52] Please note that this appendix has been sourced from Microsoft’s Speech Software Development Kit, Version 5.

SYM	*Example*	*Phoneme ID*
1	*Primary stress*	*8*
2	*Secondary stress*	*9*
aa	*father*	*10*
ae	*cat*	*11*
ah	*cut*	*12*
ao	*dog*	*13*
aw	*foul*	*14*
ax	*ag0*	*15*
ay	*bite*	*16*
b	*big*	*17*
ch	*chin*	*18*
d	*dig*	*19*
dh	*then*	*20*
eh	*pet*	*21*
er	*fur*	*22*
ey	*ate*	*23*
f	*fork*	*24*
g	*gut*	*25*
h	*help*	*26*
ih	*fill*	*27*
iy	*feel*	*28*
jh	*joy*	*29*
k	*cut*	*30*
l	*lid*	*31*
m	*mat*	*32*
n	*no*	*33*
ng	*sing*	*34*
ow	*go*	*35*
oy	*toy*	*36*
p	*put*	*37*
r	*red*	*38*
s	*sit*	*39*
sh	*she*	*40*
t	*talk*	*41*
th	*thin*	*42*
uh	*book*	*43*
uw	*too*	*44*
v	*vat*	*45*
w	*with*	*46*
y	*yard*	*47*
z	*zap*	*48*
zh	*pleasure*	*49*

APPENDIX E - STREET CONDITION SPEECH COMMANDS

GLOBAL COMMANDS

FILE MENU

Add layer – Opens a dialog to add a layer to the map view.

Working directory – Opens a dialog to set the current working directory.

Quit – Quits the application. Check to see if speech files should be saved first.

EDIT MENU

Find feature – Opens a dialog to find a feature.

VIEW MENU

Map properties – Opens a dialog to edit/set the map view properties.

Zoom all – Zooms to the full extent of the loaded data sets.

Zoom to layer – Zooms to the extent of the active layer.

Zoom – Activates the select zoom window function. Must use mouse to zoom in.

Zoom out – Activates the zoom out function. Must use mouse to zoom out.

Pan - Activates the pan function. Must use mouse to pan.

Identify feature – Opens a dialog that displays attributes of a selected feature.

LAYER MENU

Remove layer – Removes the active layer from the map view.

Remove all – Removes all layers in the map view.

Edit Legend – Opens a dialog to edit the layers in the map view.

DATA MENU

Open road database – Opens the database that stores road defects and displays existing defects on the map view.

Close database – Closes the road database

SPEECH SETUP MENU

Audio setup – Runs a microphone test routine to check environmental noise.

General training – Activates the general user training module.

Vocabulary editor – Opens a dialog for editing vocabulary.

Vocabulary builder – Opens a dialog for building vocabulary.

Train words – Opens a dialog to train individual words

New command – Opens a dialog for adding new commands.

Edit command – Opens a dialog for editing commands.

Voice options – Opens a dialog for setting voice options.

USER MENU

New user – Opens a dialog to create a new user's speech files.

Open user – Opens a dialog to select an existing user.

Save speech files – Saves changes to current user's speech files

WINDOW MENU

Tile horizontally – Tiles open windows horizontally.

Tile vertically – Tiles open windows vertically.

Cascade windows – Cascades open windows.

HELP MENU

About Mobile Street Mapper – Opens a dialog giving details about the Mobile Street Mapper

ACTIVE CONTROL GRAMMARS

GPS MENU

View GPS – Opens the GPS dialogue.

Hide GPS – Closes the GPS dialogue.

Open GPS File – Opens and runs a previously saved GPS file.

Start GPS – Starts acquiring GPS data.

Centre Map – Sets a toggle to ensure map/GPS location is always centred on the screen

Close GPS file – Stops processing a GPS file

DATA ENTRY MENU

Add defect – Adds a defect at the current GPS position

Modify road conditions – Modify/Edit defect at current GPS position (not implemented)

Quit data capture – Saves and closes the road database.

Close database – Same as “Quit data capture”.

Save data – Save spatial data.

Cancel – Cancels an existing process.

DATA FIELD GRAMMAR

DEFECT FIELD

Defect <Defect>

Defect type <Defect>

SEVERITY FIELD

Severity <Severity>

YES/NO FIELDS

Utility related <YesNo>

Seasonal <YesNo>

MAINTENANCE FIELD

Maintenance activity <Maintenance>

Maintenance type <Maintenance>

Maintenance <Maintenance>

DEFECT DIMENSION FIELDS

Width <Digit> point <Decimal>

Width <Digit> point <Decimal> metres

Width <Digit>

Width <Digit> metre

Width <Digit> metres

Length <Digit> point <Decimal>

Length <Digit> point <Decimal> metres

Length <Digit>

Length <Digit> metre

Length <Digit> metres

DATA LISTS

Digit List - Integers from 0 to 100

Decimal List – Integers from 0 to 9

DEFECTS LIST

- Distortion
- Rippling
- Raveling
- Random cracks
- Longitudinal cracks
- Wheel rutting
- Excessive patching
- Alligatoring
- Transverse cracks
- Severity List – Integers from 1 to 5

MAINTENANCE LIST

- Hot box
- Hand crew top
- Hand crew base
- Paver
- Crack sealing
- Manhole adjustment
- Other material

YES/NO LIST

- Yes
- No

Appendix F – Road Defect Schema

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.4 (http://www.xmlspy.com) by Andrew Hunter
(University of Calgary)
======================================================================
  File:   defects.xsd
  Author: Andrew Hunter
          Department of Geomatics Engineering
          University of Calgary
  Date:   August 17, 2001

  Revisions: May 27, 2002
          Updated to conform to the W3C XMLSchema Recommendation dated
          2 May, 2001. Fixed invalid type references.
======================================================================-->
<xsd:schema targetNamespace="http://www.ucalgary.ca/~ahunter/gml"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns="http://www.ucalgary.ca/~ahunter/gml"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:dft="http://www.ucalgary.ca/~ahunter/gml"
  elementFormDefault="qualified"
  version="2.1.1"
  xml:lang="en">

  <xsd:annotation>
    <xsd:appinfo>defects.xsd v2.1.1 2002-05</xsd:appinfo>
    <xsd:documentation>
      GML schema for road defect data. Copyright (c) 2001, 2002,
      Andrew Hunter All Rights Reserved.
    </xsd:documentation>
  </xsd:annotation>
<!-- import constructs from the GML Feature and Geometry schemas -->
  <xsd:import namespace="http://www.opengis.net/gml"
    schemaLocation="feature.xsd"/>
<!-- ====================================================================
     Global element declarations
===================================================================== -->
<xsd:element name="RoadDefectsModel" type="dft:RoadDefectsModelType"/>
  <xsd:annotation>
    <xsd:documentation>
      The RoadDefectModel contains all the features in the containment
      relationship called DefectMember.
    </xsd:documentation>
  </xsd:annotation>
<!-- a label for restricting membership in the Road Defect Collection -->
  <xsd:element name="_DefectFeature" type="gml:AbstractFeatureType"
    abstract="true" substitutionGroup="gml:_Feature"/>
  <xsd:element name="DefectMember" type="dft:DefectMemberType"
    substitutionGroup="gml:featureMember"/>
  <xsd:element name="Road" type="dft:RoadType"
    substitutionGroup="dft:_DefectFeature"/>
```

```
  <xsd:element name="Footpath" type="dft:FootpathType"
    substitutionGroup="dft:_DefectFeature"/>
<!-- ==========================================================================
     Type definitions for road defect model
========================================================================== -->
  <xsd:complexType name="RoadDefectsModelType">
    <xsd:complexContent>
      <xsd:extension base="gml:AbstractFeatureCollectionType">
        <xsd:sequence>
          <xsd:element name="created" type="xsd:dateTime"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>

  <xsd:complexType name="DefectMemberType">
    <xsd:annotation>
      <xsd:documentation>
        A DefectMember is restricted to those features (or feature
        collections) that are declared equivalent to
        dft:_DefectFeature.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexContent>
      <xsd:restriction base="gml:FeatureAssociationType">
        <xsd:sequence minOccurs="0">
          <xsd:element ref="dft:_DefectFeature"/>
          <xsd:element name="defectID" type="xsd:positiveInteger">
            <xsd:key name="dftKey">
              <xsd:selector xpath=".//defectID"/>
              <xsd:field xpath="defectID"/>
            </xsd:key>
          </xsd:element>
          <!-- date format: CCYY-MM-DD -->
          <xsd:element name="dateCreated" type="xsd:date"/>
          <!-- time format: hh:mm:ss.sss -->
          <xsd:element name="timeCreated" type="xsd:time"/>
          <xsd:element name="fieldOperator" type="xsd:string"/>
          <xsd:any minOccurs="0" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:restriction>
    </xsd:complexContent>
  </xsd:complexType>

  <xsd:complexType name="RoadType">
    <xsd:annotation>
      <xsd:documentation>
        A RoadType is a defect found on the surface of a road
        carriageway.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexContent>
      <xsd:extension base="gml:AbstractFeatureType">
        <xsd:sequence>
```

```
        <xsd:element name="defect" type="dft:DType"/>
        <xsd:element name="severity" type="dft:SType"/>
        <xsd:element name="maintenance" type="dft:MType"/>
        <xsd:element name="utility" type="xsd:boolean"/>
        <xsd:element name="seasonal" type="xsd:boolean"/>
        <xsd:element name="width">
          <xsd:complexType>
            <xsd:simpleContent>
              <xsd:extension base="dft:DistType">
                <xsd:attribute name="units" type="dft:UnitsType"
                  use="required"/>
              </xsd:extension>
            </xsd:simpleContent>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="length">
          <xsd:complexType>
            <xsd:simpleContent>
              <xsd:extension base="dft:DistType">
                <xsd:attribute name="units" type="dft:UnitsType"
                  use="required"/>
              </xsd:extension>
            </xsd:simpleContent>
          </xsd:complexType>
        </xsd:element>
        <xsd:element ref="gml:location"/>
        <xsd:any minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>

<xsd:complexType name="FootpathType">
  <xsd:annotation>
    <xsd:documentation>
      A FootpathType is a defect found on the surface of a footpath or
      along  the curb and channel of a carriageway.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexContent>
    <xsd:extension base="gml:AbstractFeatureType">
      <xsd:sequence>
        <xsd:element name="defect" type="dft:DType"/>
        <xsd:element name="severity" type="dft:SType"/>
        <xsd:element name="maintenance" type="dft:MType"/>
        <xsd:element name="utility" type="xsd:boolean"/>
        <xsd:element name="seasonal" type="xsd:boolean"/>
        <xsd:element name="width">
          <xsd:complexType>
            <xsd:simpleContent>
              <xsd:extension base="dft:DistType">
                <xsd:attribute name="units" type="dft:UnitsType"
                  use="required"/>
              </xsd:extension>
```

```
          </xsd:simpleContent>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="length">
        <xsd:complexType>
          <xsd:simpleContent>
            <xsd:extension base="dft:DistType">
              <xsd:attribute name="units" type="dft:UnitsType"
                use="required"/>
            </xsd:extension>
          </xsd:simpleContent>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="c_g_Length">
        <xsd:complexType>
          <xsd:simpleContent>
            <xsd:extension base="dft:DistType">
              <xsd:attribute name="units" type="dft:UnitsType"
                use="required"/>
            </xsd:extension>
          </xsd:simpleContent>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="replaceBlock" type="dft:BType"/>
      <xsd:element name="replace" type="dft:RType"/>
      <xsd:element name="boulevardBuildup" type="xsd:boolean"/>
      <xsd:element ref="gml:location"/>
      <xsd:any minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:extension>
 </xsd:complexContent>
</xsd:complexType>

<xsd:simpleType name="DType">
  <xsd:annotation>
    <xsd:documentation>
      Defects allowed on a carriageway, curb or footpath.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="Distortion"/>
    <xsd:enumeration value="Rippling"/>
    <xsd:enumeration value="Ravelling"/>
    <xsd:enumeration value="Random cracks"/>
    <xsd:enumeration value="Longitudinal cracks"/>
    <xsd:enumeration value="Wheel rutting"/>
    <xsd:enumeration value="Excessive patching"/>
    <xsd:enumeration value="Alligatoring"/>
    <xsd:enumeration value="Transverse cracks"/>
    <xsd:enumeration value="Sheet asphalt overlaid"/>
    <xsd:enumeration value="Tripping edge"/>
    <xsd:enumeration value="Catch basin displacement"/>
    <xsd:enumeration value="Cracks"/>
    <xsd:enumeration value="Crumbling"/>
```

```
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="SType">
  <xsd:annotation>
    <xsd:documentation>
      Defect severity, 1 being minimal and 5 excessive.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:positiveInteger">
    <xsd:maxInclusive value="5"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="MType">
  <xsd:annotation>
    <xsd:documentation>
      Allowable maintenance methods/techniques.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="Hot box"/>
    <xsd:enumeration value="Hand crew top"/>
    <xsd:enumeration value="Hand crew base"/>
    <xsd:enumeration value="Paver"/>
    <xsd:enumeration value="Crack sealing"/>
    <xsd:enumeration value="Manhole adjustment"/>
    <xsd:enumeration value="Separate sidewalk"/>
    <xsd:enumeration value="Mono"/>
    <xsd:enumeration value="Other material"/>
    <xsd:enumeration value="Curb and gutter"/>
    <xsd:enumeration value="Sheet asphalt"/>
    <xsd:enumeration value="Mud jacking"/>
    <xsd:enumeration value="Joint sealing"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="DistType">
  <xsd:annotation>
    <xsd:documentation>
      Decimal distances restricted to less than 100 units at a
      resolution of 0.1.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:decimal">
    <xsd:pattern value="[0-9]\.[0-9]|[0-9][0-9]\.[0-9]"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="UnitsType">
  <xsd:annotation>
    <xsd:documentation>
      Allowable distance units.
    </xsd:documentation>
```

```
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="feet"/>
      <xsd:enumeration value="meters"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="BType">
    <xsd:annotation>
      <xsd:documentation>
        Allowable block replacement values.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="Odd side"/>
      <xsd:enumeration value="Even side"/>
      <xsd:enumeration value="Median"/>
      <xsd:enumeration value="Road"/>
      <xsd:enumeration value="No"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="RType">
    <xsd:annotation>
      <xsd:documentation>
        Allowable replacement types; 1 = Separate sidewalk
                                     2 = Curb and Gutter
                                     3 = Both
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:positiveInteger">
      <xsd:maxInclusive value="3"/>
    </xsd:restriction>
  </xsd:simpleType>

</xsd:schema>
```

APPENDIX G – ROAD DEFECT INSTANCE


```xml
<?xml version="1.0" encoding="UTF-8"?>
<!-- File: gmlDefects_26_05_02_1523.xml -->

<RoadDefectsModel xmlns="http://www.ucalgary.ca/~ahunter/gml"
  xmlns:gml="http://www.opengis.net/gml"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.ucalgary.ca/~ahunter/gml defects.xsd">

  <gml:boundedBy>
    <gml:Box srsName="http://www.opengis.net/gml/srs/epsg.xml#26711">
      <gml:coord>
        <gml:X>700380.875875919</gml:X>
        <gml:Y>5662672.84569231</gml:Y>
      </gml:coord>
      <gml:coord>
        <gml:X>700385.673562891</gml:X>
        <gml:Y>5662673.59199227</gml:Y>
      </gml:coord>
    </gml:Box>
  </gml:boundedBy>

  <DefectMember>
    <Road>
      <defect>Distortion</defect>
      <severity>3</severity>
      <maintenance>Paver</maintenance>
      <utility>false</utility>
      <seasonal>false</seasonal>
      <width units="meters">0.0</width>
      <length units="meters">0.0</length>
      <gml:location>
        <gml:Point srsName=
            "http://www.opengis.net/gml/srs/epsg.xml#26711">
          <gml:coord>
            <gml:X>700380.875875919</gml:X>
            <gml:Y>5662673.59199227</gml:Y>
          </gml:coord>
        </gml:Point>
      </gml:location>
    </Road>
    <defectID>1385</defectID>
    <dateCreated>2002-05-26</dateCreated>
    <timeCreated>15:20:05</timeCreated>
    <fieldOperator>Andrew Hunter</fieldOperator>
  </DefectMember>

  <DefectMember>
    <Footpath>
      <defect>Tripping edge</defect>
      <severity>5</severity>
```

```
      <maintenance>Separate sidewalk</maintenance>
      <utility>false</utility>
      <seasonal>true</seasonal>
      <width units="meters">3.1</width>
      <length units="meters">0.0</length>
      <c_g_Length units="meters">0.0</c_g_Length>
      <replaceBlock>No</replaceBlock>
      <replace>1</replace>
      <boulevardBuildup>false</boulevardBuildup>
      <gml:location>
        <gml:Point srsName=
            "http://www.opengis.net/gml/srs/epsg.xml#26911">
          <gml:coord>
            <gml:X>700385.673562891</gml:X>
            <gml:Y>5662672.84569231</gml:Y>
          </gml:coord>
        </gml:Point>
      </gml:location>
    </Footpath>
    <defectID>1386</defectID>
    <dateCreated>2002-05-26</dateCreated>
    <timeCreated>15:23:47</timeCreated>
    <fieldOperator>Andrew Hunter</fieldOperator>
  </DefectMember>

  <created>2002-05-26T15:23:57</created>

</RoadDefectsModel>
```