EasyStreet©

A location management and data synchronization application for mobile computing

A Thesis

Presented to the

Faculty of the

School of Computer Science

Kennedy-Western University

In Partial Fulfillment

of the Requirements for the Degree of

Doctor of Philosophy in

Computer Science

September 2000

by

Steven J. Mastrianni

Unionville, Connecticut

Last updated 7/19/2000 4:38 PM

Abstract

In today's fast-paced business world, it has become extremely important to remain "connected" while on the road. Mobile workers such as salespeople, insurance agents and adjusters, truck drivers, and police officers routinely rely on mobile computing devices to perform their job. Another class of users such as physicians and visiting nurses could benefit greatly by the introduction of this technology.

Recent advances in hardware and software technology have spawned an entire new generation of mobile computing devices. While their predecessors provided relatively little processing power and few connectivity options, these new portable devices are now as powerful as some desktop computers and offer a wide range of connectivity features. In addition to basic dialup capabilities, they include an impressive array of options including infrared, radio frequency (RF) and satellite communications. The increased performance of these devices, coupled with the ability to connect to a network from virtually anywhere in the world has made them an acceptable platform for even the most demanding mobile applications.

Powered in part by this new class of devices, mobile computing is quickly becoming a way of life for an entirely new generation of workers we call "road warriors". Operating almost entirely from remote locations, these road warriors return to the main office only for an occasional meeting or to pick up materials,

while the rest of their time is spent making customer calls and visiting clients. In order to perform their job effectively, these workers require access to enterprise data including inventory status, product pricing, product availability, clinical information, and technical reports. They also need timely and reliable access to electronic mail and calendar information. However, getting access to this data from different places is not always easy. Differences in network configurations, protocols, and connection mediums can make accessing that data a daunting task. While most of these computers can maintain the network and connectivity configuration for a specific location or network, they do not have the ability to maintain a separate and distinct configuration for the different types of networks that might be encountered at different physical locations.

We will solve this problem with an application that will manage multiple, heterogeneous network configurations at any number of physical locations. The application will also give the user the ability to:

- Synchronize local and network files
- Replicate selected databases
- Send and receive email
- Cache selected Web pages

This application will be easy to set up and configure, and will make the task of managing complex network settings *easy*. We call this application EasyStreet.

Trademarks and Copyrights

Table of Contents

Figures

Tables

Chapter 1. - Introduction

The Problem

Technology and the pressures of the global marketplace have forever changed the way people work. Just a few years ago, much of the work performed was defined in the context of an 8-hour day or 40-hour week.

Today, we live in a world that never sleeps, and in an economy that that is so closely tied together that a small shift in one market is felt immediately in markets thousands of miles away. Technology has made it possible to provide instantaneous communications and financial transactions across the globe in a matter of a few seconds. e-Business and e-Commerce make it possible to buy anything from flowers to automobiles online without ever leaving home.

Faced with a shrinking pool of talented workers, high tech companies have adopted new ways to make workers more productive. One of the most popular initiatives has been telecommuting, or the ability to work from home. Still other companies have elected to save the high cost of building space by having employees operate from a “virtual office”, occasionally calling in for messages and appointments.

Each of these scenarios requires *access to data*. This data might be the company's latest price figures, inventory, or customer records, or perhaps the latest drop of source code. Users should be able to access their data quickly and easily no matter where they are located.

Connecting to various types of networks in different physical locations can be a real headache for mobile computer users. There are many types of networks currently in use, and although there are well-defined standards for these networks, network designers have been, for the most part, free to choose the type of network for a particular location. In some cases, the configuration was based on corporate policy, but more often it was based on the requirements of the software that ran on the network. Whatever the reason, the problem is that accessing these heterogeneous networks with the same mobile computer is a problem.

Some companies solve this problem by providing their own dialup access infrastructure, supplying their mobile workers with local or toll-free dialup access numbers. Once a dialup connection is established, the server authenticates the user, and if the proper security conditions are met, grants the user access to the company network. The server abstracts or "hides" the details of the network configuration from the user. This abstraction allows the user to access the network and network resources without a detailed understanding of how the

network is configured. If the network configuration is changed, the server resolves any differences and the user is unaware that anything has changed.

Other companies contract out or lease their dialup services from a large dialup provider like AT&T. In this case, mobile users dial local AT&T access numbers and are authenticated by the AT&T server. Once authenticated, the AT&T server connects the user to his or her company's internal network. (This configuration requires that the network access provider take adequate steps to insure that no unauthorized users are allowed to connect to their network. They must also make sure that the user's data transfer is secured and that confidential data is protected during transmission and while stored at the provider's site.) Once connected to the server, the server abstracts the network configuration the same way that the private server does.

In both of these dialup access situations, users don't need to know the details of the underlying network configuration to connect to the network. Once the user is connected, they can use their company-authorized services without worrying about such things as gateways, name servers, and Internet Protocol (IP) addresses. The dialup provider's server automatically routes the network traffic to the proper destination. If the user attempts to access resources on the Intranet, the server routes the traffic locally. If the user attempts to access data outside the Intranet, the server routes the data automatically through the provider's firewall.

Data returned from the Internet is automatically routed to the requesting user through the dialup server.

While both of these methods work well for dialup users, connecting to a fixed, wired network presents a much more challenging problem. While dialup providers usually rely upon a standard point-to-point protocol (PPP) connection using Transport Control Protocol/Internet Protocol (TCP/IP), wired networks can use Token Ring, Ethernet, and wireless Local Area Networks (LAN) adapters, and support TCPIP, NetBIOS, and NetWare protocols. Some networks use Dynamic Host Configuration Protocol (DHCP) to assign IP addresses to users, while other networks require that each user have their own fixed IP address. The network may provide default name lookup services or require users to indicate the fixed IP address or addresses of the network's Domain Name Server (DNS). Some Windows NT networks also require that the user indicate the fixed IP address of the Windows Internet Naming Services (WINS) server.

Current operating system implementations provide only one set of the parameters required to connect to a wired network. If a user attempts to connect to a network in a different physical location, they would have to change some or all of these network parameters to be able to successfully access the network or network resources. Changing these parameters is not easy, even for experienced users. Worse, changing the wrong parameter or using the wrong

parameter value could prevent the system from connecting to the network and in severe cases, could even render the system unbootable.

The increased dependency on computing services has led many companies to form Information Technology (IT) or Information Services (IS) organizations. In larger companies, the IT 'shops', as they are called, control access to the company network by specifying the network software and hardware components that comprise the network, and providing network access verification through the use of IDs, passwords, and accounts. Many of these companies are scattered throughout the world and have offices in remote locations, while others have offices or divisions in major metropolitan areas.

In the past, the network administrator or IT shop dictated the configuration of the local network at a particular location and it was not uncommon to find that the systems in one division of a company were not compatible with the systems in another division of the same company. Only recently have government agencies been able to share information from their local databases. For example, many local, state, and federal governments now routinely crosscheck their list of welfare recipients against a list of felons and prisoners. Once of the reasons for this was that although the networks and network software were part of the same organization, the network infrastructure was usually designed on a local basis without worrying about sharing data or software with another location. These network "islands" have only recently begun to standardize on network topology

and applications so they can share their information with other parts of their organization and to take advantage of information that exists in the other locations.

Over the past decade, there has been a movement to standardize on the TCPIP protocol, although a substantial number of Novell NetWare installations still exist. The NetBIOS protocol is still popular on Windows NT networks, and will not be going away anytime soon. Most of these networks use IBM Token Ring or industry-standard Ethernet network interface cards (NICs), although it appears that most companies will standardize on Ethernet within the next five years. However, Token Ring won't be going away anytime soon. Many of IBM's large customers had previously taken their lead and installed Token Ring adapters throughout their organization, so it may not be uncommon to encounter both Token Ring and Ethernet networks in the same location.

To add to the network configuration dilemma, the increased popularity and availability of wireless networking has introduced an entire new set of network interfaces that support infrared, cellular, RF, and satellite communications. These devices allow users to connect to a number of network topologies including Wide Area Networks (WANs), LANs, and Personal Area Networks (PANs). Many road warriors have several of these devices plugged in at the same time so as to have the option to connection via an alternate method should the desired type of

connection not be available. Mobile users will find it impossible to remember all of the settings for these networks and adapters.

Another problem is that mobile connections are unreliable. Once connected, the mobile user has no idea how long the connection will last. Disconnects over the Public Switched Telephone Network, or PSTN, are commonplace, while cellular and radio frequency (RF) connections suffer from interference and reception problems. It is important, therefore, that once connected, the user's data be retrieved as quickly and efficiently as possible and that the system be capable of resuming the data transfer if the connection is lost.

Once the data has been transferred, the user should be able to work as if they still had a persistent connection to the Wide Area Network (WAN) or Local Area Network (LAN). Requests to print a document, for example, should be carried out as if the printer was connected to the mobile computer. The user should be able to view Web pages, complete with audio and video, and should be able to click on other Uniform Resource Locators (URLs) in the Web page and see their content also.

To solve these problems, we present a mobile location management and data transfer application that we call EasyStreet. EasyStreet keeps track of any number of networks, protocols, and connectivity options across multiple physical locations and networks. It provides an intuitive, easy-to-use interface to

connection and network parameters, user IDs, passwords, and user preferences. EasyStreet makes the task of accessing data across multiple heterogeneous networks easy and effortless by maintaining detailed configuration settings on a per-location basis. It provides automatic, unattended retrieval of mail, Web pages, databases, and files, maintaining configuration information such as user logon IDs and passwords. This allows users to spend more time using their data and less time trying to get to it.

Purpose of Study

The purpose of this study is to implement the location management application to evaluate its usefulness and applicability to the problem. The nature of this application is such that we do not feel that we can judge its usefulness or usability by using mock-ups or facades, and that only by actually designing and building this application will we be able to judge its effectiveness.

Importance of Study

This study will help determine if such an application will be deployed as part of a set of software that is preloaded on all new mobile computing systems. The software will be used to help differentiate the mobile computer system from other vendors' products by providing a level of software differentiation, ease of use, and positive customer experience.

Scope of Study

The study will be performed with a group of approximately 50 end-users. The group will perform alpha and beta tests of the software under actual conditions and will note their experiences on a Web-based discussion database. At the end of the test period, the data will be analyzed to determine if the test was successful.

Rationale of Study

There are several ways to solve this problem. Two other vendors have competitive products that provide a majority of the functions necessary to provide the location management features. We could license either one of those products and include them with the computer system. The vendors of these products are well recognized in the industry and a partnership could add value to the computer system. However, none of the competitor's products provide the level of features available with EasyStreet. In the future, we could add more features to our own software without asking the vendors to do it, or letting them in on any of our research. Any intellectual property could be added without consulting with the vendor. This study will help us to understand which method makes the most sense for the product.

Definition of Terms

Road Warrior

In this document, we often use the term "road warrior" to describe the potential benefactor of our technology and applications. This is an important distinction as we view this class of mobile user as having different needs than the casual traveler. The casual traveler might leave the office every two or three weeks on

a two-day business trip to see a customer or vendor. They are generally never away from the office for an extended period of time, and they usually have someone who knows where they are and who knows how to contact them if something important arises. Road warriors, in contrast, spend most of their time on the road, and might be in the office one day a week for a meeting or to pick up materials.

Mobile and Nomadic Computing

As [Hela99] and others have noted, mobile users have different requirements than nomadic users. Travelers that take their notebook along on a business trip and use their hotel phone to retrieve their email are not mobile users, but *nomadic* users. Like members of a nomadic tribe, these users travel from place to place and set up shop wherever they land. Travelers carrying laptops with dialup modems, are, therefore nomadic users engaged in nomadic computing [Hela99].

Mobile users, however, are always on the road and rely on connectivity from a variety of sources. They almost never connect from the same physical location, and often require persistent connections while moving. This group of mobile users includes delivery services, disaster aid workers, insurance adjusters, law enforcement, and similar occupations.

Registry

In this document, we refer to the term Registry to describe the persistent storage area for parameters and configuration information used by the EasyStreet application. In the Windows environment, the Registry is the actual name of the persistent database used by the Windows operating system and Windows applications. In the context of this document, when we refer to the term Registry, we mean this to include any persistent form of database storage accessible by both operating system and applications.

Overview of Study

This study consists of the software development of an application to make mobile connectivity easier. It involves the design and development of a location management and data synchronization application for the Windows™ operating system. Although this application was developed on Windows, most of the concepts and design points are applicable to other operating systems including OS/2 and Linux.

What is Mobile Computing

When we think of a personal computer, we normally think of our office, the monitor placed on the desk with a mouse nearby, and the familiar whirring of the cooling fan in a large box-like computer enclosure. This is the domain where many of us do our work. It is the primary venue for programmers, scientists, graphic designers, librarians, secretaries, telephone operators, and hundreds of other types of workers. In most cases, it would be impossible to perform the work associated with these occupations without the use of a personal computer. The power of these high-end Pentium-class machines has enabled mobile users to perform the type of work that was once relegated to mainframes and supercomputers. Performance has continued to increase at a pace that will likely render Moore's Law[1] obsolete within five years.

Yet many jobs just can't be done at a desk. It would be impossible, for example, for an insurance claims adjuster to survey the damage caused by a flood or fire from his desk, or to view the damage to an automobile following an accident. Taking statements at the scene of a crime, recording and analyzing evidence or reconstructing an accident scene can't be done from a desk while tethered to a personal computer. In these cases, much of the information is gathered and then

[1] Moore's Law, coined by Gordon Moore in 1965, states that the capacity of a microchip can doubles every 18 months.

assembled and analyzed later, often with the help of a computer. Insurance agents need to spend their time visiting clients and perspective customers. Any time spent in the office is valuable time that could have been spent generating new business and it can't be recouped. An insurance adjuster surveying the damage from a tornado or flood needs to get the information collected quickly and sent back to the office so the displaced residents can get back in their homes as soon as possible. And of course, delivery services must maintain an accurate schedule and location of their cargo to be able to respond to inquiries from shippers and receivers.

As recently as two years ago, most mobile computers were simply used for data collection because they lacked the power to perform complex calculations. Data was usually collected on the device and later analyzed by another program running on a desktop machine or server. Although primarily used to capture data, these mobile devices had few storage options available for storing large amounts of data. Even if the device had an adequate amount of storage, it would not run for an extended period of time without a recharge.

In spite of these drawbacks, many companies have deployed these devices as part of their business. Forward-thinking companies like Federal Express, Airborne, and United Parcel Service have been using wireless cellular technologies to improve their delivery service and to keep accurate information on the location of packages. A few large insurance companies have issued

mobile devices to their agents, while several home health service providers have deployed them to their home care staff. Many companies, however, have been reluctant to embrace mobile computing as an integral part of their business strategy. One of the obvious reasons for this is that mobile computers have been too expensive to justify, and until just recently, the cost of wireless communications has been too prohibitive.

Today we appear to be at the beginning of what could be the largest deployment of mobile computing devices in history, the results of which could have a profound and lasting effect on the way business is transacted “on the road”. There are three major factors that have caused this to happen, and they have all converged to allow this radical change to take place.

The first and most obvious change is the enormous popularity of the Internet and the effect it has had on business, industry, government, education, and science. The Internet has now become “the place” to find or buy anything.

Second, mobile devices have finally become powerful enough to handle even the most demanding computing applications.

Third, the increased popularity of wireless communications and a large number of subscribers have resulted in lower costs for wireless connections due to economies of scale. Also, many wireless carriers, buoyed by the influx of cash

from mergers are offering deep discounts and promotional deals to gain market share.

It should be noted here that along with many other experts on this subject, we recognize that there is a distinct difference between *mobile* computing and *nomadic* computing. While mobile computing involves connecting while moving, nomadic computing involves connecting while stationary. Nomadic users travel from point to point carrying their mobile computer, and then connect when the circumstances permit. This might be at a hotel, airport lounge, or company office, but generally requires the user to place the computer on a table or fixture of some type before attempting to connect. In certain circumstances, nomadic users can also become mobile users. An example of this would be connecting to a network using a cellular phone while flying or driving.

It is our opinion that traveling users tend to be nomadic or mobile, but not both, and that the type of applications used determines which type they are.

Mobile Computing Devices

In the past, mobile computing devices were usually assumed to be notebook computers. This is primarily because notebook computers were the only types of mobile computing device that had sufficient processing power to run the user's desktop applications while on the road. Connectivity was generally limited to a

dial-up modem installed internally or through the addition of a plug-in PCMCIA card.

The primary method of connecting to a network was through a dial-up connection over a PSTN line or very slow cellular link. Both of these methods involved dialing an access number supplied by the access provider. The obvious problem with a dial-up connection is that the user had to be tethered to a phone of some sort. For PSTN, it meant being in a stationary location, such as a phone booth at the airport or the phone in a hotel room. For a cellular connection, it meant being in an area of good reception and usually stationary. Cell coverage was fairly poor, and connections were prone to frequent disconnects as the user traveled through hilly terrain.

Today's mobile computing devices are smaller and lighter, and more importantly much more powerful than their predecessors are. Advances in battery technology and power management software allow today's mobile devices to operate for a much longer period of time before recharging. This is in spite of the fact that the number of peripheral devices installed on these systems has increased dramatically.

While previous devices relied on PSTN or cellular dialup connections, a new class of mobile devices has emerged with many different methods for connecting to a network. These include internal infrared transceivers, wireless two-way

radios, wireless pagers, and satellite receivers and transmitters. We are beginning to see an emphasis on high-speed connectivity via these new wireless mediums, with some future implementations promising up to 1.5 megabits per second on wireless connections. Such technologies will enable the use of high-speed, real time video and audio streaming with mobile devices, allowing users to watch high-resolution video, listen to music, or collaborate with others in real time.

In spite of these new features, these mobile computers tend to be slimmed-down or miniaturized versions of their predecessors. Although they provide more utility at a greatly reduced cost, they still remain somewhat difficult to setup, configure and operate. There are several efforts underway to change the look and feel of these devices, and to change the way we interact with them. The idea is to make these devices information “appliances”, capable of interacting with users and other devices through a variety of input and output modalities including touch, natural language, eye movement, and even thought processes. Some of these efforts are discussed in the next chapter.

Notebook Computers

Today’s notebook computers are smaller and lighter than ever. Most high-end notebook systems are less than one inch think, yet provide the level of processing power that was limited to desktop machines just a year ago. Full size

notebook computers still provide two important features that continue to make them desirable, a large display and a big keyboard. With these features, the notebook computer can double as a mobile device and stationary workstation. These systems offer large amounts of storage using traditional disk drives, while some offer high quality video, surround sound and built-in DVD drives.

Many of these high-end systems are now offering attachments to make them even more useful, such as digital cameras, wireless radios, infrared transceivers, and wireless network cards. While still the mobile device of choice, they may become less and less attractive as different input and output modalities appear on the smaller devices.

Sub-notebooks

Sub-notebooks, the notebook computer's smaller cousins, have also become quite popular. Although they lack the large display and keyboard of the full-size notebooks, they are much easier to carry and offer good performance. It is sometimes difficult to tell if a computer is a sub-notebook or what we often refer to as a handheld device. Some clamshell style computers such as the NEC 850 have a full QWERTY keyboard and reduced size display, and can be supplied with a built-in modem and networking card. Such features make this class of device desirable for mobile use. The NEC 850, for example, does not use a

spinning disk for storage, but instead uses a compact flash card, which provides a much more power-efficient method of albeit reduced storage.

Handhelds and Palm-style Devices

This class of device is becoming more popular due in large to the popularity of 3COM's PalmPilot. These devices come in a wide variety of configurations, and run one of three handheld operating systems: PalmOS, Windows CE, or EPOC. The PalmOS operating system is used on the 3COM PalmPilot and IBM WorkPad, while Windows CE is used on a large number of devices offered by Microsoft's Windows CE partners. The EPOC operating system is the foundation of the Psion series of mobile computers and the result of a joint software effort between Ericcson, Nokia, and Psion.

The devices are small and light, and have concentrated more on conserving battery life than on connectivity. Most provided only analog dial-up capability using some type of snap-on dongle that contained a modem. Recently, however, several of these devices have begun to include infrared transceivers and will soon begin to ship with wireless radios and network connections. Using the infrared and wireless communications features, these devices are able to replicate data such as address books, calendars, and email with desktop machine or even dial-up servers.

Cellular Phones

Once the status symbol of only affluent businessmen and techno-geeks, cellular phone are now standard issue for everyone from elementary school students to senior citizens on the golf course. Economies of scale have provided users with low-cost connections and enhanced coverage even in remote areas. Recently, the Iridium phone was introduced (for those willing to pay the $17 per minute charge). The Iridium phone communicates via satellite, providing coverage from anywhere in the world. In one demonstration of the Iridium phone, a user at the North Pole called a reporter in the United States and had a conversation that was almost as clear as if the person were next door on a conventional phone.

New cellular phones with built-in web browsers and email clients are beginning to appear. In the next few months, many of these phones will include the new Bluetooth wireless peer-to-peer networking system. This wireless technology will allow these cellular phones to share data with other cellular phones, pagers, notebooks, and desktop computers. Although Bluetooth operates over a limited distance, other types of wireless communications such as the new 2.4GHz 802.11 networks will increase the range of these devices. In the next few years, we should see phones that incorporate new third-generation 3G wireless technologies that will allow phones to be used with streaming audio and video. For more information on Bluetooth, the reader is directed to the Bluetooth web site at www.bluetooth.com.

Smart Pagers

Pagers are no longer just the simple devices that beep or vibrate when a call is received. Recently introduced pagers have built-in keyboards, email software, end even the capability of checking stock quotes via the Internet. They contain powerful, real time operating systems capable of performing simultaneous operations and even providing some level of computation. They are able to screen calls, categorize them, and store a significant number of messages, in some cases with attachments.

These devices are beginning to appear with the same types of communications hardware and software that are found in the new smart phones, notebooks, and desktop computers. Pagers with these capabilities are able to share data with other pagers, phones, or computers. Users can replicate mail, calendars, to-do lists and reminders with their desktop system, eliminating the need to enter data manually into the pager.

Mobile Connectivity

Mobile users require access to data, and gain access to that data by connecting to a network using a variety of communications mediums including analog dialup, cellular, satellite, infrared, and radio frequency (RF). However, there is also a large group of mobile users that connect to a fixed network using a wired connection. An example would be the employee of a large corporation that travels to that company's various office locations and connects to a wired LAN at each location. While simple dialup access may be acceptable while traveling, there are significant advantages to connecting directly to a wired LAN. One obvious advantage is the higher speed available on the LAN. Another advantage is the ability to use network resources, such as shared network disks and printers. Each of these connectivity methods is covered in more detail in a variety of publications. They are discussed here only in the context of the location management and data access features of EasyStreet.

The standard protocol used to connect to most wired and wireless networks is TCP/IP. TCP/IP utilizes a 32-bit numbering scheme called IP addressing to identify and communicate with devices that are connected to the network. Each device is assigned, either statically or dynamically, a 32-bit IP address that is used to uniquely identify that particular device on the network. When others need to communicate with that device, they specify the IP address of that device as

part of the communications protocol and then broadcast that data on the network. Each computer or device that is listening for network messages “sees” the data and checks the see if the IP address in the data matches its own. If not, the packet is discarded and the device begins listening again. If the IP address in the data matches the current system, the data is then passed up the driver stack to be processed.

There are several excellent texts such as [Tann96a] that describe TCP/IP and the network protocols that are supported by the TCP/IP stack. While it is not the purpose of this paper to explain TCP/IP or other network protocols in detail, we feel that a review of the basic principles of TCP/IP is necessary for an understanding of the location management features of EasyStreet.

Internet Protocol Basics

Network communications between computers are based on a layered architecture of software that allows the computers to access the network without having to know the details of the underlying network structure and hardware configuration. Easy software layer communicates with the layer above or below it with a predetermined set of communications called a protocol. Each protocol layer is responsible for translating messages to and from the layer above it and the layer below it. Together, the layers are referred to as the **protocol stack** (see Figure 1-1 – The TCP/IP protocol stack).

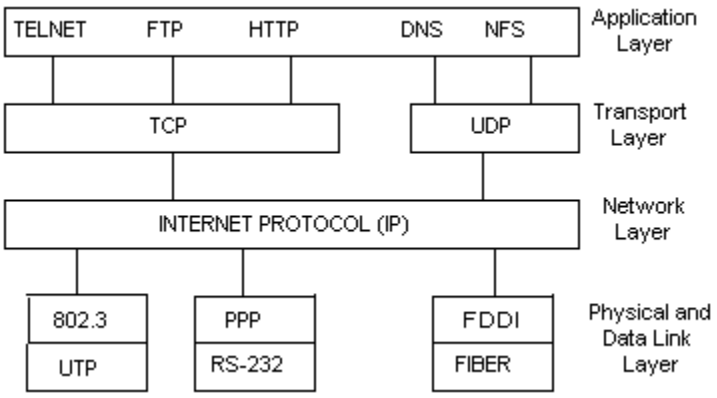

Figure 1-1. The TCP/IP protocol stack.

The physical layer handles the transmission of bits over the network. It is responsible for sending and receiving raw data over the network. At the physical layer level, the most common protocols are Ethernet and Token Ring. A significant number of older DOS and Windows-based machines may still be using ARCnet and Novell NetWare. The physical layer is actually composed of two layers, the physical and data link layers.

The next layer, the internet layer, is responsible for delivering packets where they are supposed to go. The packets can travel independently of each other and arrive at the destination at different times. The receiving system must assemble the packets together in the correct order. The idea behind this protocol is that

each packet can be rerouted on the way to its destination to avoid congestion or broken connections.

The third layer is called the transport layer, and is responsible for making sure that the packets get to the destination without errors. Two protocols are used at this layer. The first is called the Transport Control Protocol, or TCP. TCP regulates the flow of data so that a fast transmitter cannot overwhelm a slower receiver, and is also responsible for reassembling the received packets into the correct order. The second protocol used at this level is called User Datagram Protocol, or UDP. This protocol is used for sending packets without sequencing or flow control, and is typically used for broadcast-type messages where no reply is required.

The top layer is called the Application Layer. This is where applications can send and receive data using a higher-level protocol. Some of the protocols that you would find using this layer are Telnet, FTP, SMTP, NNTP, HTTP, and DNS.

Most mobile connections and networks are based on TCP/IP. This means that each device on the network has a fixed or automatically assigned IP address so that other computers (and users) on the network know how to contact it. The network's DNS server performs the mapping of an IP address to the name of the computer system. In order for a computer on the network to connect to another computer, it has to know the other computer's IP address or network name. If it knows the other computer's IP address, it can contact the other computer by

issuing a request directly to the other computer's IP address. The most common method, however, is to use the other computer's name which is then translated by the network's DNS server into an IP address.

IP addresses can be assigned statically by the network administrator or assigned automatically by a server on the network. If the mobile computer has a static IP address, it must also be configured with the IP address of the network's DNS server. When the mobile computer needs to connect to another computer on the network, it sends the name of the other computer to the network's DNS server, which translates the name into an IP address. The network may also require the mobile computer to know the address of the network's WINS server. The WINS server translates requests by name to the proper IP address.

No two machines can have the same IP address or a conflict will occur. All IP addresses are 32 bits long and are organized by classes (see Figure 1-2 – IP address formats). They are written in dotted decimal notation, where each of the four bytes that comprise the IP address are written as numbers from 0-255 and are separated by a dot.

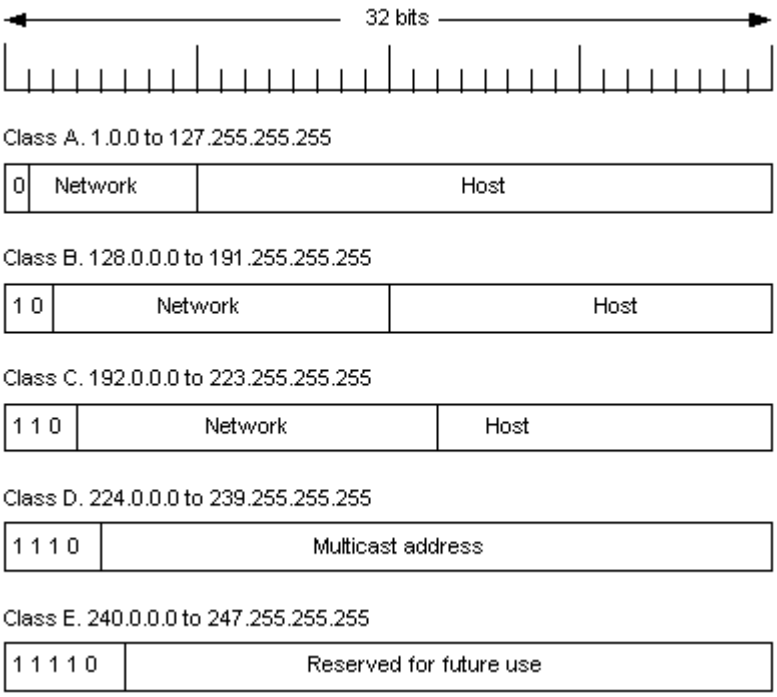

Figure 1-2. IP address formats.

If the network supports Dynamic Host Configuration Protocol (DHCP), a DHCP server on the network assigns an IP address automatically to every computer that connects to that network. At the same time the DHCP server assigns the IP address, it also sends the connecting computer the address of the network's DNS server. This allows the connecting computer to access the network's name server without having to know the name server's IP address in advance. If the IP address of the name server changes, the DHCP server simply sends the DNS server's new IP address the next time a connection is requested.

Certain IP addresses are referred to as special IP addresses and provide special functions (see Figure 1-3 – Special IP addresses).

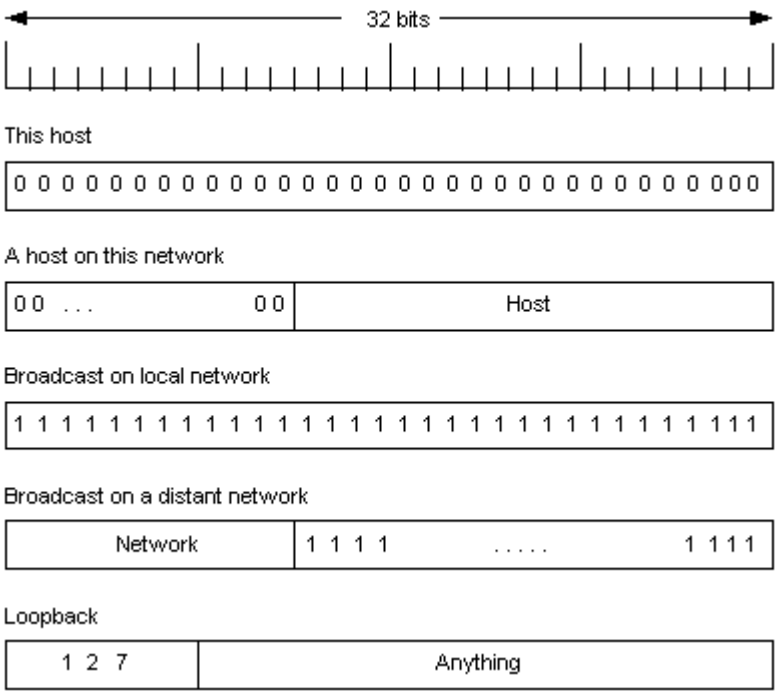

Figure 1-3. Special IP addresses.

Access Points

In general, the mobile computer connects to the network through an **access point**. Depending on the type of network, the access point can be close to the mobile system or a great distance away. In the case of dialup over a Public Switched Telephone Network (PSTN) line, the access point is a modem connected to a dialup server. In the case of infrared or a wireless LAN, the access point is an infrared transceiver located in close proximity to the mobile computer. In both cases, the server also acts as a gateway to the internal network.

The ability to connect (and stay connected) to the network is important for efficient use of the connect time, so the location of the access points can be critical. With a cellular connection, for example, it is important to have correct placement of the cells so as to provide a seamless handoff from cell to cell as the user passes through the network. In the case of a short-range wireless network such as Bluetooth, the location of the access points in a building is important to insure that the connection is maintained as the user walks from room to room.

Dialup

By far, the most popular and widespread method of connecting to a network is the standard PSTN analog dialup connection. Dialup connections require a modem to be installed in or connected to the mobile computer. The analog dialup connection is the most inexpensive method of connecting to a WAN such as the Internet because it utilizes the existing telephone infrastructure. Analog dialup has the obvious drawback of requiring the mobile computer to be in close proximity to a phone or phone jack, a requirement that makes the user less mobile.

Dialup users need not be concerned about the physical location of the network nor its physical configuration. The dialup access provider abstracts that information for the dialup user and provides the user with access to their resources in a transparent fashion. This can be done using a variety of methods,

but is most often accomplished with the provider's gateway and domain name server (DNS). When the user asks to display a Web page, for example, the provider's DNS server looks up the Web site name, converts the name to an IP address, and issues the request. The same lookup is performed for access to email.

Analog Cellular

Many mobile users connect to their network using some form of cellular communications. Although the cost of cellular has dropped significantly over the last decade, cellular connections are still expensive and not practical for many mobile users. Cellular systems suffer from problems of intermittent connectivity, dropped calls and relatively low bandwidth, along with a lack of standards for cellular communications. The low bandwidth is a major problem because limits the usefulness of analog connections.

In the United States, the Analog Mobile Phone System, or AMPS, is the standard cellular system in use. The AMPS system automatically transfers active phone connections from cell to cell to provide seamless communications while moving. The cells are located at variable distances based on terrain, and are generally located at distances of two to ten miles apart. Although the cellular providers have made cellular connections more reliable, the system is still less than optimal. Transmission rates are usually no more than 4800 baud, with some even lower, and modems sometimes lose synchronization when the call is

switched to a new cell. There are currently over eight million subscribers to AMPS in the United States.

Providers have introduced a packet protocol that is sent on a voice line when it is inactive. When data is sent, the cellular carrier's system looks for an open channel and sends the data on that channel. Voice calls take precedence, so if another voice call needs the channel, the cellular carrier's system gives it up. This protocol, called Cellular Digital Packet Data or CDPD, utilizes the existing cellular infrastructure and uses IP to communicate with devices.

Digital Cellular

In the United States, the most popular technologies for digital cellular communications are Time-Division Multiple Access (TDMA) and Code-Division Multiple Access (CDMA).

Unlike the traditional AMPS system that allocates a radio frequency channel for each user, TDMA divides each channel into six time slots with two slots for each signal. This increases the effective throughput by three times over AMPS. It also has a side effect of wasting bandwidth if all of the slots are not utilized. Hughes Systems Network implemented an improvement on the slot allocation called Enhanced TDMA, or ETDMA. This method allocates the slots dynamically based on traffic, resulting in a much more efficient use of bandwidth.

CDMA allows many systems to use the same channel concurrently using a technique called spread spectrum technology. With spread spectrum technology, the signal is spread across the entire frequency range of the channel. A special digital code is allocated for each user, preventing interference between signals on the same channel.

Another digital cellular service in the United States is Personal Communications System, or PCS. PCS is available in limited locations, primarily in densely populated metropolitan areas. PCS provides users with a subset of the features that are normally available on notebook computers, such as the ability to check email messages, browse the Web, and check the price of their favorite stock. A PCS device is normally assigned a static IP address so that the PCS carrier can communicate directly with the device. PCS users can also take advantage of some extra features such as email forwarding and voice mailboxes.

Local Area Networks

While we often think of the mobile user as someone that dials into the network from a remote location, there is a large class of users who travel to another location and connect to a LAN at that new location. These users have been classified as *nomadic* because they travel from place to place and connect at the new location in a stationary fashion. This differs from the true mobile user who

generally connects while moving, and is an important distinction that has been noted by [Perk97, Saty93, Zasl95].

Wireless LANs

Wireless LANs have become a popular method of connectivity for mobile and nomadic users. There are several technologies that provide wireless network connections, and some in the specification stage that promise to bring speeds of one megabit per second and higher within the next five years. A new 3rd generation protocol called Wideband Code Division Multiple Access, or WCDMA promises speeds of two megabits per second in a stationary office configuration and up to 384K bits per second in a mobile or wide area environment. Once deployed, this third-generation (3G) technology will enable mobile devices to be used for applications that are currently impractical, such as the ability to view or capture high-definition video and audio.

The Bluetooth technology promises seamless interoperability between different devices that have implemented the Bluetooth standard. This relatively low-cost radio frequency technology has a limited range of approximately ten meters. The key to the success of the Bluetooth technology will be the deployment of a standard applications interface that will allow Bluetooth-enabled applications to run on various devices.

The 802.11 and OpenAir interface standards currently provide up to 2 megabits per second in the 2.4-gigahertz band. The range of these technologies is approximately 200 meters, much larger than Bluetooth. While 802.11 supports IP, it does not support multicast, IPV6, or the RSVP protocol that may limit its usefulness in the future.

The cost of 802.11 is also much higher than Bluetooth, and requires properly positioned access points, especially inside buildings of metallic construction. When the access points are positioned correctly, devices that utilize this technology can be configured dynamically to “extend” the connection to a wired LAN. Once connected, the device acts as a proxy or bridge for other devices that require network access. Other devices can then connect to the wired network using the first mobile device as a bridge to the network.

Infrared

Infrared communications emerged in the mid 1990’s as a method for connecting to other devices and access points over a limited distance. Standardized and promoted by an industry group called the Infrared Data Association, or IrDA, infrared uses light that has a slightly longer wavelength than visible light, approximately 900 nanometers [Perk97]. It is used primarily at distances of 15 meters or less, and provides speeds as high as four megabits per second.

Early infrared transceivers required a line-of-site connection and could not be offset from the other device by more than 30 degrees without risk of losing the connection. Newer versions of infrared transceivers support a non-directional form of infrared communications called *diffuse IR* at rates of up to four megabits per second. Diffuse IR does not require a line-of-sight connection.

IR enjoyed some early success by providing wireless connectivity to printers. Printer manufacturers such as Hewlett Packard joined the IrDA and were instrumental in shaping the standards for infrared communications. Not surprisingly, they were also the first printer manufacturers to offer IR connectivity in their low-cost laser printers.

Recently, IR has surfaced again as a means of wireless peer to peer communications on the 3COM Palm Pilot.

Satellite

Satellite communications are made possible by the many satellites that are orbiting the Earth. These satellites can be geostationary (GEO) or low-earth orbiting (LEO). GEO satellites orbit at 36,000 kilometers above the earth and boast a very large field of view (FOV) of approximately 13,000 kilometers. LEO satellites orbit at a much lower altitude, approximately 16,000 kilometers above the earth, and provide a much smaller FOV of approximately 6,000 kilometers.

Both types of satellites are used for mobile communications, although none are deployed in large numbers. Even though a large number of satellites have been placed in orbit, satellite communications are still slow and too expensive for normal use.

Because of the smaller FOV, LEO satellites hand off communications much more frequently resulting in frequent interruptions in the data flow. GEO satellites don't need to hand off communications as often, but the satellite's altitude introduces delays in sending and receiving data. These delays can be as long as 250-500 milliseconds, and can affect quality and performance when viewing or listening to video and audio streams.

In November of 1998, Iridium LLC became the world's first company to offer global satellite communications and paging services. Iridium utilizes a network of 66 LEO satellites combined with existing terrestrial cellular systems enabling users to communicate from any location on earth no matter how remote. Iridium is capable of sending and receiving data at up to 2400 bps. At over $17.50 per minute, however, the system remained too expensive for most users. On August 13, 1999, Iridium filed for Chapter 11 bankruptcy and began to seek a buyer. After failing to secure the necessary capital, Iridium ceased operations on March 21, 2000.

Chapter 2. - Review of Related Literature

Importance of Location Management

Location management plays an important role for the mobile and nomadic user. Mobile and nomadic users normally access their data from a foreign location. By foreign, we mean a location that does not contain the same physical connectivity features or network resources normally found at the user's home office. There might be a different type of printer available than the user expects, or perhaps no printer at all. The foreign network will most likely have a different name server address, gateway address, or even a different protocol. The user may use a static IP address at the home office, but require a dynamic IP configuration at the new location. Knowing how to locate and change all of these parameters is a daunting task, and can lead to a very frustrating experience while trying to get connected.

To provide for a more positive connectivity experience, we examine below the features we believe to be important in solving this problem.

Access to Data

The most important factor for mobile devices is connectivity, specifically, the ability to access critical data regardless of location [Zhao97, Yeo94c]. Mobile computers are an extension of the user's desktop or corporate network infrastructure. As such, they complement the type of functions and amount of data available on the more powerful desktop or server machines. In general, the resources available on a mobile device are much less than that of a desktop machine or even a portable notebook computer. Consequently, the applications that run on the mobile devices are in most cases the "lighter" versions of their larger, more powerful desktop counterparts, while the data stored on the mobile device is usually a selected subset of the main database. However, due to the dynamic nature of a mobile host's connectivity, providing network support for a mobile host can be a much more complex task than for its stationary counterparts [Zhao97]. Connectivity will be the single biggest differentiator between the universe of devices today and the universe of devices 10 years from today [Hela99Hela99].

Mobile workers generally have high latency and limited bandwidth for communications [Bure97]. It is therefore important to get the data quickly as there is no guarantee how long the connection will last. The software should allow the user to specify the priority of the replication tasks based on the connection type and location information. It should also allow the user to

configure each of the replication services to provide the maximum benefit while connected. For example, if the user is connected on a slow cellular or dialup connection, the software should, if the user allows, change the order of the replication. The software should also allow the user to eliminate graphics, video, audio, and large attachments to insure that the most important data is sent or received. The user should be able to limit the amount of data transferred per transaction, and to reject certain embedded applications or programs in the transferred data.

Location Transparency

Another factor is the support for disconnected operations. Ideally, mobility should be completely *transparent* to users. Transparency relieves users of the need to be constantly aware of the details of their computing environment, thus allowing them to focus on the real tasks at hand [Saty93].

Location-dependent Behavior

Depending on the physical location, users expect to perform different tasks. While on an airplane, a user does not have access to the color printer at the office. Yet he should be able to “print” his documents as if the printer is connected locally. If the system can’t detect the printer, the document should be spooled to the hard drive for printing at a later time when connected to the

network. The documents should be stored in a queue and then released in whole or in part to be printed when a connection exists. The user should also have the ability to redirect the print to another device. For example, if the user directed the output to be sent to the color laser printer and the only available printer is a black and white laser, he should have the ability to send it to the black and white printer seamlessly.

The system should also provide the ability to selectively start certain programs when at a particular location. For example, if the user works at a graphics arts studio, he might use a graphics program such as Adobe Photoshop to view his work. Instead of having to start all of his programs manually, the user should be able to specify what programs are launched automatically when he is at that particular location. If the user then travels to the country to do some writing, he would not likely need his graphics application started but would instead prefer that his word processor be started and that it would be started already editing where he left off.

Disconnected Operation

Mobile applications should be designed for use by disconnected users who require both online and deferred access [Bure97]. The software must be able to quickly capture the transferred data and cache it locally for offline access. The software manages the local data cache to insure that there is enough disk space

to handle the cached data. The user should be able to expand or limit the amount of disk space reserved for the cache, and be able to flush the cache if desired. The cache must be indexed so users can browse the cache history and roll back the cache in the event of an unexpected disconnection.

The most common disconnection scenario has been a user detaching his or her laptop and taking it home to work in the evening or over the weekend. Mobile users should be able to continue working even though the system is disconnected or weakly connected[2]. They should be able to review email, respond to it, and send new messages as if they were connected. The next time the system is docked or connected to a LAN, the pending operations should be performed. If the disconnection occurs during the execution of a distributed computation, then the computation may need to suspend until reconnecting to the network [Nobl94a, Saty93]. These frequent disconnections of mobile computers must be handled by a robust connection protocol [Zasl95].

The overall goal is to minimize the need for active intervention by users to cope with the consequences of mobility [Perk97]. An important aspect of this goal as it relates to connectivity involves the replication of data between the mobile device and a desktop or server. Replication is much more than the copying of data or an object. It must also address the implementation and management of the complete copying process [Bure97].

[2] By weakly connected, we mean slow connections such as a 4,800 baud cellular connection or intermittent connection to some type of low-bandwidth access point.

An important feature that substantially improves the usefulness of disconnected operation is the ability of the system to hoard data [Saty93]. Users should be able to specify the data to be hoarded on the system and the criteria for what data to discard during the hoarding process, using a hoarding profile.

Service Discovery

The information-gathering and administration aspects of the network configuration procedure can be arbitrarily difficult and error prone [Perk97]. Even if the configuration is known, it is possible (and likely) that the configuration had changed from the last time the configuration information was updated. In the interim, devices could have been moved or removed and are no longer configured as expected.

Without prior knowledge of the particular network configuration, the system should, whenever possible, determine the configuration through service discovery. This discovery might be possible using DNS and DHCP, but the most optimal method would be to use the Service Location Protocol, or SLP. SLP is a proposed standard for discovering and contacting network services. It defines an overall protocol used by Directory Agents (DAs), Service Agents (SAs), and User Agents (UAs). SAs advertise the services they provide by publishing them in a directory managed by the DAs. Users can then query the service database to

determine what services are available and then use those services to configure the system automatically. The services can also be published to a directory service provider such as Lightweight Directory Access Protocol, or LDAP.

Using SLP, an application can discover the network IP address and port number of the service. It can discover a printer on the network and evaluate its capabilities to see if it fits the current requirements. If the printer has the required capabilities, the application can select the correct protocol, configure the service and begin printing with no user intervention [Solo98]. It eliminates the need for the application or operating system to know the name of the printer, and makes network administration much easier.

Support for Multiple Adapters

To achieve connectivity in any place at any time, mobile hosts will likely require more than one type of network device [Zhao98]. The software that provides network-specific location management and connectivity should allow the user to specify which type of adapter to be used for that particular location. The user may connect to a Token Ring LAN while in the office, but need to connect to an Ethernet LAN at the hotel or airport lounge. The user may also wish to configure the system to favor a particular type of connectivity. For example, if the user enters a room that has an infrared access point and a Bluetooth access point, the user may opt to make the Bluetooth connection the favored medium,

emphasizing speed over cost. There may also be times that the user wants to use a dialup connection even when other access points are available. The software should allow the user to specify a set of connectivity preferences based on speed, cost, Quality of Service (QOS), or personal preferences.

Chapter 3. - Methodology

Approach

We decided the implement the first version of EasyStreet the Microsoft Windows platform, specifically Windows 98. While there was significant interest in developing this technology for the Linux operating system, we felt that developing the software for Windows would be sufficient to validate our design.

Windows 98 was chosen for the first implementation because it easier to administer and rebuild in the event of a catastrophic failure during program development. Windows 98 is an “open” system, in that users can do almost anything without being thwarted by users IDs, passwords, and system security. This makes for an easier development environment, where programmers are free to modify system parameters, registry settings, and system services without worrying about the security aspects of the system.

The development of this program involved the modifying of a large number of system parameters “on the fly”. Causing a major system failure in Windows 98 was much less likely to result in the need to totally rebuild the operating system from scratch. Because of its lax security, Windows 98 proved to be a much more

forgiving environment than Windows NT. Once the design was validated on Windows 98, we would then validate it on Windows NT.

While this approach did make development less of a hassle, it did make the development of the Windows NT version more difficult because much of the software was never tested on the Windows NT system. Worse, many of the system parameters we were modifying in Windows 98 were not in the same location in Windows NT. Differences in operating system libraries, architecture, registry keys, and file system required many parts of the code to be encased in “if” clauses. This led to somewhat fragmented code despite the abstraction provided by the C++ classes.

Data Gathering Method and Database of Study

Data was gathered from three sources. The first was a Web-based problem-reporting database where users were able to post details of bugs they found or problems that hey encountered while installing or using the software. The second method was a Web-based discussion database, where users were given the opportunity to discuss features that they thought were missing or that needed improvement, and to post their general comments on usability and ease of use. The third source was a survey form that was filled out at the end of the testing period that contained several questions and an area to enter any comments about the software.

Validity of Data

While the technical data presented in this report are accurate, some of our conclusions and indeed our thesis are based on our best judgment given the facts presented here. It is quite possible that others may form a different opinion or view based on the evidence.

However, we believe our data to be accurate, in part because our conclusions are based on the results of an actual pilot program. Over 100 users were given access to the software and asked to report back on their observations. Each user was given access to a Web-based problem-reporting database where they could post questions, report problems, and get the latest information on the software. We used the Web site to post information on known problems, workarounds, and upcoming releases. Using the information from the database, we continued to modify and iterate the program function during the pilot deployment.

Originality and Limitations of Data

The data collected during study are original but limited in scope because most of the participants were already somewhat computer-literate before being given the software to test. Further studies should be done to judge the ease-of-use during

installation and operation. These studies should be performed using a group of testers that are not computer-literate, or that have very little experience with configuring or using the system.

Summary

Based on the data we were able to gather as well as our actual experiences with these devices, we are convinced that the EasyStreet software makes the task of location management and access to data easy for the mobile worker. It simplifies the configuration of the system, allowing the user to change the entire network configuration of the machine with a single click, and without knowledge of the underlying data. The user is able to change complex system parameters through a simple and intuitive user interface, minimizing or eliminating the possibility of rendering the system inoperable. The addition of the data replication, synchronization, POP mail, Lotus Notes mail and Web page hoarding make EasyStreet a desirable addition to the software on a mobile computer or notebook.

Chapter 4. - Analysis and Implementation

Design Overview

User Interface

For the user interface, we explored several methods for displaying location information. We first tried a simple pull-down menu, but users objected to the lack of information. We then tried a tree view, using the standard Windows display tree, but users found that too cluttered. We finally decided upon an object view similar to the view used by Windows to show installed printers and dialup networking objects. Users became quickly familiar with this method because it was designed like a standard Windows application. We spent a great deal of time making all of the keystrokes and mouse operations work exactly like they work in native Windows. Another issue that we struggled with was whether or not to use the Microsoft Foundation Class, or MFC, libraries for design of the user interface. We eventually decided to use MFC, a decision that not only made coding and testing easier, but also made the program easier to use.

Multiple reboot

We closely examined all parameters to determine if changing one of the parameters really required a reboot or if the reboot was being performed for no valid reason The reason was to keep the number of reboots required to a minimum, and at worst to require no more reboots that Windows.

Local vs. Global Settings

A major design point was what settings for a particular location should be local in scope, and what settings should be global in scope. User Ids and passwords were first designed as global (the same for all locations). We soon found that users had different user IDs and passwords for different locations. There were some exceptions, however, because mail program such as Lotus Notes use the same ID from any physical location. Another issue was that of Web page hoarding selection. We first thought Web page cache selection should be local, but users complained about having to set the hoarding selection for every location. We then changed it to global, and found that users objected to the same Web pages at every location. We finally arrived on a hybrid model that allowed the settings to be local and allowed the local settings to be easily propagated to other locations or groups of locations.

Email integration

Our design included supported for email applications, yet we found that each email application had its own proprietary data format and method of cataloging mail. Some of the mail clients used a MAPI interface, while others used a proprietary object-oriented interface like Microsoft COM. Integrating our application with the various email clients was a challenge, and narrowed significantly the number of email clients that were eventually supported.

Web Page hoarding

Web page hoarding presented many of the same problems that we encountered with email integration. The two main browsers, Netscape and Microsoft Internet Explorer, each have their own methods for requesting Web pages and saving them to a local cache. While there was plenty of documentation about how to make HTTP requests, there was very little information regarding specific interfaces to the browsers. There was also very little information about how to manage the browser cache or use the integral parser and COM interface.

Access to Data

Three important features are necessary to provide location-independent access to data. These features are location management, connection management, and synchronization management.

Location Management

The settings for connectivity, networks, protocols, printers, and files are often different for each location. A company's Mobile, Alabama office might have an HP LaserJet 5 printer installed, while the office in Juneau, Alaska might have an Epson 800 inkjet printer installed. The network in Juneau might be set up to allow access to a local server for saving and retrieving files, while the location in Mobile might not even have a network installed. The communications and network settings for each physical location can be, and usually are, different. Moving back and forth between these locations can be painful, as the user must edit these parameters for each physical location.

For dialup, the access provider usually resolves differences in name servers, routers, and WINS servers, allowing the user to log onto the network from any physical location once the dialup connection has been established. In the case of

a wired LAN, connection, however, there is no system in place to resolve the differences in the LAN configuration.

Current Windows 98 and Windows NT operating systems do not permit these settings to be saved on a per-location basis. Windows does save some dialer settings on a per-location basis, but these settings are not what make traveling to different locations difficult. What's needed is a way to save all relevant settings and data on a per-location basis, and to allow those settings to become the default settings at a particular physical location. To manage location-specific settings, EasyStreet supplies a software component called the Location Manager.

Connection Management

The ability to connect to the network is a critical part of the mobile computing lifestyle. Without a connection, there's no way to access the critical data necessary for users to do their work. Prices, quotations, inventory levels and urgent mail must be delivered in a timely fashion. Most users don't have the time or knowledge to change all of the Windows parameters associated with connectivity, or to manage them over several different locations. EasyStreet solves this problem by maintaining the connectivity settings on a per-location basis. When the system is booted, the user is presented with the standard Windows logon dialog with one extra menu selection that specifies a location.

When the user selects the location, the logon process continues and the system is configured with the parameters located in the parameter database for that location.

Once the system has been configured, the Connection Manager is responsible for instantiating the connection, monitoring the state of the connection, providing a log of connection state changes, and launching the Synchronization Manager at a specific time or time interval.

Two modes of connections are supported. The user can configure EasyStreet to connect manually or automatically. In the automatic mode, the user can configure the connection to be established on a specific date, time, or time interval. For instance, a user could setup EasyStreet to dial every half hour from Monday through Friday but not Saturday or Sunday. The user can also choose to drop the dial connection at the end of the synchronization or leave the dial connection intact. When connected to a LAN with a LAN type connection, synchronization is performed across the LAN without dialing.

The Connection Manager keeps a detailed log of every connection including the start time, stop time, location, connect speed, and average throughput in bytes per second. This is important to help identify bottlenecks in the networking

infrastructure, and to determine if they are the result of a bad local connection or poor overall network performance.

Synchronization Management

Synchronization management is the ability for users to configure the content that gets delivered to their system. Five data options are available in EasyStreet that are configurable through the Location Manager user interface. The supported file/mail options are Notes mail and Notes database replication, POP3/IMAP mail, Microsoft Exchange mail, file synchronization, and Web caching or *hoarding*.

EasyStreet allows users to define sets of data (files, web pages, email) to "synchronize" between a local copy and a copy somewhere else on the network or locally on the hard disk. Synchronization can be done manually or automatically, depending on the user's preferences. Automatic synchronization is useful because it saves time. Instead of connecting, opening each program and downloading the data individually and then closing the connection, synchronization is performed automatically and, if necessary, at predefined intervals. EasyStreet caches user IDs and passwords for the synchronization clients and therefore does not interrupt the user to enter his or her ID or password when required by the synchronization client application. Using the

Location Manager's graphical user interface, the user can select the criteria that will be used for synchronization.

Service Discovery

The information-gathering and administration aspects of the network configuration procedure can be arbitrarily difficult and error prone [Perk97]. Even if the configuration is known, it is possible (and likely) that the configuration had changed from the last time the configuration information was updated. In the interim, devices could have been moved or removed and are no longer configured as expected.

Without prior knowledge of the particular network configuration, the system should, whenever possible, determine the configuration through service discovery. This discovery might be possible using DNS and DHCP, but the most optimal method would be to use the Service Location Protocol, or SLP. SLP is a proposed standard for discovering and contacting network services. It defines an overall protocol used by Directory Agents (DAs), Service Agents (SAs), and User Agents (UAs). SAs advertise the services they provide by publishing them in a directory managed by the DAs. UAs can then query the service database to determine what services are available and then use those services to configure the system automatically. The services can also be published to a directory service provider such as Lightweight Directory Access Protocol, or LDAP.

Using SLP, an application can discover the network IP address and port number of the service. It can discover a printer on the network and evaluate its capabilities to see if it fits the current requirements. If the printer has the required capabilities, the application can select the correct protocol, configure the service and begin printing with no user intervention [Solo98]. It eliminates the need for the application or operating system to know the name of the printer, and makes network administration much easier.

At the time of this writing, however, SLP has not been formally deployed and still exists as a proposal. SLP is a product of the Service Location Working Group of the IETF, and is available only for research or non-commercial purposes. More information on SLP can be obtained from the SLP Web site, www.srvloc.org.

There are several other methods that can be used to determine the network configuration. Many of these methods are specific to a particular network installation. DHCP, for example, requires that the network administrator store system and configuration parameters on the DHCP server, and that the clients know how to interpret and use those parameters. Using DHCP only works if the network administrator and the client systems agree on a format for the DHCP Options data. This requires that the clients have some knowledge of the network configuration, a situation that rarely occurs outside a company's network infrastructure.

For Java-only environments, Sun's Jini works much the same as SLP, and Novell's NDS and SAP work essentially the same for NetWare networks. For this implementation of EasyStreet, we decided not to implement Jini or SAP. However, the plug-in architecture of the Hierarchical Discovery Servers (HiDS) in EasyStreet can accommodate these other protocols easily.

Standard Windows™ User Interface

EasyStreet uses the standard Microsoft Windows™ user interface controls and operations. Double-clicking on an object will open that object if the object supports the *open* operation. Highlighting an icon and clicking the right mouse button over the icon will allow the user to display and change the properties

associated with the icon. Holding down the Shift key while selecting the icon allows multiple items to be selected.

Some configuration procedures allow the user to accept or cancel any changes that may have been made to the configuration settings. In these cases, there will always be a *Cancel* and an *OK* button. Pressing *OK* will save the new parameters and complete the operation, while pressing *Cancel* will cause any changes to be discarded.

In some cases, an *Apply* button will also be present. Pressing the *Apply* button will cause the parameters to be saved for the current dialog only. Many of the configuration dialogs in EasyStreet are presented as tabular pages, that is, the pages resemble a small notebook with small tabs at the top or side of each page. When changing parameters on a particular page, pressing the *Apply* button will cause the changes to the current page only to be saved. Pressing the *Apply* button will only affect parameters on the current page, and will not cause parameters on other pages to be saved.

ToolTips are employed on all menu icons. A ToolTip is a textbox that appears when a user taps and holds a stylus on a ToolTip control. ToolTip controls can be used in dialog boxes and other application windows. A ToolTip appears on the upper left side of the point where the user first taps the screen. ToolTips can be

used to extend the name of a control, and can also direct a user to Help files or printed documentation. If the mouse cursor is passed over the top of a menu icon and paused with the mouse cursor on top of the icon, a small caption will open to describe the purpose of that icon (see Figure 4-1. ToolTip for the import icon.). The contents of the caption are programmable.

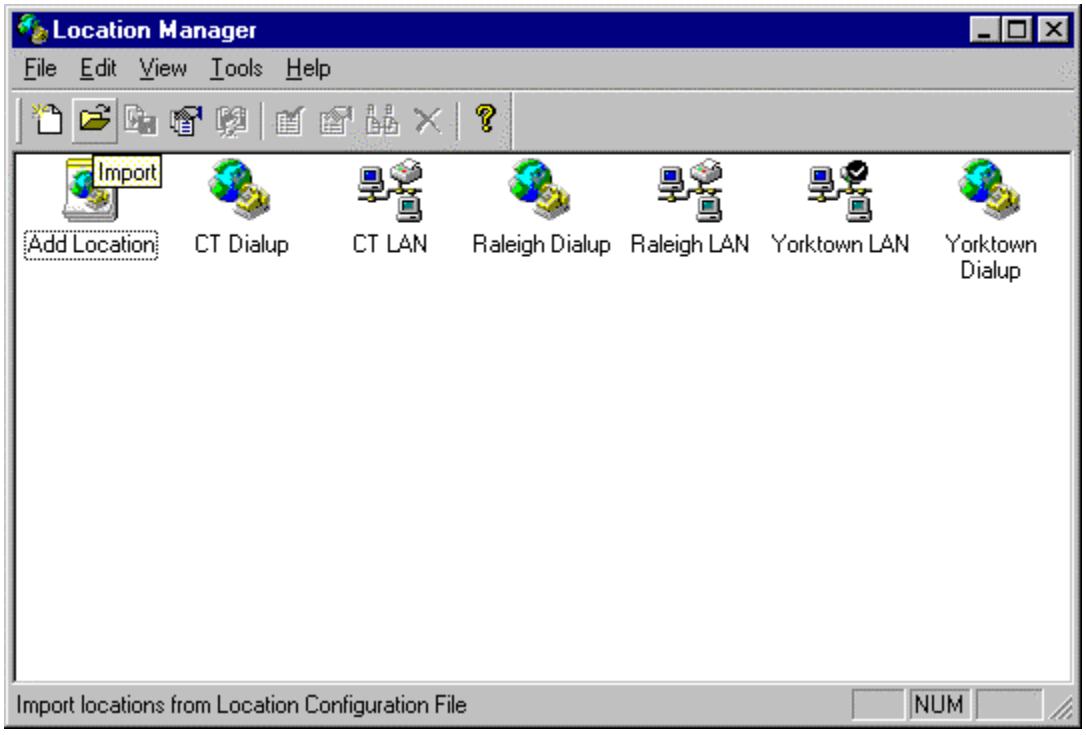

Figure 4-1. ToolTip for the import icon.

EasyStreet Architecture

EasyStreet is designed as a multithreaded Windows application using standard Windows libraries and user interface components. The two main programs are the Connection Manager and the Location Manager. Each program accesses the configuration data through a common registry access DLL (see Figure 4-2. EasyStreet block diagram.)

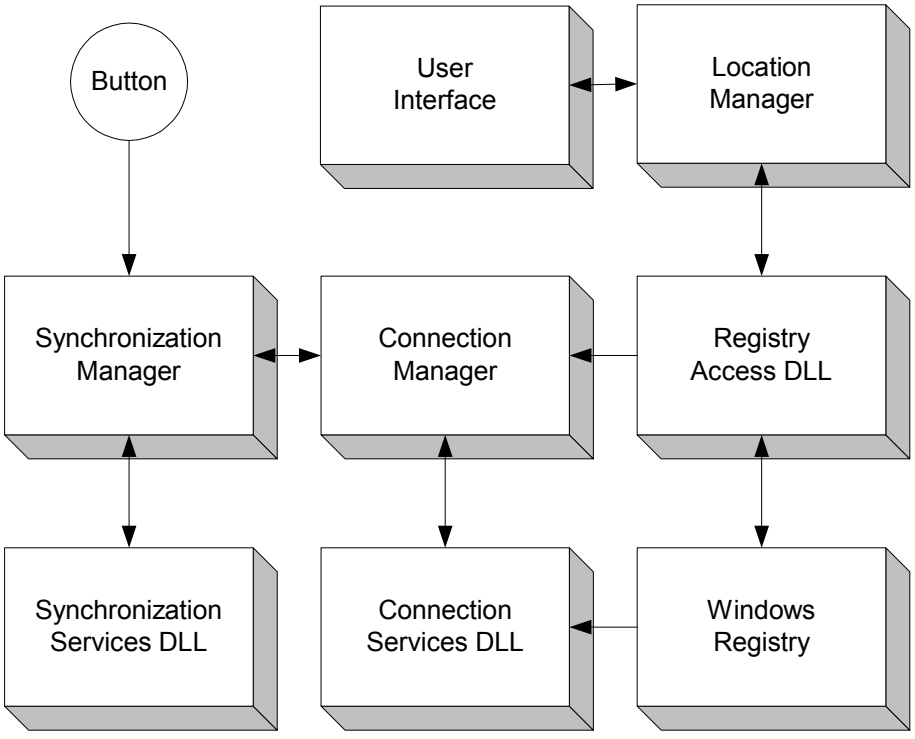

Figure 4-2. EasyStreet block diagram.

EasyStreet is a Win32 application that uses the standard Microsoft Foundation Class libraries for the user interface implementation. Although this does restrict the user interface to a Windows platform, its also allows the application to take advantage of the accessibility features of Windows. These features include very large fonts, screen magnification, large menus, and mouse trails. These features are necessary for compliance with United States Government requirements for accessible applications.

Location Manager

Location Manager addresses the needs of the corporate mobile user by providing the ability to configure multiple protocols, network stacks, IP addresses, name servers, WINS servers and domain names on a per-location basis. The Location Manager provides the user with the ability to easily import, select, and manage the connectivity settings for each defined location.

To minimize the need for manually entering location parameters, the Location Manager provides an *Import* function that allows pre-configured location information to be used. This pre-configured location information can be preloaded with the system or downloaded from a Web site and imported into the Location Manager. The location data for a location or group of locations can be exported and sent to others as a simple text file attachment or posted on a Web site for downloading by other users.

Once the data has been imported, the user can browse or change the location data if necessary. New, updated or changed location data can be imported at a later date if the information for that location has changed. User-specific parameters will not be overwritten when importing the updated location information. If the location data is valid, the user only has to select the location to be used as the current location.

Location Manager maintains the current settings for printers, networks, connectivity, network resources, and user preferences on a per-location basis. Location Manager provides a graphical user interface for the user to view or change the location-specific parameters. The Location Manager is at the center of the EasyStreet strategy, and provides the core set of database services required to support multiple physical locations.

Locations

Locations are objects that represent the settings for a particular physical location. The location objects appear in the Location Manager window as icons, one for each location. Each icon represents a location definition record in the location database. Location objects are manipulated using the standard Windows user interface paradigm. Location objects are created from a Location Configuration

(LCF) file (see Figure 4-3 - EasyStreet sample location configuration file). The Location Configuration File is designed to be a simple ASCII file in the INI file format, permitting it to be easily parsed, edited, and sent over the Internet.

A location is the computer's representation of different places and ways of connecting to the network in those places. For example, one location might be called "Office" and another might be called "Home". The name is arbitrary, and can be easily set or changed by the user through the Location Manager user interface. It is saved to identify the location object in the Location Manager window. The name of a location should be descriptive to help identify the particular location, such as "Office at Home". The location name can be up to 254 characters long, but should be kept reasonably short to make viewing easier.

```
[Chicago LAN]
ConnectionType=LAN
[Chicago LAN.Geographical]
Country=United States of America (1)
State=Illinois
City=Chicago
[Chicago LAN.Network.TCPIP]
SubnetMask=255.255.255.128
Gateways=
SocksFile=socks_chic.cnf
DNSEnabled=1
Domain=mydomain.chic.com
DHCPServer=diz.mydomain.chic.com
DNSServers=
DomainSuffix=mydomain.chic.com,chic.com
WINSEnabled=1
WINServer1=9.2.228.53
WINServer2=
EnableDHCPForWINS=0
[Chicago LAN.Network.NetBIOS]
Workgroup=mygroup
Domain=mydomain
[Chicago LAN.Sync.WebCache.WebPages.Page3]
URL=http://www.cnn.com
Update=Daily
Depth=0
```

Figure 4-3. EasyStreet sample location configuration (LCF) file.

Location Types

EasyStreet separates locations into two distinct types, LAN locations and Dialup locations. A LAN location is a location that is configured to access the network using NetWare, TCPIP and NetBIOS on an Ethernet or Token Ring adapter, and also includes locations that connect via a wireless technology such as 802.11. Dialup locations are those locations that require the system to actually dial a number and connect to a network or service provider via a modem. EasyStreet treats LAN locations and Dialup locations differently in the user interface. For example, if the location is a LAN type location, the user will not see or be allowed

to change the settings for phone numbers or dial access codes, as they don't apply for a LAN type connection. Likewise, a Dialup connection will not allow the user to view or change any LAN settings such as an IP address or the Socks configuration file (CNF) name.

It is possible, however, to be connected to a LAN while the current location is set to a Dialup type. In this case, EasyStreet will still dial the phone to perform synchronization even though a LAN connection already exists because its primary method of connecting is set to Dialup.

Changing from a LAN type connection to a Dialup type connection or Dialup type connection to a LAN type connection requires a reboot of the system to make the changes effective.

The Current Location

In the EasyStreet environment, one location must be selected as the *current location*. The current location is the location that is used to initialize the settings in the system when it is started or rebooted. The current location is always identified by a check mark next to the icon that represents the current location in the Location Manager window. This method of identifying the default location with

a check mark is similar to the way Windows identifies the default printer, and this method was chosen specifically because of its similarity to Windows.

When EasyStreet is installed for the first time, there is no current location selected. EasyStreet makes a copy of the current system settings and places those settings in a special location called the Default Settings. This action allows the system to be rebooted after the EasyStreet installation but before a default location has been chosen.

Location-specific Tasks, Applications, and URLs

Since EasyStreet always knows the city, state, and time zone of the location, it is natural that this location information be used to configure certain options automatically for the user. EasyStreet preloads several Web pages into the Web page cache depending on the physical location.

Most travelers are interested in the weather at their location, so EasyStreet loads the local weather page for the location by assembling a URL from the www.weather.com base. If the user travels to Chicago, for example, concatenating the weather subdirectory with the cities subdirectory, and adding the state and city forms the complete URL www.weather.com//weather/cities/us_il_chicago.html that is then inserted in the Web page cache configuration.

In the United States, EasyStreet also preloads the address of the page for any state into the Web page cache by building a simple URL from the two character state abbreviation. Using the government URL for states, the URL for Illinois is built as www.state.il.us and preloaded into the Web cache.

EasyStreet allows the user to set the local system time to the time zone at the current location. Setting the system time to the time at the new location allows the EasyStreet file synchronization algorithms to correctly identify newer files at the new location even if the time in the current time zone is earlier than the time in the previous time zone when the files were last synchronized.

Users can select a list of programs that will be started automatically at the new location. For example, the user may want to launch a productivity application when at the office, but a chat application or game when they are at home. EasyStreet allows these programs to be launched on a per-location basis, and keeps a list of these executable programs as port of the location object for a particular location.

Default Settings

The Location Manager maintains a special location entry that is not visible in the Location Manager window called the Default Settings. The only way to view or edit the Default Settings is to select *Default Settings* from the *File* menu in the Location Manager window. The Default Settings are used to initialize a new location with a base set of location data so as not to render the system unbootable if the system is restarted before a current location can be selected. If a new blank location is created, the Default settings are used to populate the new location with a bootable set of parameters.

Whenever a new location is created, the parameters in the new location are populated with the parameters from the Default Location. The Default Location acts as a template, initializing parameters in the new location to known values. The user can go back at any time and edit the parameters in the Default Settings location, and the next time a new blank location is created, the default values will be copied into the new location. This insures that the critical parameters necessary for normal operation are propagated to all new locations. It also prevents a user from creating a location that might result in an unbootable system.

Since the Default Settings location is created when EasyStreet is first started, the parameters used for that location remain fixed unless they are modified using the *Edit Defaults* option from the *File* menu in the Location Manager window.

However, if the user adds another modem, changes the network card from Token Ring to Ethernet, or changes the machine name, the configuration parameters in the default settings will new be incorrect. The user can reset the defaults to the values of the current location by selecting the *Refresh Defaults* option in the *File* menu of the Location Manager window. This causes the default snapshot to be updated to the new machine configuration. From this point on, any new location created is initialized with the new updated parameters. The Default Settings should be updated whenever any changes are made to the system using the Reset Defaults menu option.

Hierarchical Discovery Service (HiDS)

Service discovery can be a complicated process, especially when connecting to networks where the configuration is not known in advance. To enable this feature, EasyStreet provides a checkbox that the user can select to enable the automatic discovery of the network configuration.

The service discovery function returns a list of available services, names, attributes, etc. These may be known services or discovered when the user is connected to the network. Services discovered during the connection can be dynamic or saved by location for use at a later time. Information regarding these services is stored in a persistent database. When a location is selected, an automatic binding agent connects the services to the location.

To provide this complex automatic discovery service in EasyStreet, we have developed a new service called the Automatic Hierarchical Discovery Service, or HiDS. When directed, HiDS attempts to discover the network configuration using a variety of methods, and then exports its findings through a set of callable services. Users can specify if HiDS should be invoked automatically, manually, or not at all. Many of the discovery methods used by HiDS employ protocols that have been developed by working groups and consortiums, and most use some form of RPC to accomplish their task.

HiDS is object-oriented in its design. It abstracts the code that gathers the configuration data, and hides the configuration data in private data structures. The calling program uses method calls to obtain the configuration data. This level abstraction allows HiDS to manage the details of the network configuration data and to resolve issues such as byte ordering in big endian and little endian systems. HiDS uses a hierarchical method of discovery to determine the type of network the system is connected to. Discovery is attempted in the following order of precedence.

1. Previous Knowledge
2. Salutation
3. Service Location Protocol (SLP)
4. Directory services (LDAP)
5. Domain Name Services (DNS)
6. Dynamic Host Configuration Protocol (DHCP)

Once the configuration has been determined, the information is stored in a persistent internal database. Applications can then query the database for the network configuration and use that information to connect to a foreign network. The HiDS database can also be populated with the network configuration data of known locations by using the HiDS *Import* method. The data is presented in a flat file form to HiDS and converted to internal representation by the Import function.

The database can also be exported using the HiDS Export method. Once exported, the configuration data can be sent via normal email to other users or posted on a Web site.

HiDS uses several components to determine the network configuration. To understand how the HiDS service works, it is important to review the operation of each of the hierarchical methods and their components.

Previous Knowledge

The desired method of service discovery is not to have to discover services, but to rely upon foreknowledge of network services. If the foreknowledge does not exist, the discovery service begins a hierarchical ordered search for available network services. This is generally the most reliable method to determine the configuration of a network. Although minor changes to the network configuration may frequently occur (such as a printer becoming unavailable), major changes to the network configuration are rare. It is highly unlikely, for example, that a network administrator would change the network protocol or the IP address of the name server. Applications in a networked environment tend to be written to run in the same environment. A major change in the network configuration would render many of these applications unusable and cause unnecessary havoc.

The calling program can specify whether or not HiDS should use the existing network configuration if a valid configuration is found. If a valid configuration is found, HiDS returns a “successful discovery” status to the calling application. The calling application can then query HiDS for the network configuration parameters. If the caller specifies that a discovery should be performed regardless of the contents of the network configuration, the discovery process continues with the next step.

One of the goals of EasyStreet is to have the network configuration preloaded and not require the configuration and resources to be discovered. We expect, for example, that all of the Marriott hotels will provide the same type of network topology to maintain standardization among the hotels and to benefit travelers. It is logical that the Admiral’s Club in major airports will follow the same guidelines to minimize support problems and to provide users with a good connectivity experience. Once the configuration has been established, the user should be able to use the generic Marriott or Admiral’s Club network configuration anywhere in the United States. We acknowledge that providing the same generic configurations for independent franchises may be more difficult, but that we will likely be able to gain network access using a generic Ethernet adapter with TCPIP and DHCP.

Salutation Protocol

The Salutation Protocol is a product of the Salutation Consortium Inc., a working group consisting of 30 companies representing various facets of the computer industry. These companies provide a wide range of products including computer hardware, software, fax machines, copiers, and online services. Their goal is to provide a standard protocol for discovering devices and their capabilities on the network while hiding the details of the underlying protocol and implementation from applications. Salutation also provides a service broker to manage resource interactions called the Salutation Manager.

Salutation utilizes Service Function Units, or SFUs, to provide a way for services to advertise themselves, which is analogous to SLP's Service Agents, or SAs. Salutation clients are called Client Functional Units, or CFUs, and are analogous to SLP's User Agents, or UAs. Requests for resources are sent to the Client Salutation Manager that communicates with other Salutation Managers to locate the desired service. If the desired service is located, the Client Salutation Manager reports its findings to the CFU. The Salutation Manager acts as a broker or proxy for between the CFUs and SFUs and handles the details of registering for the service and contacting the SFU. This abstraction allows the CFU to contract for services without understanding where the services are located or the protocol used to access the service.

Salutation is an “open architecture”, and as such is not controlled by any one company but by the Salutation Consortium. Because it is open, there are no royalties. Salutation is independent of any program language, network protocol, or operating environment. While other protocols are dependent on IP, Salutation can operate on networks that use any protocol, including NetWare. Because it is language independent, Salutation may be more desirable than Jini, which relies exclusively on Java, or UPnP, which relies heavily on Microsoft and Intel.

More information on the Salutation protocol can be found at the Salutation Web site, www.salutation.org.

Service Location Protocol (SLP)

SLP provides a flexible and extensible environment for service discovery on the intranet. The extensibility of SLP is provided by the ability to add or remove network resources dynamically, and to have those changes reflected to the clients via the Service Agents. When a User Agent, or UA, needs to obtain information about the services available, it contacts the Service Agent, or SA, to determine what services are available.

An SA represents each service on the network. Printers fax machines, copiers, databases, and mail servers all have their own SA that is responsible for advertising their services. Collections of similar services are grouped together

into administrative domains called *scopes*. UAs can specify the scope of service when requesting services. For example, a UA might request the services of a 300 dpi black and white laser printer. A 1200 dpi color laser printer is also suitable for the job, and will likely appear in the same administrative group as the 300 dpi black and white laser printer.

SAs advertise their services by registering with a Directory Agent, or DA. The SAs specify their capabilities with a list of attribute-value pairs at the time of registration, and can add or modify these attribute-value pairs at any time. They can also include a time that the service will be available for, either forever or for a limited time. When that lifetime is exceeded, the service is removed from the DA and no longer advertised. In the event of a catastrophic failure such as a power loss, services can be explicitly removed the directory agent. Once the user has registered for a service, the service is guaranteed to be there until the user releases the service.

What makes SLP so flexible is the ease in which services can come and go without affecting the normal operation of the network. When a new service is added, the service agent advertises its services. The user agent discovers the service, and if it wants to use that service, it makes a request to the service agent for that service. If granted, a service handle is returned to the UA and used by the application to access the service. When the application has finished with the service, the user agent releases the service, making it available to other

applications. Once the application is granted access to the service, the service agent insures that the service remains available to the application until the service is released.

With SLP, the UA specifies the service it needs by specifying the service type and attributes in a message sent to a DA, and waits for a response. In some installations, no DAs are implemented, so the UA must send its requests directly to the SAs. If the service is available, the user is returned a handle that is used to access the service. When the user is finished with the resource, it returns the handle to DA or SA, making the device available to other users.

Lightweight Directory Access Protocol (LDAP)

LDAP, or Lightweight Directory Access Protocol, is an emerging standard for users to look up services, and for services to advertise their availability. The user must know beforehand the address of the directory to begin the search at, as well as the database schema of the LDAP data. Unlike SLP, LDAP does not make available the attribute descriptions for existing resource. LDAP uses the X.500 naming convention that has been standardized by the ISO specifications which makes LDAP more desirable by those companies that have standardized on the ISO requirements.

As the name implies, LDAP is based on a *directory*. In the context of service discovery, a directory is a special-purpose database used to store information about services. This information is usually stored as collections of *attribute-value* pairs that represent the services and their attributes. It is important to point out that the directory database is not a typical database. It cannot handle a large number of updates, it cannot provide atomic transactions, and it can't be used to store files like a normal directory.

The LDAP directory is more like a Windows INI file[3] or registry database. If a program uses an INI file or registry settings to configure itself, the registry or INI file must exist on that machine, very often in the same file directory as the application. An LDAP enabled application can publish its preferences in an LDAP directory, allowing those preferences to be accessed from a different machine or physical location. For example, many users configure their desktop by arranging icons, shortcuts, buttons, windows, and other controls to accommodate their personal preferences. If that user logs on to another machine, he or she inherits the default configuration of the desktop layout on that particular machine. LDAP allows those settings to be stored in a directory on another system anywhere on the intranet or Internet. When the user logs on to another system, the preferences stored in the LDAP directory can be used to configure the desktop on the new machine.

In order for applications to take advantage of the LDAP directories, those applications must be modified to be LDAP-aware. In the Windows implementation, the LDAP services and protocol are implemented as a Windows DLL. The application program must be linked to the LDAP DLL import file. When the application is run, the operating system loads the DLL with the application. Although it is desirable for applications to be modified to support LDAP, a program can be written to invoke specific LDAP APIs to discover various services on the network and report back to the application with the proper attribute-value pairs that belong to a particular service.

Domain Name Services (DNS)

Some network configuration information can be discovered using the domain's name server. However, before the client can use the name server, it must get the IP of the name server to be able to resolve the names. This is a classic chicken-and-egg situation where the client knows the name of the name server but the dame can't be resolved to an IP address because the client doesn't have the IP. To get around this, the DNS server contains what's called an "A" record that contains the IP address of the name server.

[3] The Windows INI file is a flat ASCII file that contains attribute-value pairs. It is read by the application and parsed by the standard Windows libraries to extract the data. The program then uses that data to configure the application.

DNS SRV Record

This proposed method of service discovery requires that information regarding the available services be resident on the network's domain name server. External users that wish to obtain information regarding available services query the network's name server to discover services that are stored in a special DNS record called the SRV record. When services come or go, the SRV records in the domain name server must be updated, making DNS SRV a less dynamic form of service agent. SRV records are lumped together, making it difficult for users to locate specific services.

DNS MX Record

The client can query the DNS server for the MX record that contains the address of the mail server responsible for sending mail to systems on the domain. The client can then use the contents of the MX record as the address of the mail server. Using MX records, the mail server responsible for sending mail on the network can be located anywhere in the network. There can be more than one mail server, and they can be arranged in order of precedence. The client can contact the primary mail server, for example, and if it is not available, the client can try the next mail server in the list until it finds a mail server that is active.

DNS Start Of Authority (SOA)

Using DHCP, the client can get the address of the network's Domain Name Server. Using this address, then client then queries the DNS server for the Start Of Authority, or SOA, record. This record contains the domain name and email address of the person responsible for administration of the domain.

DNA NS

The client can query the DNS server for the authoritative DNS server that is delegated to provide name lookup for that particular domain. The NS record contains the name of the authoritative DNS server for the domain.

DNS PTR

The client can query the DNS for the name of a network client that is associated with a particular IP. Thus if the client knows the name of a system or resource, it can query the DNS server for the IP address so the client can contact the system or resource directly.

Dynamic Host Configuration Protocol (DHCP)

DHCP is a request/response protocol, meaning that the DHCP client sends a Discover message seeking out any DHCP servers on the network. If a DHCP server is found, the server responds with an Offer message. The client examines the configuration information in the Offer message, and chooses the offer to accept. The client then sends the server a Request message with the offers it wants, and the server sends an ACK message back to the client of the offers are granted. Once the server has granted the offers, those parameters are locked in and are no longer available to other clients. The most common use of the DHCP protocol is the automatic assignment of an IP address to a client, and it is performed as follows:

1. Client broadcasts message to locate a DHCP server
2. Server responds with proposed IP address
3. Client agrees and sends Request to server
4. Server responds with ACK to confirm IP was granted

The DCHP protocol is implemented with DHCP messages. The DHCP messages are:

- DHCPDISCOVER – the client sends this to discover the DHCP server
- DHCPOFFER – the server sends the offer to the client
- DHCPREQUEST – the client requests one of the returned offers
- DHCPACK – the server acknowledges and grants the request
- DHCPNAK – the server denies the client request
- DHCPDECLINE – the client declines the server's offer
- DHCPRELEASE – the client releases the temporary IP address
- DHCPINFORM – the client requests information from the DHCP server

DHCP is built on top of the BOOTP protocol. It allocates or "leases" IP addresses for network clients and provides storage of parameters for clients on the network. The network administrator stores the parameters on the DHCP server, and when the clients contact the DHCP server, the server hands out the parameters to the clients that request them. The DHCP server provides a persistent storage of the network parameters so they can be restored in case of a power failure.

The DHCP client sends the DHCP server a request to lease an IP. The association of the IP address a particular client is called *binding*, which means that the IP address is bound to the client for a predetermined amount of time. This time is called the *lease time*, and the value determines how long the IP

address is valid. When the lease time expires, the client sends the DHCP server a request to renew the current IP address. If the request is granted, the IP lease time is reset. If not granted, the client may ask for a new IP address from the DHCP server. When the IP is granted with an ACK from the server, the new IP address is bound to the client for the duration of the lease time.

The parameter data stored on the DHCP server is sent to the clients when they request it in special fields called Options. The Options field is a variable length field and can contain any information that the DHCP server and client agree upon beforehand. This makes the DHCP discovery process valuable only of the server and clients are aware of each other and have previously agreed up the format and content of the server-resident data.

The Options data consists of 76 variable length values. Each Option is specified by a reserved value defined in the DHCP RFCs. Some of the most popular request that are issued by the client to the DHCP server are:

- Gateway (router) address
- Domain Name Server address
- Cookie server
- LPR server
- Domain name
- Broadcast address
- NIS server list
- NetBIOS scope
- POP3 server
- NNTP servers
- SMTP servers

For users that connect to the same network all the time, DHCP is an easy way to provide information about network services, although it will work only on the intranet, and not on the Internet. Unlike agents that advertise the services available on some networks, the DHCP client must specifically ask for the configuration data it is interested in, and the network administrator must always keep the server-resident data current.

HiDS Implementation in EasyStreet

If the EasyStreet location object contains the network configuration for the target location, that configuration is used to connect to the network. By default, the discovery service always relies upon known information about the network configuration and will only attempt to determine the network configuration if information is missing, incorrect, or the user requests a new discovery scan.

Once the configuration has been determined, the information is automatically stored in the location object for that location. If the changes to the network configuration require a reboot, the user is instructed to reboot the system to effect those changes. Even if the configuration is already known, the user can request the discovery service to perform a new discovery to determine if the configuration has changed or if new devices have been added. If the changes to the network configuration require a reboot, the user is instructed to reboot the system to effect those changes. Even if the configuration is already known, the user can request the discovery service to perform a new discovery to determine if the configuration has changed or if new devices have been added.

EasyStreet uses a single function call to discover the network configuration and services available. The function is called with three parameters.

The first parameter specifies how the discovery function locates resources and configures the network. The caller can select the specific discovery method to use or allow the discovery function to run automatically using the hierarchical method.

The second parameter specifies the services or configuration information to locate. The calling program can select from several options such as ALL, DNS_SERVER, LOCAL_GATEWAY, and several others. The discovery service will attempt to locate the specific information using the method specified by the first function argument.

The third parameter contains a pointer to the location object that is created and owned by the calling program. Once the parameter is found, the location object is updated with that configuration information. It is quite possible that the information requested by the calling program cannot be located using any of the prescribed methods. In this case, the information in the location object is not updated and the status NOT_FOUND is returned to the caller. The calling program can then choose the next course of action, which might be to use another discovery method or abandon the search entirely.

As the name suggest, HiDS performs its discovery in a hierarchical fashion. It first checks to see if the network configuration is known. It verifies the values in the location object, and if enough information is found to connect to the network,

the discovery services returns SUCCESS. The user can specify that the discovery be performed even if the configuration exists and appears to be correct and complete. This option is set to OFF by default because the discovery process can take several minutes to complete.

First, the discovery module attempts to discover the network configuration and resources using the Salutation protocol. The module calls slmQueryCapability() to determine of the Salutation Manager (SLM) is present, indicating that the network supports the Salutation protocol. The SLM returns a list of SLM Ids that are linked to SFUs. The module then does a life check on the SLM and then verifies that the Service Functional Unit (SFU) is available. If so, the discovery module calls slmOpenService to access the service described by the SFU. The SLM then in turn calls fnOpenService to ask the specific Functional Unit for access to the service.

The client then calls slmTransferData to send data to the selected Functional Unit. The SLM calls fnReceiveData to received data from the Functional Unit. This communication proceeds until there is no more data to be sent or received, and the client calls slmCloseService to return the service to the SLM. The SLM in turn calls fnCloseService to release the service from its current obligations.

In the next step, the discovery module attempts to discover the network configuration using SLP. The discovery module, posing as a User Agent (UA),

looks for services on the network using the SrvTypeRqst message in a unicast or multicast fashion. If the user is searching for a group of similar services, the multicast request is used to contact the DAs, while the unicast is used to contact a specific DA. The SA responds with a list of host names or IP addresses of the SAs that match the scope of the UA's request in the SrvTypeReply response. The UA then issues a SrvRqst for the service, and receives SrvRply with the status of the request and the address of the service. The UA then contacts the service to retrieve the attributes of the service with the AttrRqst message, and receives the AttrRply message with the contents of the selected attributes.

In the next step, the discovery module attempts to discover the network configuration using LDAP. One slight hitch here is that the call to ldap_init() that initializes the LDAP library a returns a session handle requires the name of the domain. If the domain name exists in the location object, the LDAP prefix is simply added to the domain name. If the domain name is not known, the discovery module uses DHCP to obtain the domain name, and then adds the LDAP qualifier. For example, of the domain is called abcboats.com, the LDAP host name becomes ldap.abcboats.com. The discovery module calls ldap_init(), and if a handle is returned, it uses a series of LDAP function calls to gather information about the network and uses that information to fill in the location object. If a valid handle is not returned, the module assumes that LDAP is not installed or support on the network, and proceeds to the next step in the discovery process.

In the next step, the discovery module attempts to locate the network configuration using DNS services. If the DNS server is known, the discovery module requests information about the network using the DNS MX, SOA, SRV, and NS record queries. If the address of the DNS server is not know, the discovery module uses DCHP to find the address of the DNS server, then uses the DNS record queries to discover the network configuration and services.

In the next step, the discovery module uses several DHCP messages to query the DHCP server on the network for configuration information. Using DHCP, the module attempts to location the basic network configuration and services, including the default gateway, LPR server, cookie server, network broadcast address, NetBIOS information, POP3 server, news server and NIS server list.

In the final step, the discovery module sets the default configuration to DHCP.

Caller ID

The Caller ID feature in EasyStreet allows a user to connect inexpensively using a local analog phone line without knowing what the area code is in the new location. This is very helpful because the area codes for certain localities have been changing rapidly as the phone companies scramble to provide more numbers for their customers. This feature can also be quite useful for the traveler with very little time to spare while running from gate to gate trying to make a connecting flight.

To use the Caller ID feature, the user connects his or her system to an analog phone line and clicks the Synchronize button. EasyStreet configures the modem and dials a special 800 number. When the connection is established, the server sends to the client the area code of the location that the user is dialing from. EasyStreet saves the area code and disconnects the phone line. Using the area code as an index, EasyStreet locates the nearest local number, initializes the modem and automatically redials the call using the new number. EasyStreet performs the required synchronization and then hangs up the phone. The local number is saved in the location record and will be used to connect the next time. Depending on the user's preferences, EasyStreet can also change the user's system clock to reflect the time and date at the new location.

In addition to providing an automatic dialup number selection, the area code is used to provide specialized local URLs for travel-related information such as weather, flight schedules, entertainment information and restaurant recommendations. The area code is mapped to a particular geographic area and the travel-related URLs and parameters are built using the location information. The synchronization engine then uses those URLs to cache the selected information so it can be viewed off-line, providing the user with location-specific travel information.

Travel Schedule

This feature allows the use to import a travel schedule in electronic form. The data contained in the travel schedule is use to automatically select a location, set the time and date for that location, and automatically perform location-specific operations at the new location. EasyStreet extracts the destination information and flight times, and saves this to be used the next time the system is restarted. It also takes a snapshot of the system clock at the time the data is imported.

If the system remains on, EasyStreet continually compares the target time (the snapshot time plus travel time) to the system clock. When the target time is reached, EasyStreet displays a dialog asking the user if it is OK to change to the new location. If the answer is yes, EasyStreet automatically selects the new

location as the default, and performs any location specific tasks as specified by the user.

Connection Manager

The Connection Manager is a multi-threaded Win32 application that provides connection and synchronization services for EasyStreet. The Connection Manager is responsible for dialing the phone and establishing a dialup connection for dialup type locations, and for establishing a LAN connection for network type locations.

Connections can be established manually or automatically. In the case of a wired LAN, the connection is assumed to be instantiated at boot time. The Connection Manager periodically monitors the LAN connection using periodic pings of the local gateway as an indication of connectivity.

In the case of a Dialup type connection, the actual connection may not have been instantiated even though the Connection Manager is running. For example, the user may have configured EasyStreet to connect once every hour, download any new data and then hang up the phone. In this scenario, the Connection Manager will automatically instantiate the connection, call the Synchronization Manager to get the new data, and then hang up the phone.

The Connection Manager is started at boot time and runs in the background as a lower priority task. It is always running, monitoring the time and date to determine if it is time to perform a synchronization. EasyStreet can be configured to perform manual or automatic synchronization using the Synchronization Settings for the location. If EasyStreet is configured for manual synchronization, synchronization will be performed only when the user clicks the *Synchronize* button on the Connection Manager dialog.

Connection-type Preferences and Logging

The Connection Manager monitors the type of connection and the average data rate for any instance of a connection. The type of connection and average throughput is logged to allow a post-connection evaluation of the connection.

While a connection exists, the Connection Manager monitors all operations and reports the status of the operations to the user interface and the log file. The Connection Manager actually calls the Synchronization manager to perform the synchronization and replication tasks, and checks the return codes from the various data retrieval modules.

EasyStreet allows the user to set preferences for certain types of connections. For low speed connections, the user can choose to ignore audio and video streams, large attachments, and bitmaps. The user can also specify the priority

for each operation to perform while connected. If, for example, the user is waiting for an important email message, EasyStreet can be configured to fetch the user's email first before performing any other operations.

Synchronization Manager

The Synchronization Manager is a multi-threaded Win32 application that provides file synchronization, mail synchronization, and Web page hoarding.

File Synchronization

File synchronization is the synchronization of user-specified files on the local hard disk with local files or files on mapped network drives. The Synchronization Manager does not maintain a separate copy of each file for synchronization, but instead maintains a list of files along with necessary meta-information needed to determine if a file has changed since the last synchronization. This permits files that need to be synchronized to remain in their current location and to remain accessible by other applications.

With the Synchronization Manager preferences, the user can select the type of file replication from three methods: host overwrites client, client overwrites host, or synchronize the files. The user can also select if on which type of connection the replication should take place. For instance, the user may not want large files

to be replicated on a cellular connection. The user can also select what the default behavior is for conflicts, and can specify if the file or file extension type should be synchronized automatically or manually. The user can omit specific file types (GIF, JPEG, BMP, etc.) from the replication process.

Using the Synchronization Manager graphical user interface, the user specifies the set of the files that need to be synchronized by file extension. This specification includes the source and target file name, source and target file location, and the policy used for replication. Files are specified by their extension. Only copy semantics are supported for synchronization.

When the Synchronization Manager is launched, it checks to see if a connection already exists. If not, the Synchronization Manager calls the Connection Manager to establish a connection. Once connected, the Synchronization Manager traverses the list of files to synchronize and performs the synchronization. For each file in the list, the Synchronization Manager performs the following steps:

1. Opens a socket connection with the server.
2. Authenticates the user to the server. If the user is denied access, the Synchronization Manager flags an error and continues with the next file on the list.
3. Retrieves the timestamp of the file from the remote server
4. Determines if the file at the server has changed. This is determined by comparing the timestamp of the file as of previous synchronization.
5. Determines if the local copy of the file has changed by comparing the timestamp of the file as of previous synchronization.
6. Updates the Meta information (timestamp etc) associated with each file in its local data structures.
7. Updates the Meta information (timestamp etc) associated with each file in its local data structures.

The file synchronization module uses the following policies.

- If the server copy is missing then the local copy is propagated to the server.
- If the server copy is deleted then the local copy is not deleted.
- If the local copy is deleted then this deletion is not propagated to the server.
- If the local copy is missing then the server copy is retrieved from the server.

- If the local copy has changed and the server copy has not changed then the file is sent to the server.
- If the server copy has changed and the local copy has not changed then the local copy is overwritten with the server copy
- If both the local copy and the server copy have changed then based on the conflict resolution policy specified by the user, appropriate action is taken. The action could be to keep server's copy, keep local copy, or skip reconciliation and notify the user.

Mail Synchronization

The mail synchronization retrieves POP3 mail from a mail server and sends any mail that has been queued by the user using an SMTP server. The supported mail clients are Lotus Notes™ and Qualcomm Eudora Light.

If the Lotus Notes mail client was specified, the Synchronization Manager initiates Notes Replication. The Synchronization Manager uses the Lotus Notes C++ APIs to implement mail synchronization. The Synchronization Manager does not duplicate any functionality that Lotus Notes provides for replication. The mail synchronization scheme does the following.

1. Initializes the Notes runtime using LNNotesSession::Init().
2. Retrieves the Mail database from NOTES.INI file using LNNotesSession::GetMailDatabase().
3. Retrieves the Mail Server using LNDatabase::GetServer().
4. Initiates replication with the Mail Server using LNNotesSession::Replicate(). Note that this call synchronizes all the database replicas that reside on the server. If a user has multiple replicas of their mail database on the machine (a highly unlikely but possible scenario), all the replicas will be synchronized. Replication statistics are logged in the EasyStreet synchronization log file.

Lotus Notes replication requires that the user enter a password to authenticate itself and initiate the operation. This could be inconvenient if the Synchronization Manager is launched by automatically by the Connection Manager. To programmatically supply the password to Lotus Notes, the Synchronization Manager uses a Lotus Notes C API facility called the Extension Manager. The Synchronization Manager registers a callback with the Lotus Notes runtime, and supplies Notes with the password when the callback occurs.

Web page hoarding

Web page hoarding handles pre fetching and caching of user specified web pages for disconnected browsing. Using the Synchronization Manager graphical user interface, the user specifies the URLs of the pages to hoard.

For each URL in the list, the Synchronization Manager performs the following steps.

1. Opens an HTTP socket connection with the URL server.
2. Performs a conditional GET operation for the URL. A “conditional GET” allows a page to be retrieved if it has been modified since a specified time. If the Web server returns an HTML page, the Synchronization Manager updates the local copy. It also parses the HTML page to retrieve any contained URL references that are not in its URL list.
3. Updates the meta-information (timestamp etc) associated with each URL in its local data structures. Additional URLs hoarded during this synchronization are also inserted in the URL list.

Configured as a local proxy server, it can run with any web browser and does not depend on any server-side infrastructure. The services provided by Web Hoarder include the following:

- Selection: Specification of what web pages should be hoarded and how hoarding should be performed.
- Browsing: Searching and viewing of hoarded pages.
- Administration: Configuration of the hoarder and monitoring of the hoarding process.

Web Hoarder provides two mechanisms for the user to quickly locate and browse a hoarded page. First, Web Hoarder maintains a directory of the hoard and displays it with a tree-structured view. Base URLs are listed by categories (a predefined category named *ALL* contains base URLs from every category). Each URL has all embedded links as its children. Base URLs within each category may be sorted by various attributes, such as

- URL
- Name
- Date of download
- Date of modification
- File size

Web Hoarder downloads web pages using a list of URLs as the starting point. URLs (called *base URLs*) may be added to the list via a number of means:

- The user explicitly enters a complete URL.
- The user grabs the current web page from a browser.
- The user supplies a bookmark file, all URLs in which are imported to Web Hoarder.
- Web Hoarder picks up URLs embedded in locally replicated e-mail.

If desired, further information may be specified for each site to be hoarded, including:

- Mnemonic name for the site, which defaults to the title of the starting page.
- Category under which the site should be filed. Categories are a mechanism to group related or similar sites. The user may define categories in any way that meets his needs.
- User ID and password, if required by the site.
- Update frequency, i.e., how often the hoarded site should be refreshed. Options are no update (one-time download), hourly, daily, weekly, monthly, yearly and any time interval the user specifies.

Filters are used over fast and slow network connections respectively. A filter controls what pages linked to the base page should be hoarded. A filter allows the user to specify the type of hoarding using the following criteria.

- File type. The user can request that certain types of files are included/excluded from hoarding. The user provides a list of common file extensions grouped by MIME type, subject to additions. The user may make selections based on file extensions and/or MIME types.
- File size. The user can instruct Web Hoarder to bypass files that are beyond a given size.
- Depth. This controls how many levels of links should be followed from the start page.
- Location. The user can also specified that hoarded pages should be limited to those in the same directory or on the same site as the base URL. Separate location properties may be given for regular web pages and for resource pages (images, sounds etc.).

Filters are reusable, and thus make it easy to hoard different web sites using the same control method.

While disconnected, HTTP requests from the browser are directed to Web Hoarder, which in turn services the requests from the hoard, i.e., the collection of hoarded pages. Web Hoarder also allows multiple HTML forms to be filled in and queued without being connected to the network. These forms are sent to the server the next time a connection is established.

Application Programming Interface

The Persistent Store Manager implements platform specific functions for accessing the persistent store. It implements the following APIs.

readFile(char *pathName, char &dataBuffer) -- returns the contents of the file in the dataBuffer.

writeFile(char *pathName, char* dataBuffer) -- writes the contents of the dataBuffer to the specified file location.

deleteFile(char *pathName) -- deletes the specified file.

statFile(char *pathName) -- returns the meta information associated with a file. The meta information includes access control, last modified, ownership information associated with the file.

The synchronization engine implements the core logic for performing synchronization. It implements the following APIs.

int **connect** (char *serverName, int portNumber, char *userName, char *userPassword, ConnectionHandle * hConn) -- establishes a TCP/IP connection with the SyncManager at the specified (server, port). Uses the userName and password information for authentication.

```
int synchronize(ConnectionHandle *hConn, char *sourceFile, char
*destFile, void *policy) -- synchronizes the sourceFile with the
destination file located at the server specified by the connection
handle information. The policy information for synchronization can be
passed in as a parameter (TBD).

int disconnect(ConnectionHandle *hConn) -- disconnects the connection
with the specified (server, port).
```

The communication subsystem implements APIs for sending and receiving messages. It implements the following APIs.

```
int connect() -- Establishes a connection with the server.

int sendMessage(MessageHandle *msg) -- sends the designated message to
the server.

int receiveMessage(MessageHandle *msg) -- receives a message from the
server.

int accept() -- Waits for a client to connect.

int disconnect() -- Disconnects the current connection.
```

EasyStreet Operation

EasyStreet requires one location to be selected before the system is restarted. This location is referred to as the Current Location. Only one location can be the Current Location at any given time.

To allow the user to select the Current Location at boot time, EasyStreet replaces the standard Windows login dialog with its own login dialog. In addition to the user ID, password, and domain logon fields normally found in Windows login dialog, the EasyStreet login dialog contains a field where the user can select the name of the location to be used to set up the system. The currently installed locations are presented in drop-down form, allowing the user to pick one of the locations to be used as the current location. After selecting a location and then clicking *OK*, the system is configured with the data from the location selected and the boot process continues.

If the user does not want to change the location at boot time, or doesn't want to use the alternate EasyStreet login dialog, EasyStreet can be configured so that the standard Windows login dialog will appear at the next reboot by clearing the checkbox at the bottom of the login dialog.

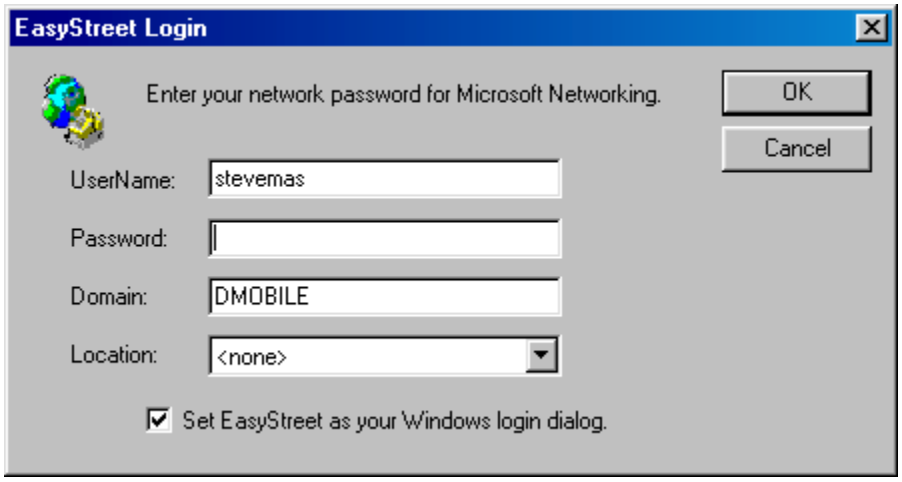

Figure 4-4. EasyStreet boot dialog.

The Connection Manager status dialog

While the Connection Manager is running, it continually monitors the time and date to determine if it is time to synchronize. It also monitors the status of all connections and synchronization operations, and reports that status to a central status dialog that is part of the Connection Manager. The status dialog is displayed by default, but can be minimized to the task bar by double-clicking the small ‘x’ on the top right-hand portion of the status dialog. If the user clicks the ‘x’, the window will no longer be visible. To make the window visible again, the user simply clicks the Connection Manager button on the Windows task bar with the left mouse button.

The connection status is continuously updated even when the status dialog is minimized. Note that unlike some programs, clicking on the ‘x’ on the Connection Manager window will not stop the Connection Manager but will just minimize the dialog. A small icon will appear in the Windows task bar indicating the Connection Manager is still running. Double-clicking on this icon will make the Connection Manager dialog visible again.

Connection and synchronization progress is monitored by the Connection Manager and displayed in the Details dialog. To view the status of a current operation, the user clicks the Details button on the Connection Manager status dialog. The status dialog expands in size, revealing the status of the current or past synchronization, the time the next synchronization is scheduled to run, and the status of the system battery if the system has a battery installed.

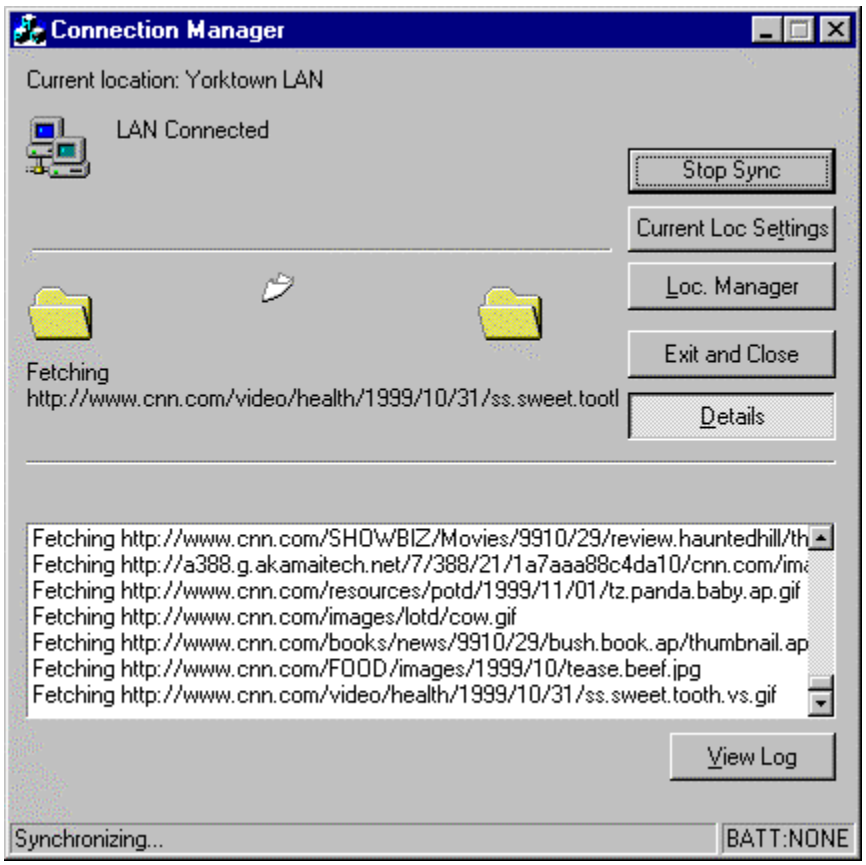

Figure 4-5. Connection Manager status dialog details view.

When the operation is complete, the scroll bar in Details section of the status dialog can be used to scroll through the completion status of each operation.

The content of the status dialog is only valid while the Connection Manager is running. If the Connection Manager program is stopped with the *Exit and Close* button, the status information will not be displayed in the Details dialog the next time Connection Manager is restarted. However, the Connection Manager does keep a detailed log file that can be accessed by clicking the *View Log* button. This button launches the Windows WordPad application to view the log file,

where the log file can be saved with different name, saved to a floppy disk, or printed.

```
*********************************
Log file opened 09/20/99 12:08:53
*********************************
[12:8:54:200]  LAN Connected
[12:8:59:690]  Starting File Synchronization
[12:8:59:690]  Filter Options = [since = August 16, 1999, ignoreover=1 MB]
[12:8:59:750]  File Synchronization Completed
[12:9:0:20]    [LotusMailSync]
[12:9:0:20]    Starting Lotus Notes Mail Synchronization
[12:9:0:20]    URLs = 1, Attach = 1, Loc = stevemas.nsf
[12:9:1:280]   [LotusMailSync] to retrieve new mail
[12:9:1:280]   Retrieving Mail
[12:9:10:130]  [LotusMailSync] Notes Sync failed - None of the databases has a replica
[12:9:10:130]  Notes Sync failed -- None of the selected databases has a replica
[12:9:10:350]  [LotusMailSync]
[12:9:10:570]  Starting Lotus Notes Database Synchronization
[12:9:10:730]  [LotusDatabaseSync]   Notes Database Synchronization Completed
[12:9:11:10]   Lotus Notes Database Synchronization Completed
[12:9:11:340]  Starting Web Page Synchronization.
[12:9:11:670]  Fetching http://dmobile.watson.ibm.com/ec
[12:9:14:850]  Fetching http://w3.ibm.com/think/images/bt_091699.gif
[12:9:15:130]  Fetching http://w3.ibm.com/think/images/bt_091699.gif
[12:9:15:400]  Fetching http://www.ibm.com/i/v9/m/en/i_download.gif
[12:9:15:680]  Fetching http://w3.ibm.com/images/odot.gif
[12:9:16:0]    Fetching http://www.cnn.com/images/1999/04/free.email.120.gif
[12:9:16:280]  Fetching http://www.ibm.com/i/c.gif
[12:9:16:550]  Fetching http://w3.ibm.com/applets/stockapplet/images/reloadicon.gif
[12:9:16:830]  Fetching http://w3.ibm.com/applets/stockapplet/images/reloadicon.gif
[12:9:17:160]  Fetching http://www.ibm.com/i/v9/m/en/hn_home.gif
[12:9:17:430]  Fetching
               http://a388.g.akamaitech.net/7/388/21/c7c8272db45981/cnn.com/images/99
[12:9:17:710]  Fetching
               http://a388.g.akamaitech.net/7/388/21/7223b3eca936a5/cnn.com/images/19
[12:9:17:980]  Fetching http://w3.ibm.com/images/buttons/btn_arrow_333.gif
[12:9:18:310]  Fetching
               http://a388.g.akamaitech.net/7/388/21/978eb42451f831/cnn.com/images/19
[12:9:18:590]  Fetching
               http://a388.g.akamaitech.net/7/388/21/978eb42451f831/cnn.com/images/19
[12:9:18:860]  Fetching http://www.cnn.com/ads/advertiser/barnesandnoble/9906/top1.gif
[12:9:19:140]  Fetching http://www.cnn.com/ads/advertiser/barnesandnoble/9906/top1.gif
[12:9:19:470]  Fetching http://www.cnn.com/ads/advertiser/bellsouth/9904/115x90date.gif
[12:9:19:740]  Fetching http://www.cnn.com/ads/advertiser/freeshop/9908/115beachd.gif
[12:9:20:10]   Fetching http://w3.ibm.com/images/thinkblot_home.jpg
[12:9:20:340]  Fetching http://www.cnn.com/ads/advertiser/cnn/9909/CNN_Home_Button.gif
[12:9:20:620]  Fetching http://www.ibm.com/i/v9/icons/ra_kb.gif
[12:9:20:890]  Fetching http://w3.ibm.com/images/buttons/btn_ticker.gif
[12:9:21:170]  Fetching http://w3.ibm.com/news/features/images/announce2_0917.gif
[12:9:21:500]  Fetching http://www.cnn.com/chat/images/chat.news.gif
[12:9:21:770]  Fetching http://www.cnn.com/chat/images/chat.cnn.gif
[12:9:22:50]   Fetching http://w3.ibm.com/images/buttons/lou_button.gif
[12:9:22:320]  Fetching http://www.cnn.com/images/9808/icons/audio_icon.gif
[12:9:22:650]  Fetching http://www.cnn.com/images/9808/icons/audio_icon.gif
[12:9:22:930]  Fetching http://w3.ibm.com/images/buttons/lou_message.gif
[12:9:23:200]  Fetching http://w3.ibm.com/images/buttons/lou_message.gif
[12:9:23:470]  Fetching http://w3.ibm.com/images/buttons/lvg_hear_small.gif
[12:9:23:750]  Fetching http://www.cnn.com/images/9808/icons/video_icon.gif
[12:9:24:80]   Fetching http://www.ibm.com/i/1999/09/main_lean.gif
[12:9:24:350]  Fetching http://www.ibm.com/i/v9/f/en/privacy.gif
[12:9:24:630]  Fetching http://w3.ibm.com/images/footer/w3.gif
[12:9:24:900]  Fetching http://w3.ibm.com/images/footer/w3.gif
[12:9:25:230]  Fetching http://www.cnn.com/images/1999/09/up.gif
```

```
[12:9:25:510]  Fetching
               http://www.cnn.com/SHOWBIZ/Music/9909/17/wb.supergroup/thumbnail.jpg
[12:9:25:780]  Fetching
               http://a388.g.akamaitech.net/7/388/21/1a7aaa88c4da10/cnn.com/images/19
[12:9:26:110]  Fetching
               http://www.cnn.com/resources/potd/1999/09/20/tz.miss.amercia.pig.ap.gi
[12:9:26:390]  Fetching http://www.cnn.com/images/lotd/dollarsign.gif
[12:9:26:660]  Fetching http://www.cnn.com/FOOD/images/tease.bread.gif
[12:9:26:930]  Fetching http://www.cnn.com/FOOD/images/tease.bread.gif
[12:9:27:260]  Fetching http://www.cnn.com/video/world/1999/09/19/pope.gif
[12:9:27:540]  Fetching
               http://a196.g.akamaitech.net/7/196/21/000/cnn.com/images/1998/11/point
[12:9:27:810]  Fetching
               http://a196.g.akamaitech.net/7/196/21/000/cnn.com/images/1998/11/point
[12:9:28:90]   Fetching
               http://a196.g.akamaitech.net/7/196/21/000/cnn.com/images/1998/11/point
[12:9:28:420]  Fetching
               http://a196.g.akamaitech.net/7/196/21/000/cnn.com/images/1998/11/point
[12:9:28:690]  Fetching
               http://a196.g.akamaitech.net/7/196/21/000/cnn.com/images/1998/11/point
[12:9:46:160]  Fetching http://dmobile.watson.ibm.com/ec/_themes/sumipntg/sumbul1d.gif
[12:9:46:430]  Fetching http://dmobile.watson.ibm.com/ec/_themes/sumipntg/sumbul1d.gif
[12:9:46:710]  Fetching http://dmobile.watson.ibm.com/ec/_themes/sumipntg/sumbul1d.gif
[12:9:47:40]   Fetching http://dmobile.watson.ibm.com/ec/_themes/sumipntg/sumbul1d.gif
[12:10:17:910] Web Page Synchronization Complete.
[12:10:18:20]  Idle
********************************
Log file closed 09/20/99 12:10:35
********************************
```

Figure 4-6. Sample Connection Manager log file.

Changing the current location

To select a new location as the current location, the user first opens the Location Manager by double clicking on the Location Manager icon on the desktop.

The user then locates the location to select as the new default location. The mouse cursor is placed over the location icon and the users clicks the right mouse button once. When the menu appears, the user selects *Set as Current Location*. A small check mark appears next to the location icon indicating that this location is the new default location. (See Figure 4-7. Chicago LAN selected as

the current location.). If the selection requires a reboot, the users is prompted and asked if they want to reboot now, reboot later, or shut down.

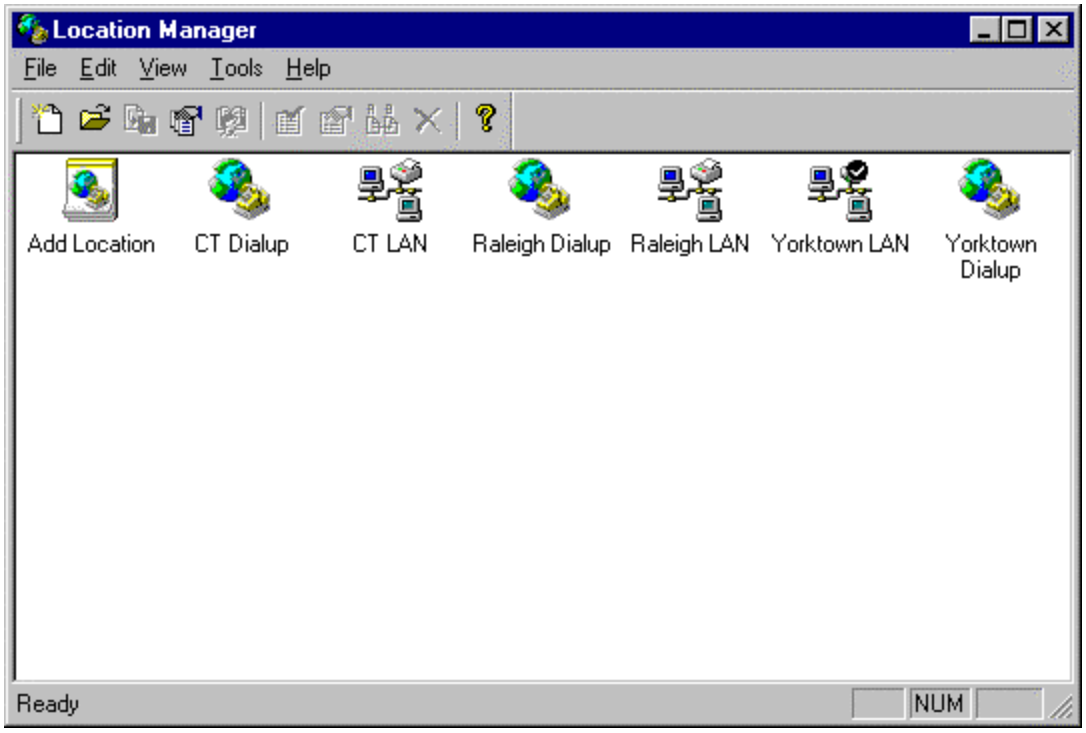

Figure 4-7. Chicago LAN selected as the current location.

Importing a location or locations

To import a new location or set of locations from a location configuration (.LCF) file, the user starts the Location Manager by double-clicking on the Location Manager icon on the desktop.

From the Location Manager menu, the users selects *File ->Import* or clicks the import icon in the menu bar. The user then highlights the location file that they wish to import. The location configuration file always has an extension of .lcf. If the location file does not appear in the file locate dialog, the user can browse the list of LCF files that are available using the *Browse* button. When the user has located the file they wish to import, they highlight the file, click *Open*, and the new locations will be added to the Location Manager. If the user does not wish to add the locations to your current configuration, they can use the *Cancel* button to abort the import operation.

To import more than one location file at the same time, the user highlights the location files while holding down the Shift key, and then clicks *Open*.

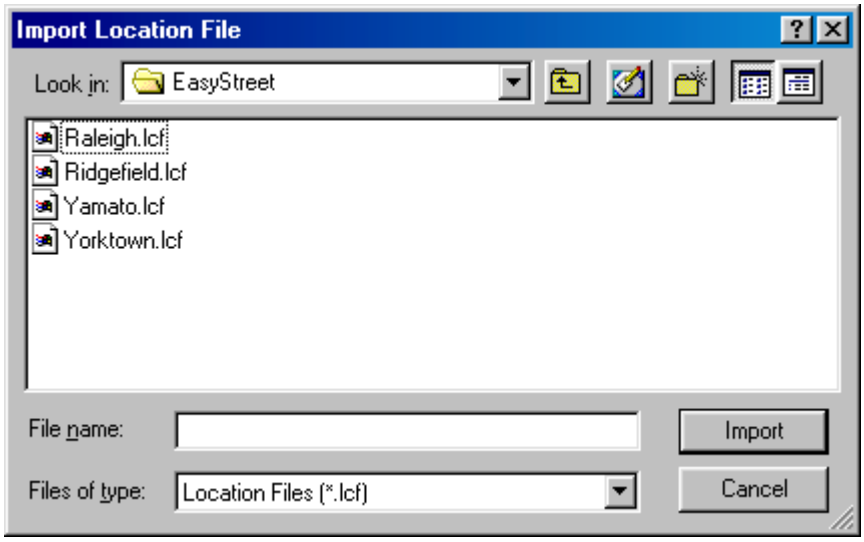

Figure 4-8. Import location file dialog.

Cloning an existing location

To clone an existing location, the user first opens the Location Manager.

From the Location Manager menu, the user selects *File->Add Location* from the menu or click the file icon in the menu bar. The *Add New Location* dialog box appears and gives the user a choice of importing location data from a file, cloning an existing location or creating a completely new location.

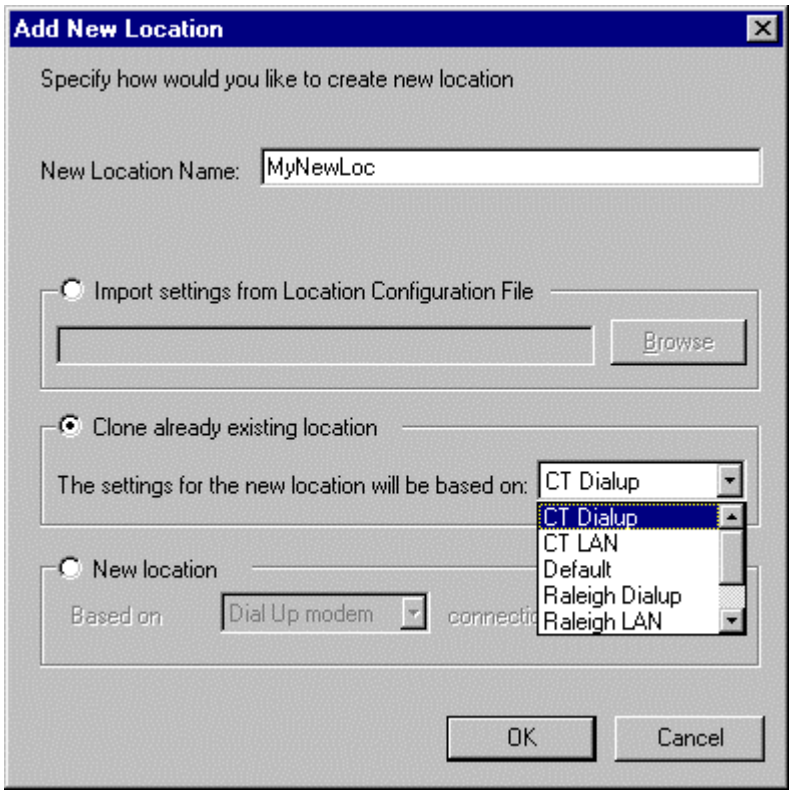

Figure 4-9. Cloning a location.

The user selects *Clone already existing location*. The dialog prompts the user to select the location to clone from the list of current locations. From the list, the user selects the location to clone and clicks *OK* to continue the operation or *Cancel* to abort the operation. If the user clicks *OK*, the system creates the new location using the name that was entered in the Location Name field. The location can be renamed using *Rename* function. Cloned locations are populated with the properties of the cloned location.

Creating a new location

To create a blank location, the user first opens the Location Manager.

From the Location Manager menu, the user selects *File->Add Location* from the menu or click the file icon in the menu bar. The *Add New Location* dialog box appears, giving the user the choice of importing location data from a file, cloning an existing location or creating a completely new location.

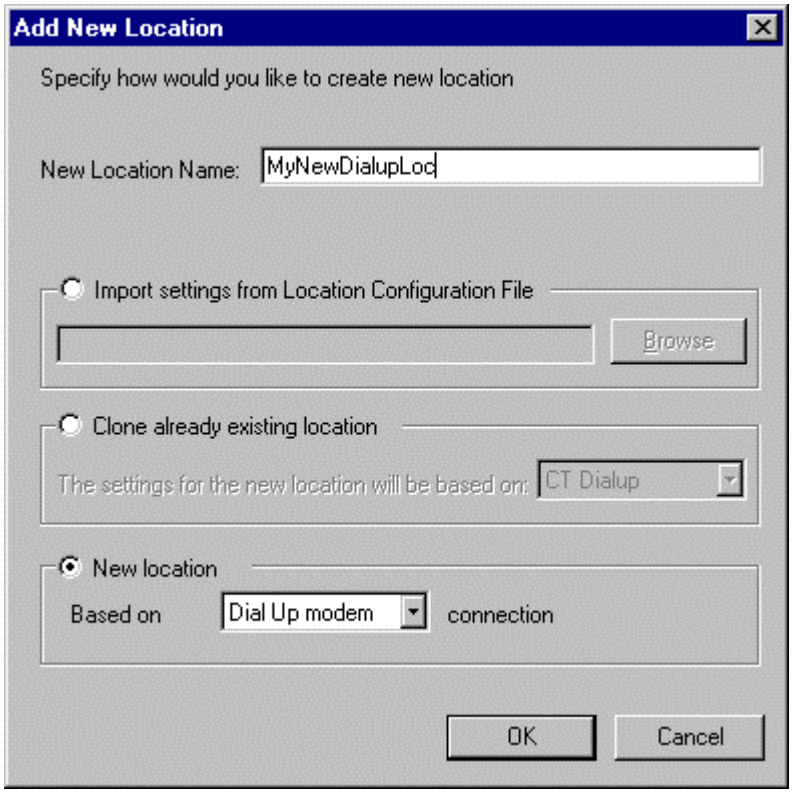

Figure 4-10. Creating a new location.

The user selects the *New Location* radio button and enters the name of the new location in the *New Location Name* field. Then, using the *Based on* selection box, the user selects the type of location to be created. Currently, only LAN and Dialup locations are supported. EasyStreet does not allow for the creation of a completely blank location because using the bank location as the current location would render the system unbootable.

Deleting a Location or Locations

To delete an existing location, the user first opens the Location Manager.

The location to delete is selected by highlighting it with a single click of the left mouse button or by placing the mouse cursor over the top of the location icon. The user clicks the right mouse button and selects *Delete* from the menu by a single click of the left mouse button. A dialog appears to confirm the choice to delete the location. Clicking *No* aborts the deletion and clicking *Yes* deletes the selected location. Once deleted, a location cannot be recovered unless a backup of the location database exists.

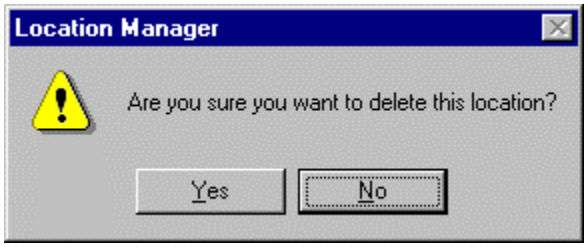

Figure 4-11. Location delete confirmation dialog.

Multiple locations can be deleted at the same time by highlighting (selecting) more than one location before selecting *Delete*. To select multiple locations, the user presses and holds the *Shift* key while selecting the location icons to delete.

When the selection is completed, the user places the mouse cursor over any one of the selected icons and clicks the right mouse button once. From the popup menu, the user clicks *Delete* and is prompted to confirm the delete operation by clicking *OK* to delete the locations or *Cancel* to abort the deletion.

Exporting an existing location or group of locations

To export an existing location or group of locations, the user first opens the Location Manager.

The location or locations to export are selected by highlighting (selecting) the locations. From the *File* menu, the user selects *Export,* enters the location and file name of the new exported location file and clicks *OK*. It is not necessary to enter the LCF extension in the file name, as the system will append it automatically. The new location file will be created with the .lcf extension, and can now be sent via email or uploaded to a Web site to allow other users to use it.

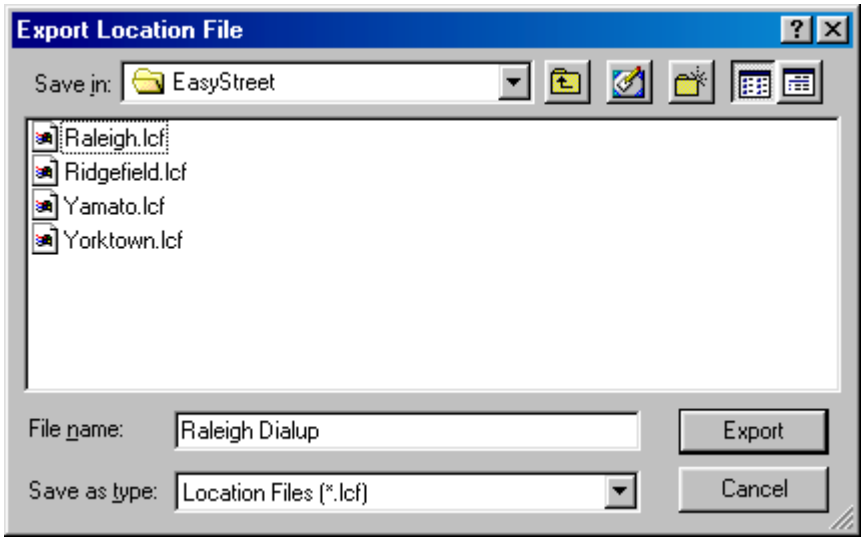

Figure 4-12. Exporting Location Information

The user can also export multiple locations by highlighting (selecting) more than one location before clicking *OK*. To select multiple locations, the user presses and holds the shift key while selecting the location icons to export. When the user is done selecting the icons, they select *File ->Export* from the menu and enter a file name to be used for the exported location information. It is not necessary to enter the LCF extension in the file name, as the system will append it automatically. When ready to export the locations, the user clicks the *OK* button. The location data is written out to a location file that can now be shared with other users or placed on a Web site for downloading.

Propagating Location Parameters

In some cases, the user might want to propagate some location settings to other locations. Rather than having to change the machine name parameter for every configured location, the Location Manager allows the user propagate the changes made in one location to other selected locations.

To propagate settings to another location, the user highlights the data to propagate. In the case of multiple items such as Web pages or files, multiple items can be selected by holding the *Shift* key down while highlighting the data items to propagate. This is the standard Windows convention for selecting multiple objects. When all of the data items have been selected, the user clicks the right mouse button and selects *Propagate*. A dialog appears with the names of the current locations. To propagate the current data to a single location, the user highlights that location and clicks *OK*. To propagate the data to multiple locations, the locations are selected by highlighting each one while holding down the *Shift* key. The user then clicks the *OK* button to propagate the data.

Backing Up and Restoring the EasyStreet Configuration

EasyStreet provides functions to save and restore configuration using the *Backup* and *Restore* menu options in the EasyStreet main window. The EasyStreet configuration should be backed up whenever a change is made. In the event of a disk crash, the EasyStreet configuration can quickly be restored. If the user has more than one system, the backup can also be used to reload a second machine with the same configuration. The Backup and Restore functions save and restore the entire EasyStreet configuration including location information and personal settings.

To backup the EasyStreet configuration, the user first opens the Location Manager.

From the Location Manager menu, the user selects *File->Backup*. A dialog appears and prompts the user for a file name for the backup file. The file extension for an EasyStreet backup file is always .ECB. The user enters the file name of the backup file or selects a backup file to overwrite, then clicks the *OK* button. It is not necessary to enter the ECB extension in the file name, as the system will append it automatically.

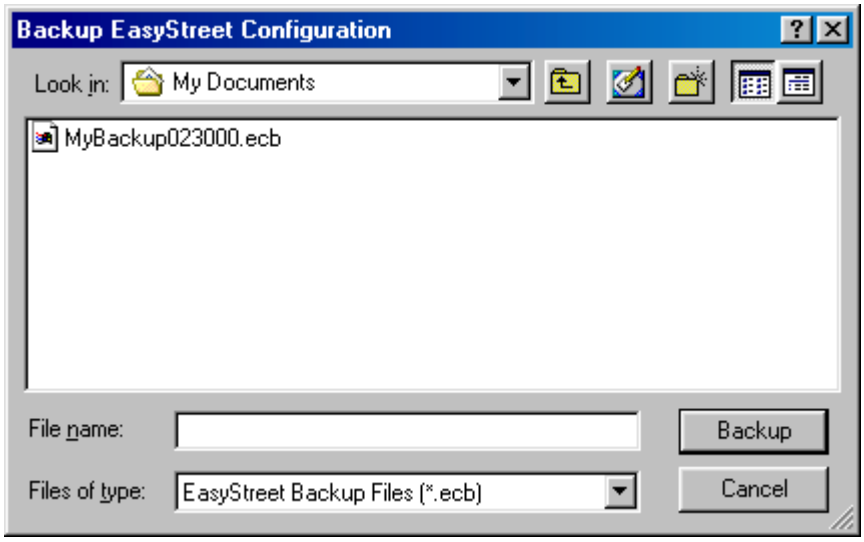

Figure 4-13. Backing up the EasyStreet configuration.

To restore the EasyStreet configuration, the user first opens the Location Manager.

From the Location Manager menu, the user selects *File->Restore*. A dialog appears and prompts the user to enter the file name for the restore operation. The extension of the EasyStreet backup file is always .ECB. The user enters the file name of the file to be used to restore the configuration and clicks *OK*. The settings are then restored from the configuration file.

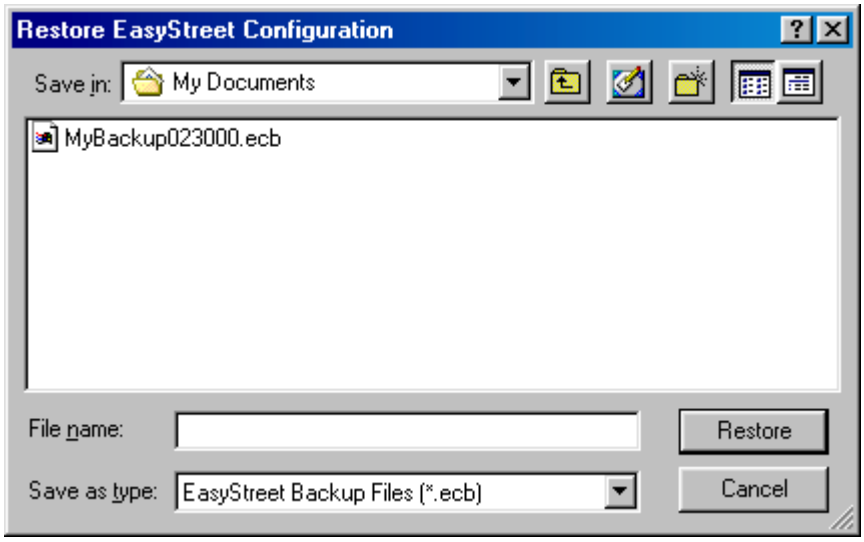

Figure 4-14. Restoring the EasyStreet configuration.

Viewing and Editing Location Parameters

Location-specific parameters are kept in the location database in the Windows registry. To access the parameters for a particular location, the user must first select a location and then open the properties for that location. EasyStreet provides an administrative software package to help a support organization create and maintain the location files.

Normally, the EasyStreet configuration information is preloaded by a company's support organization. The user simply imports the location data from the specified file and the EasyStreet configuration is almost complete. For most users, the configuration supplied with their EasyStreet installation will be

sufficient. However, some users may elect to change some of the location-specific parameters to reflect their personal needs.

EasyStreet allows the user to customize file synchronization, mail delivery, and Web page caching on a per-location basis. Using the Location Manager user interface, the user can browse their settings and make changes to a variety of settings depending on their needs. Although there many settings that can be changed, users need not be afraid to look through the various categories to gain an understanding of the kinds of parameters that can be changed. If the user makes a mistake or is unsure of a change, the *Cancel* button can be pressed to ignore any changes that may have made. The user can also delete the location and recreate it using the import, create or restore functions.

User Interface

The EasyStreet location parameters are viewed and edited using a standard Windows user interface component called the Property Sheet. The Property Sheet is a collection of tabular pages called Property Pages. All of the parameters in the Property Sheet are saved using the *Apply* button at the bottom of the Property Sheet. Clicking *Apply* applies the all of the changes to the current Property Sheet without exiting the configuration procedure. Clicking on OK will

cause the all of the parameters in the Property Sheet to be saved and the Properties window to be closed.

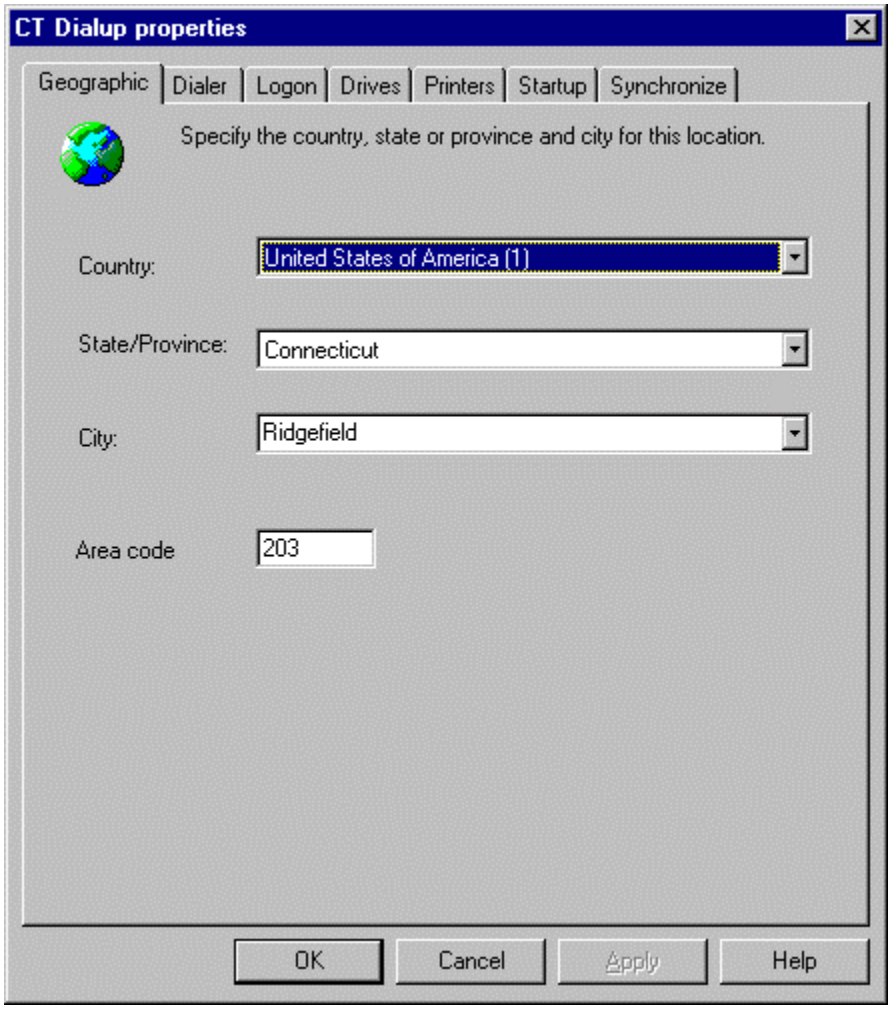

Figure 4-15. Geographic settings dialog.

Geographic Information

To view or edit the geographic information, the user first opens the Location Manager.

The location to be edited is selected by placing the cursor over the location to edit and clicking the right mouse button to open the configuration dialogs for that location.

The Geographic page is selected by clicking on the Geographic tab in the Properties window. Using the pull-down controls, the user selects the country, state or province, and city or town that this location will be associated with. The country is selected first, followed by the state or province, and finally the city or town. If an appropriate city cannot be located from the database, the user can enter the city in the city field using the keyboard. Finally, if the location is a dialup-type location, the user enters the area code that will be used for this location. The area code will be used later to help narrow down the choices of the dialup access numbers presented for selection.

EasyStreet uses the geographic location to group or sort locations by country, state and city, and to better identify the dialup access numbers to use. This data becomes part of the location data for this location.

When the user has finished editing the geographic information, they can select *OK* to save the changes and close the properties dialog. If the user clicks *Cancel*, the changes are not saved and the properties dialog is closed. If the user clicks on another one of the settings tabs, the changes to the current page are saved temporarily, but are not saved permanently to the location database until the user clicks the *Apply* or *OK* button. Using the *Apply* button allows the user to permanently save the changes on the parameter page before continuing on to the next parameter page but without closing the Properties window.

Dialer Settings

To view or edit the dialer settings, the user first opens the Location Manager.

The location to edit is selected by placing the cursor over the location to edit and clicking the right mouse button to open the configuration dialogs for that location. Note that the Dialer settings will only appear for dialup location types.

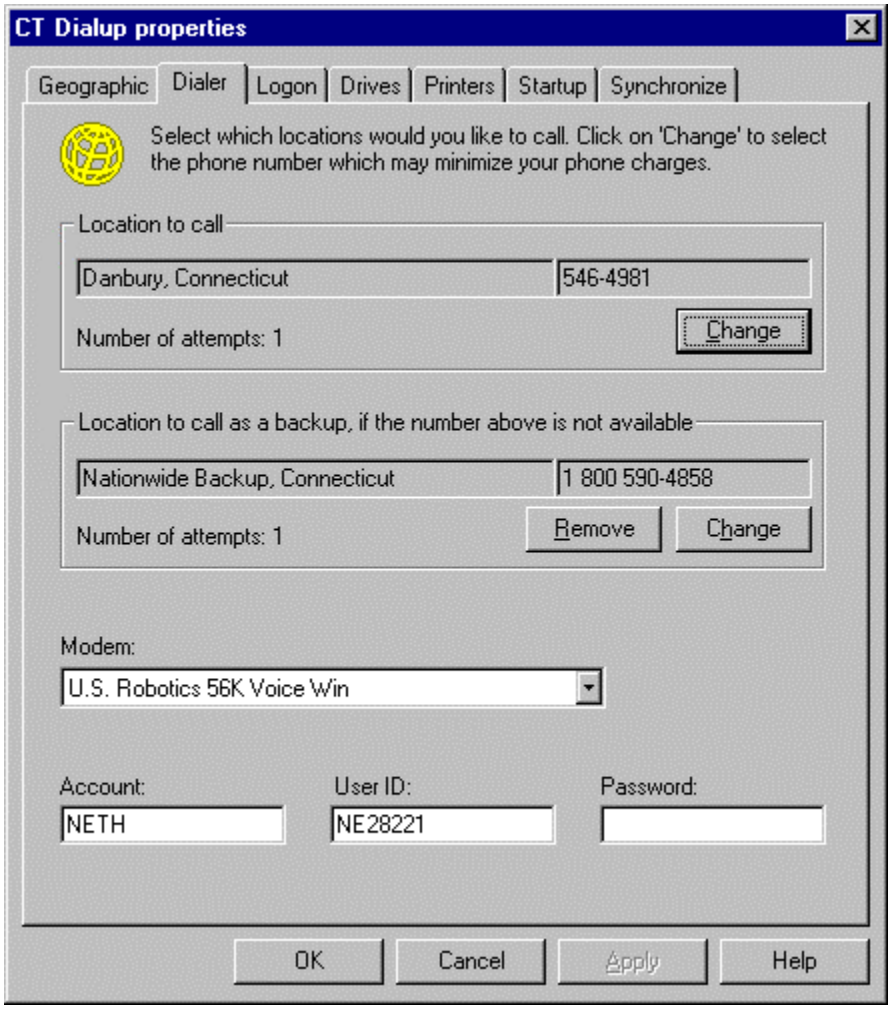

Figure 4-16. Dialer settings dialog.

The *Dialer* page is selected by clicking on the *Dialer* tab in the Properties window. To select or change the primary dialup number, the user clicks the *Change* button next to the primary dialup number field. A dialog appears that allows the user to select the primary dialup number and the number of times they want to retry that number if the call fails. Remember that with some secure dialup systems, trying a number three times with the wrong password could cause the dialup account to be disabled by the system administrator. Once the primary

dialup number has been selected, the user clicks the *OK* button to save their preferences.

To select a backup dialup number, the user clicks the *Change* button next to the backup dialup number field. A dialog appears that allows the user to select the backup dialup number and the number of times they want to retry that number if the call fails. Once the primary dialup number has been selected, the user clicks the *OK* button to save their preferences.

When EasyStreet is started for the first time, a snapshot of the current machine settings is saved in the Default Settings location. Whenever a new location is created or imported, the parameters in the new location are populated with the parameters from the Default Location. The Default Location acts as a template, initializing parameters in the new location to known values. The user can go back at any time and edit the parameters in the Default Settings location. The next time a new location is created, the default values will be copied into the new location. This insures that the critical parameters necessary for normal operation are propagated to all new locations. It also prevents the user from creating a location that might result in an unbootable system.

When a new location or set of locations is imported, the area code of the location and the suggested dialup access phone numbers are read from the location file

and saved in the dialup settings for that location. If a new location is created, the user chooses the country, state and city for that new location. Using this information, EasyStreet attempts to locate the correct dialup access numbers based on the geographic data and saves that data as part of the location. If a match for the country, state and city cannot be found, the user will have to select the dialup number using the dialer dialog.

Logon Parameters

To view or edit the logon parameters, the user first opens the Location Manager.

The location to edit is selected by placing the cursor over the location to edit and clicking the right mouse button to open the configuration dialogs for that location.

The *Logon* page is selected by clicking on the *Logon* tab in the *Properties* window.

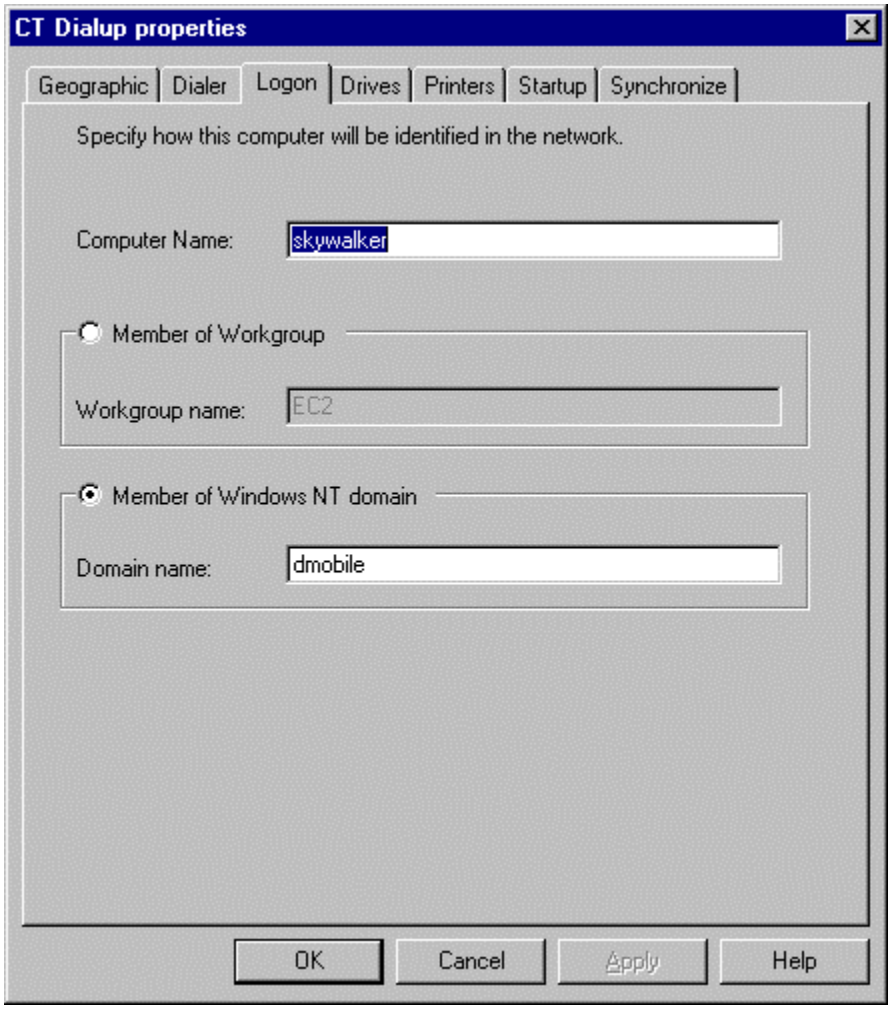

Figure 4-17. Network logon dialog.

The computer name will already be filled in using the current computer name from the Windows registry. The user can change the name of the computer for this location, or accept the name that EasyStreet has filled in from the Windows registry. In most cases, the computer name will not have to be changed. However, there is a remote possibility that if the user attempts to connect to a network at a different location that a computer with the same may already exist and be connected to the network. If this happens, the name of the computer can

be changed using this dialog so it will not conflict with an existing computer name. The name the user selects is valid for this location, and a different computer name can be used for other locations.

If the user is a member of a workgroup, the workgroup name is entered in the Workgroup field. Being a member of a workgroup allows the user to “see” other machines that are members of the same workgroup, and allows for the sharing of files and resources within the workgroup.

If the system needs to log onto a Windows NT domain, the user checks the *Member of Windows NT domain* button and enters the domain name. The user must have the proper user ID and password to log onto the domain, and the machine name must be unique. If the user’s machine is running Windows 9x, choosing a machine name is not critical.

However, if the user’s machine is running Windows NT or Windows 2000, the computer name is critical. On a Windows NT network, the machine name is used to validate the system’s right to log onto the network. When a Windows NT or Windows 2000 machine first logs onto an NT network, the NT domain controller establishes a special relationship with the Windows NT or Windows 2000 system and creates a special identification key that is used for subsequent network logons. If the user’s machine had already been logged on to the Windows NT

network, the Windows NT domain controller checks to make sure that the machine attempting to log onto the network has already established a relationship with the domain controller. If it has, the logon proceeds. If this is the first time the Windows NT machine is attempting to log on to the network, however, the Windows NT system administrator must have previously enabled the machine to join the network. The system administrator enables this security feature using the computer name of the system attempting to join the network. If a machine with a name that is unknown to the domain controller attempts to join the Windows NT network it will be refused. When done, the user clicks *Apply* to save the changes or *OK* to save the changes and close the Properties window.

Drives and Network Shares

To view or edit the drives or network shares, the user first opens the Location Manager.

The location to edit is selected by placing the cursor over the location to edit and clicking the right mouse button to open the configuration dialogs for that location.

The user selects the *Drives* page by clicking on the *Drives* tab in the *Properties* window. A list of mapped network drives configured for this location is displayed in the In the Map network drives section.

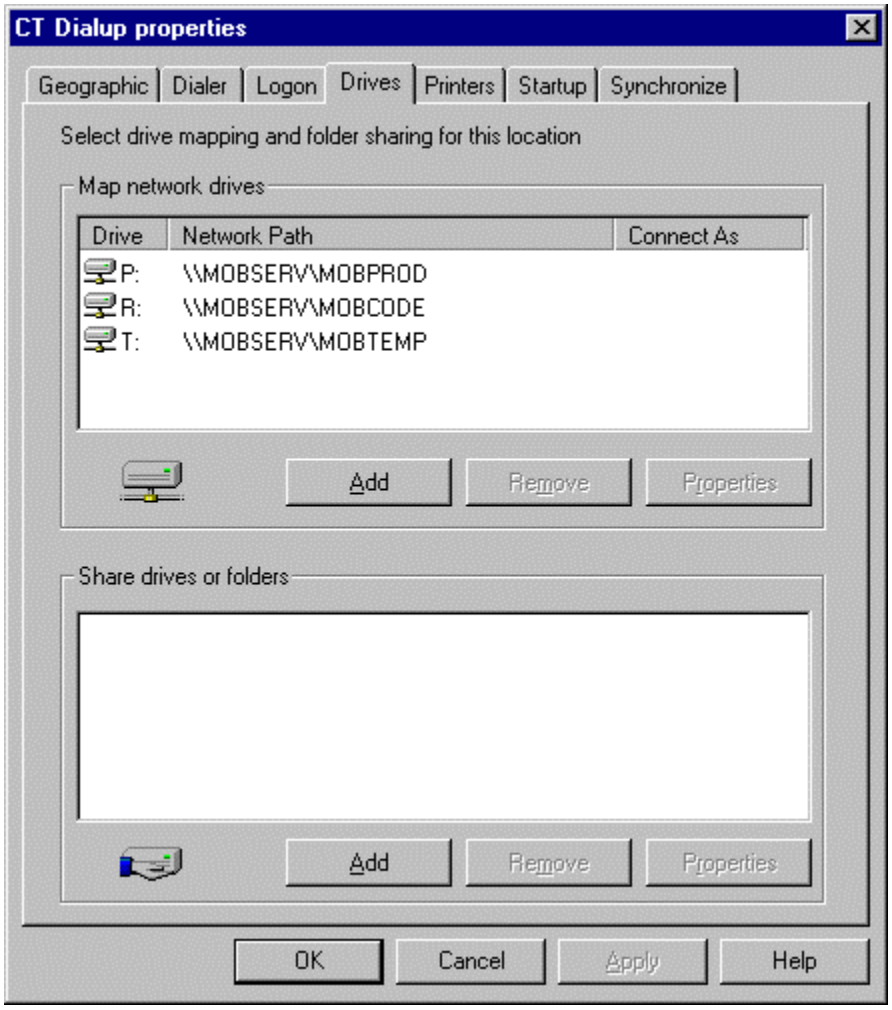

Figure 4-18. Drives and shares dialog.

To remove a currently configured network drive, the user highlights the drive and clicks the Remove button. A dialog prompts the user to confirm the deletion. The user clicks *Yes* to continue with the deletion or *No* to abort the operation.

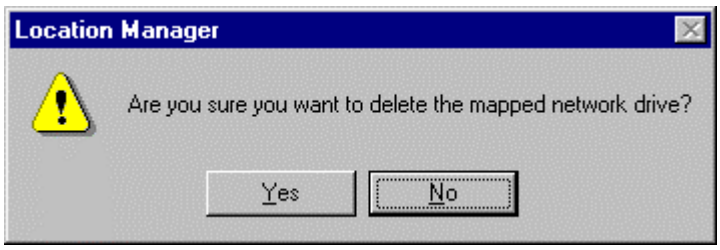

Figure 4-19. Delete mapped network drive confirmation.

To add a mapped drive, the user clicks the *Add* button. In the *Map Network Drive* dialog, the user enters the drive letter to be use for the new mapped drive and enters the location of the mapped drive. The user may also browse for a mapped drive using the *Browse Network* button next to the *Network Path* and selecting the drive to map using the *Browse for Folder* dialog. When the drive to map has been selected, the user clicks *OK* to select it or *Cancel* to abort the selection.

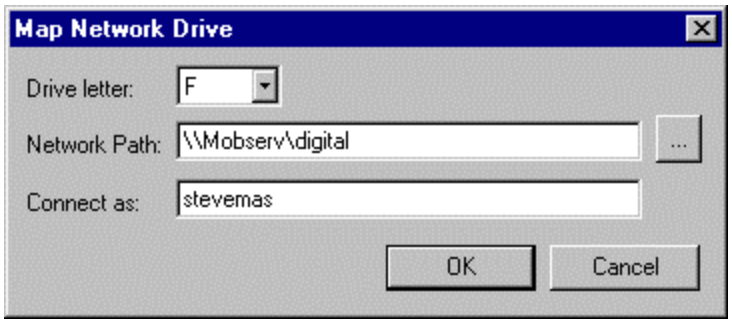

Figure 4-20. Mapping a network drive.

To edit a mapped network drive, the user highlights the mapped drive and clicks the Properties button. The *Map network drive* dialog appears, allowing the user to make the necessary changes to the network drive properties by editing the

data in the fields. When done, the user clicks *OK* to save the changes or *Cancel* to cancel any changes.

To map a drive on a network that requires a user ID and password to access the network, the user will need to enter a valid user ID and password to get access to the mapped network drive. If the user enters a user ID in the *Connect as* field, they will be prompted for a password. If no user ID in entered in this field, the user will not be prompted to enter a password. The user then clicks *Apply* to save the changes for this page or click *OK* to save the changes and close the *Properties* window. If the user clicks the *Cancel* button, no modifications will be saved.

If network shares are defined for this location, they will be listed in the *Shared drives or folders* section of the *Drives* page.

To add a share, the user clicks the *Add* button. In the *Shared drive or folder* dialog, the user enters the path to the share by clicking the *Browse* button next to the *Path* field. In the *Browse for folder* dialog, the folder or share to map is selected and the user clicks *OK* to save or *Cancel* to cancel the selection. Comments can placed in the *Comments* field in the *Shared drive or folder* dialog. The comments have no programmatic function. If the share requires a password for access, the password is entered in the *Password* field in the *Shared drive or*

folder dialog. Finally, the user selects the type of access to be granted, *Read-Only* or *Full,* by selecting the appropriate radio button.

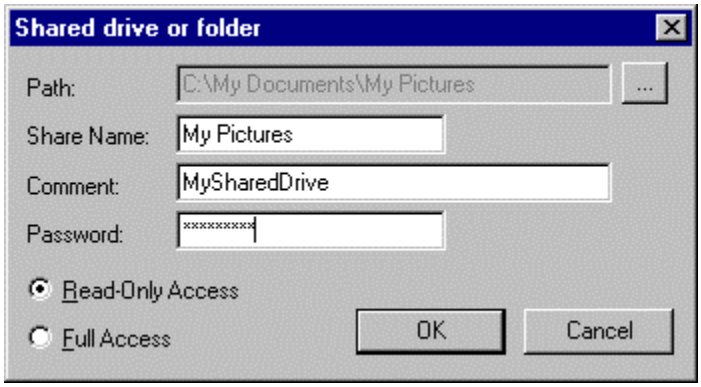

Figure 4-21. Adding a share.

To remove a share, the user highlights the share and clicks *Remove*. The system prompts the user to confirm their choice with confirmation dialog.

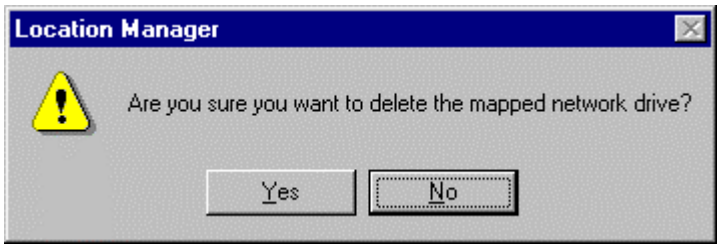

Figure 4-22. Removing a network share.

To edit an existing share, the user highlights the share and clicks the *Properties* button. The *Shared drive or folder* dialog appears, and the user can make the

necessary changes to the share properties by editing the data in the fields. When done, the user clicks *OK* to save the changes or *Cancel* to cancel any changes.

Printers

To view or edit the printer information, the user first opens the Location Manager.

The location to edit is selected by placing the cursor over the location to edit and clicking the right mouse button to open the configuration dialogs for that location. The user selects the *Printers* page by clicking on the *Printers* tab in the *Properties* window.

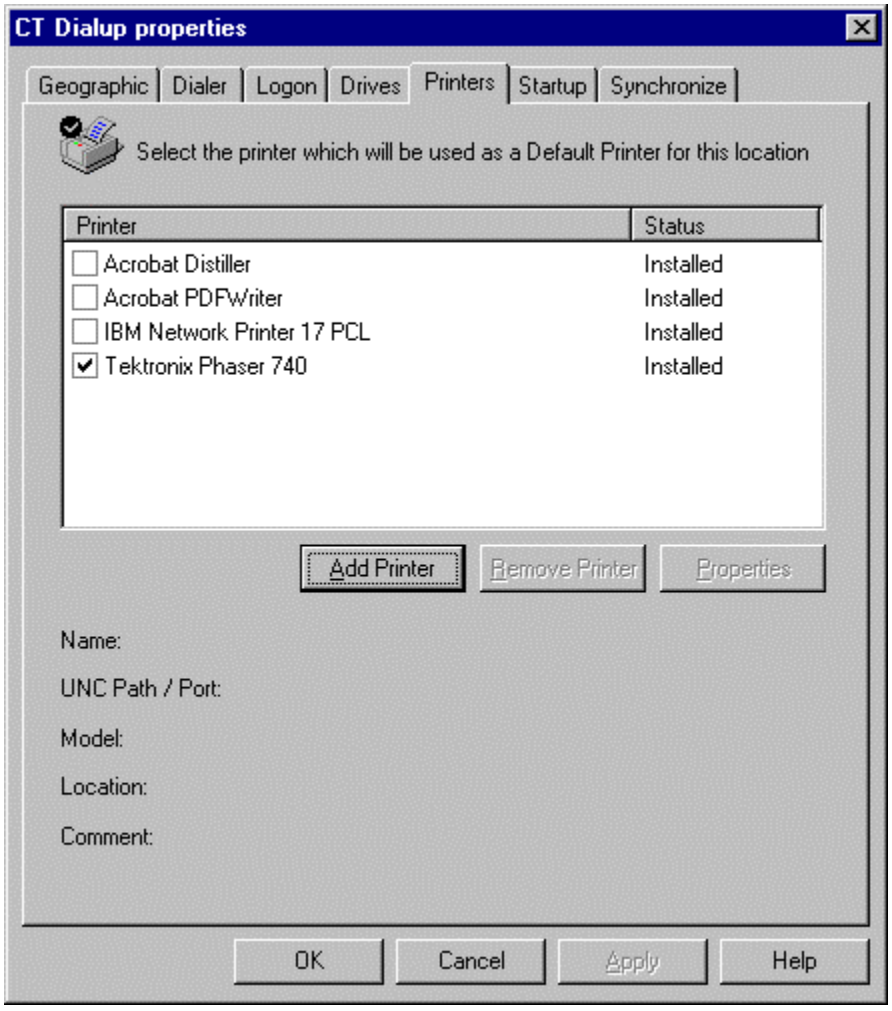

Figure 4-23. Printers dialog.

From the *Printers* page, users can add and remove printers from the current location, or edit the properties of the selected printer.

To add a printer, the user clicks the Add Printer button that will start the *Add Printer Wizard*. The *Add Printer Wizard* adds a printer using the standard Windows *Add Printer* dialogs. These are the same dialogs that are used to add a printer from the Windows control panel.

First, the user specifies whether the printer is a local printer connected to one of the LPT ports or LAN. Next, the printer type is chosen from a list of known printer types, or from a printer description (INF) file a floppy disk or CDROM. When the installer finds the correct driver, the user selects the port that the printer will be installed on and the name to assign to the printer. When the user clicks *OK*, the printer software is installed and will appear in the list of printers for this location. To select this printer to be used as the default printer for this location, the user checks the box next to the printer on the *Printers* page.

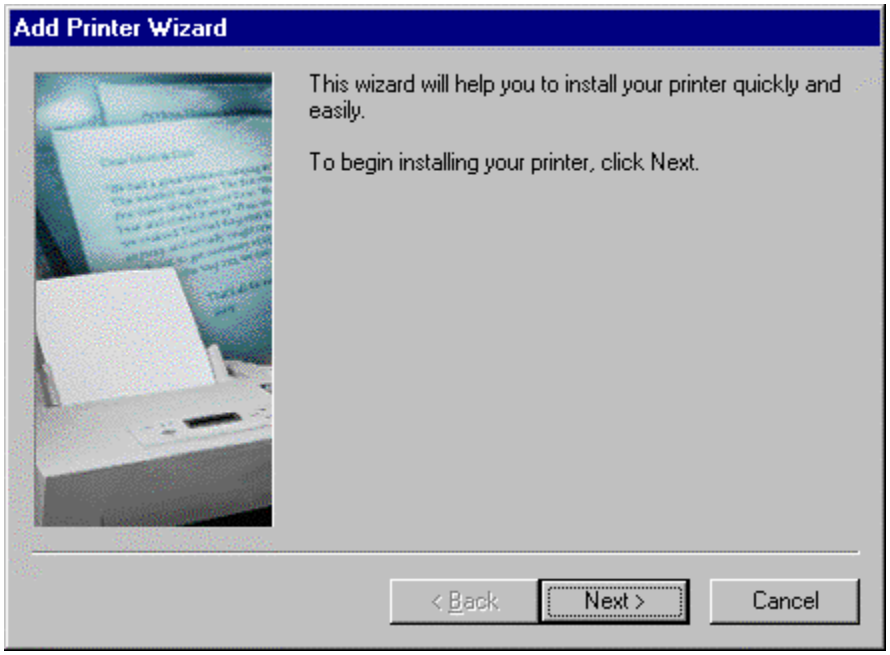

Figure 4-24. Add Printer wizard.

To remove a printer from the list of printers defined for the location, the user highlights the printer and clicks the Remove button. The printer is then removed.

To set the print properties for the printer, the user clicks the *Properties* button.

Startup Configuration

To view or edit the startup information, the user first opens the Location Manager.

The location to edit is selected by placing the cursor over the location to edit and clicking the right mouse button to open the configuration dialogs for that location. The user selects the *Startup* page by clicking on the *Startup* tab in the *Properties* window.

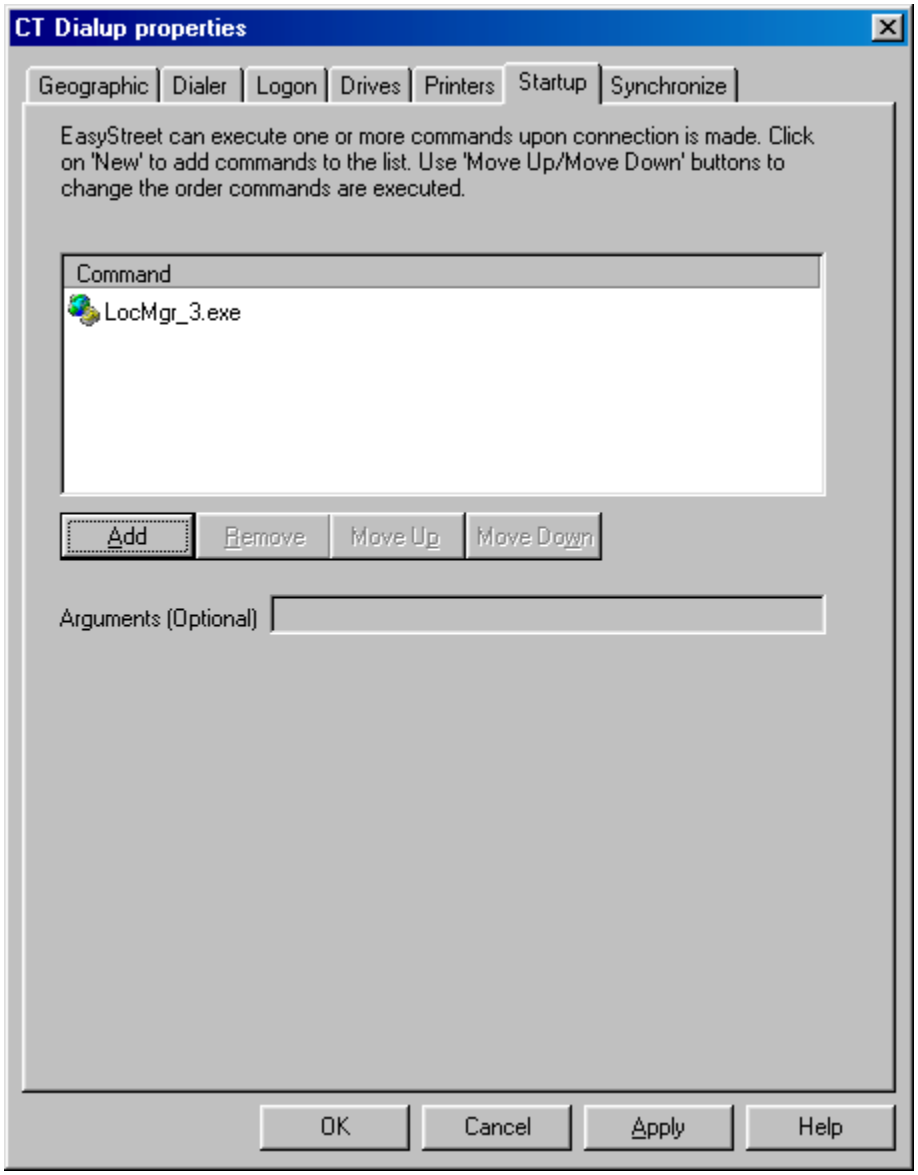

Figure 4-25. Startup settings.

The user clicks the *Add* button to add a new executable program or *bat* file to the existing list of programs to be run at the location. A file dialog appears where the user can enter the path and program name of the executable they wish to run at the new location. When the user clicks *OK*, the program executable or batch file is added to the list of programs to run at the location.

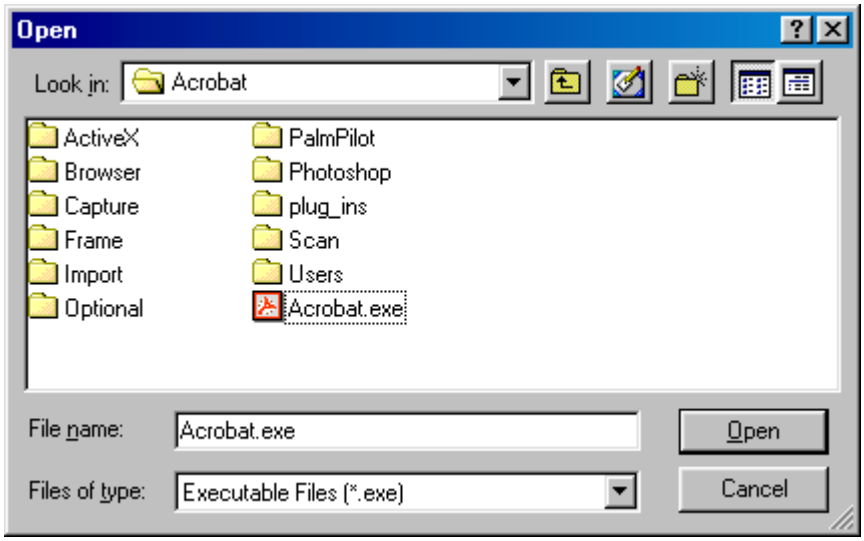

Figure 4-26. Specifying an executable program to start.

The order that programs are started can be changed, or they can be deleted from the startup list. Programs and bat files are started by the system in the order in which they appear. EasyStreet does not guarantee that programs are started in a synchronous fashion. Due to differences in program load and execution time, a program at the end of the list may actually be run before a program at the beginning of the list. There is no provision to insure that the list of applications or bat files are run in the order that they appear in the list.

To remove an executable file from the startup programs list, the user highlights the executable file and clicks the Remove button. A confirmation dialog appears to confirm the decision to delete the program from the startup list.

Figure 4-27. Removing an executable file from the startup list.

TCPIP (LAN-type connections only)

To view or edit the TCPIP settings, the user first opens the Location Manager.

The user selects the location to edit by placing the cursor over the location to edit and clicking the right mouse button to open the configuration dialogs for that location.

The user selects the *TCPIP* page by clicking on the *TCPIP* tab in the *Properties* window.

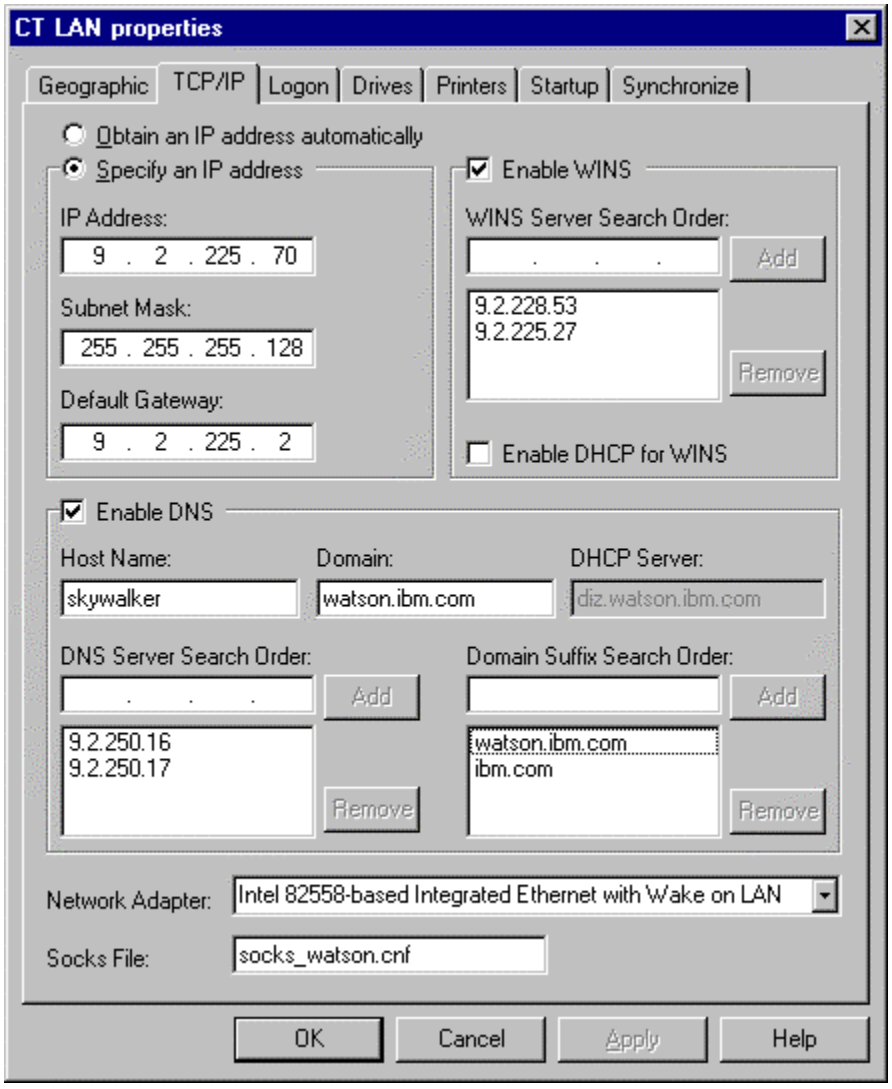

Figure 4-28. TCPIP settings.

The TCPIP dialog contains parameters that are set by the system administrator, and defines how the user's system will access internal or external networks. This information does not change often, so users should not normally have to edit any parameters in this dialog. Changing one or more of the parameters on this page incorrectly may cause the system to not be able to log on to a company's internal or external network. Before changing these parameters, users should review

their changes with their company's support organization. If the user has already changed some of the parameters but is unsure if they have been set correctly and have not clicked *Apply* or *OK* yet, the user can click the Cancel button to discard the changes. If the location the user is editing is the current location and the user has already clicked Apply or OK but doesn't want to use the modified location, they can simply set the current location to another location and reboot. The questionable location can then be deleted and then recreated us*ing the Import* or *Create* functions from the Location Manager *File->Add Location* menu.

If the network supports Dynamic Host Configuration Protocol, the user should select the radio button labeled *Obtain an IP address automatically*. If this option is selected, it means a server on the network will supply the IP address of the machine automatically. The name of the DHCP server is listed in the DHCP Server field in the TCPIP dialog. If DHCP is not enabled, the user must supply a fixed IP address, net mask, primary and secondary name server address and any other TCPIP-related data that is required for the particular network, such as WINS server addresses and domain suffix information.

In most cases, this information is preloaded on the system and the user will not have to make any changes to the TCPIP settings. The user should be sure to check with the network administrator before making changes the TCPIP settings. Should the user make a mistake and is unable to reconnect to the network after rebooting, the user can restart the system and select a known working location

from the EasyStreet boot dialog. This will allow the system to reconnect to the network. Once connected, the user can go back and delete or change the parameters in the offending location, or delete it and start configuring a new location.

If the system has a fixed IP address, the address of your primary and secondary domain name servers should be entered in the appropriate field. Only one name server is necessary, as the secondary name server is used if the first name server does not respond.

The network may have a Windows Internet Name Services (WINS) server that resolves NetBIOS names to IP addresses. A system that uses WINS broadcasts a NetBIOS name on the network and the WINS server responds with the IP address of the name. The user should contact the network administrator to find out if the system needs to specify a WINS server address.

The domain suffix information should be entered in the appropriate field. The domain suffix information is optional and specific for each network. The domain suffix allows the user to specify an abbreviated or unqualified name for a network resource. For example, when configuring a mail program, the user might wish to specify the mail server name as simply *mail*. The system concatenates the domain suffix to *mail* and the entire address is used to access the mail server.

The socks client software allows users behind a corporate firewall to gain access to the Internet. Most corporate users are connected to an internal network called an intranet because it operates within the confines of the company's network infrastructure. To prevent outside users from accessing the intranet, a *firewall* is placed between the intranet and the Internet connection. The firewall allows data from an intranet to be sent to the Internet but prevents unauthorized outside users from accessing the intranet. The socks client allows the user's data to be sent directly to the Internet through the firewall and allows data that the user has requested from the Internet to be sent back to the user's machine through the firewall. The socks configuration file specifies what IP addresses get routed through the firewall. The network administrator normally provides the socks configuration file.

Synchronization

To view or edit the synchronization settings, the user first opens the Location Manager.

The location to edit is selected by placing the cursor over the location to edit and clicking the right mouse button to open the configuration dialogs for that location. The user selects the *Synchronize* page by clicking on the *Synchronize* tab in the *Properties* window.

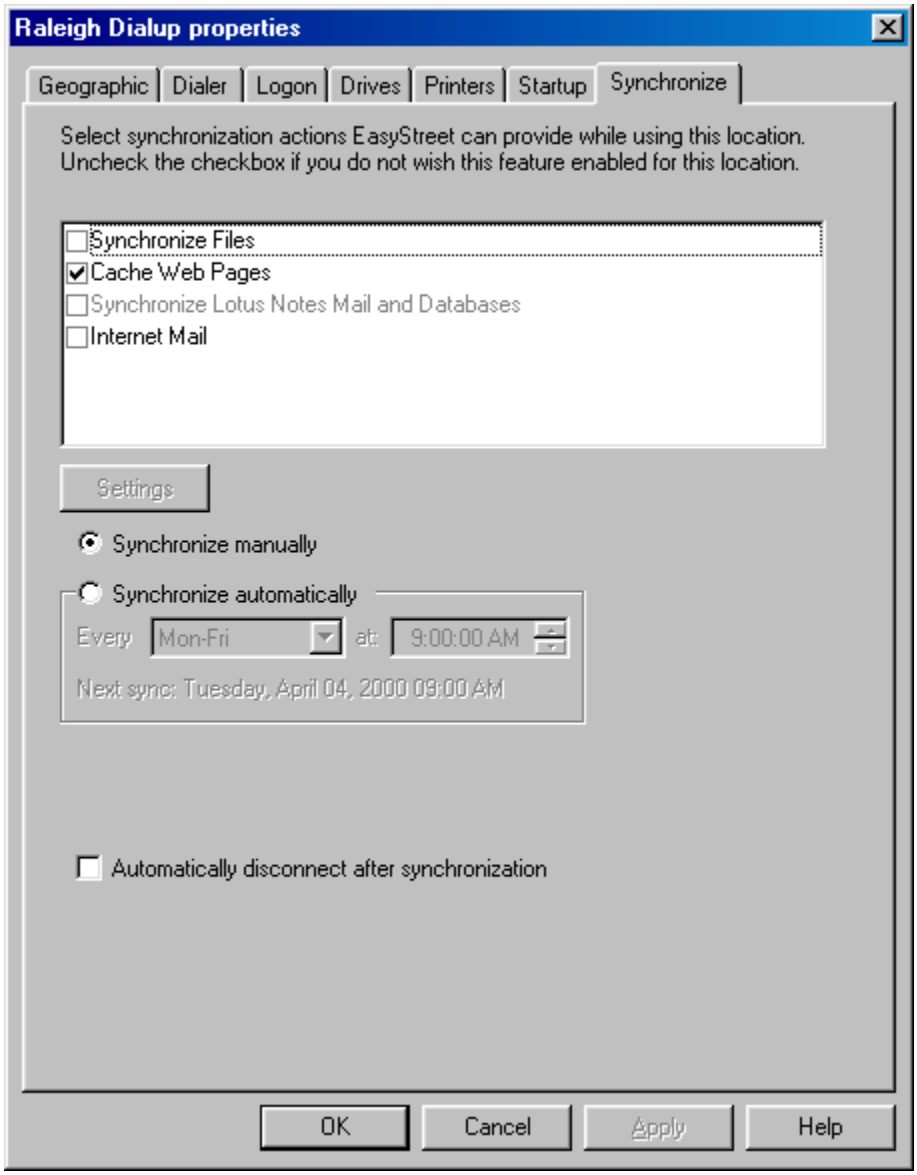

Figure 4-29. Synchronization settings.

The user can enable or disable any of the synchronization tasks using the checkbox next to the task. If the box is unchecked, the specified operation will not be performed. In this dialog, the user can also specify the way the data should be synchronized. If the user selects *Synchronize manually*, synchronization will only be performed when the user clicks the Synchronize

button on the Connection Manager status dialog. The user can specify a variety of days, times, and intervals at which to perform automatic synchronization. If the connection is a Dialup-type connection, this dialog will also contain a checkbox to specify whether or not to disconnect after synchronization. This allows the system dial the phone, synchronize, and hang up the phone at each automatic synchronization interval.

To set up file synchronization, the user highlights the Synchronize Files task and clicks the Settings button.

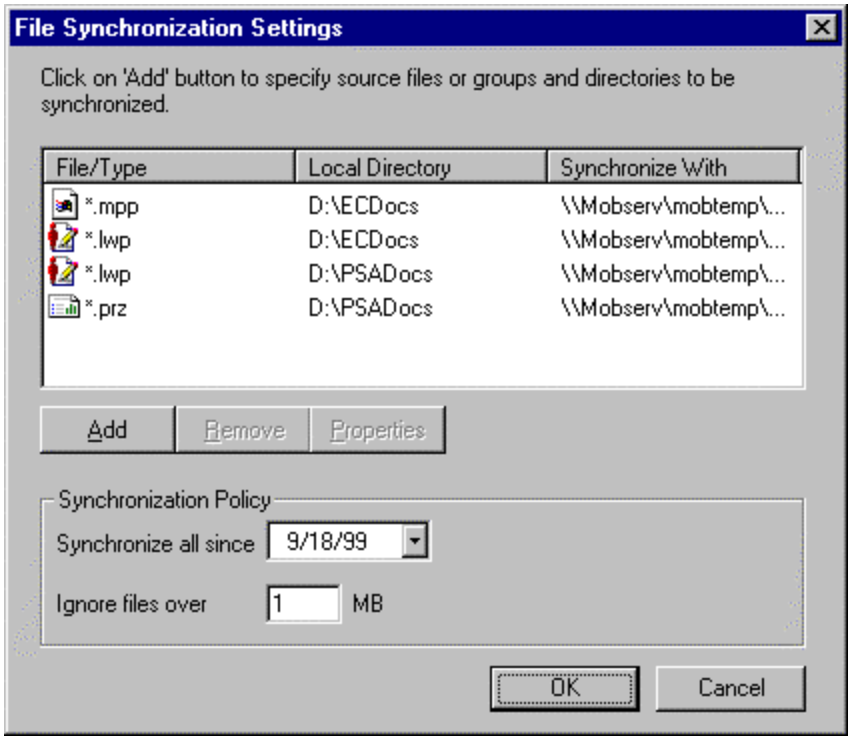

Figure 4-30. File synchronization settings.

The File Synchronization Settings dialog contains a list of the files that is currently specified for replication, along with the source and destination directories for each file. This dialog also contains the Synchronization Policy settings. These settings can be used to further define the file synchronization criteria by specifying a date and file size limitation.

To add files to the list, the user clicks the Add button.

EasyStreet identifies the files to be synchronized using the file type extension. For example, the file extension for Freelance files is PRZ, and for Microsoft Word files the extension is DOC. The File/Type field contains a list of the currently configured file extensions. If the user clicks on the arrow button in the File/Type field, a small drop-down menu appears with the currently configured list of extensions. The user can select an extension, or enter a new extension type in the File/Type field.

If the source directory is known, it can be entered in the Source Directory field. The user can also use the browse button (the small square button with three dots on the bottom) to browse the drives to locate and select the files of that type. The extension that is specified acts as a filter, and only the files with the specified extension are listed. To display all files, the user selects the *.* type extension. If

the destination drive requires a user ID and password to access the drive, the user enters those values in the appropriate field and then clicks OK to accept the changes or Cancel discard them.

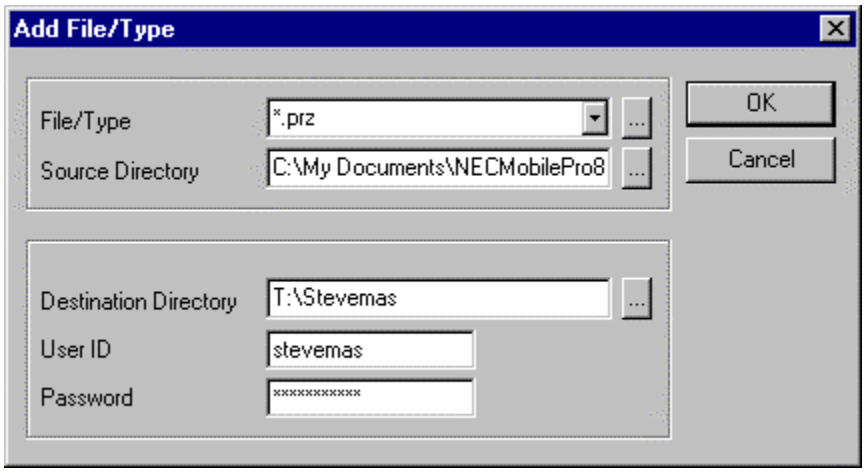

Figure 4-31. File/Type and directories selected.

To remove a file from the list of files to be synchronized, the user highlights the file and clicks the Remove button. A dialog appears to confirm the user's choice to delete the file replication entry. If the user clicks *Yes*, the entry is removed. If the user clicks *No*, no action is taken.

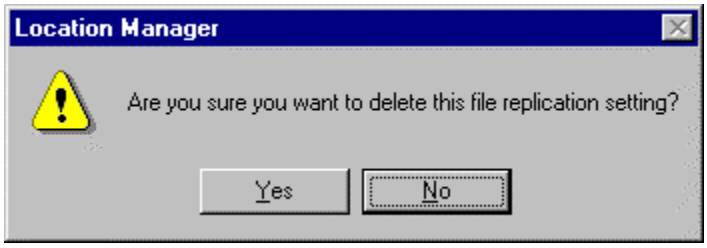

Figure 4-32. Delete confirmation for file replication settings.

To edit an existing file replication setting, the user highlights the file replication entry and clicks the Properties button. The Add File/Type dialog appears with the settings filled in for that specific file entry. The user edits the settings and clicks *OK* to save the changes or *Cancel* to ignore the changes and leave the entry as it was.

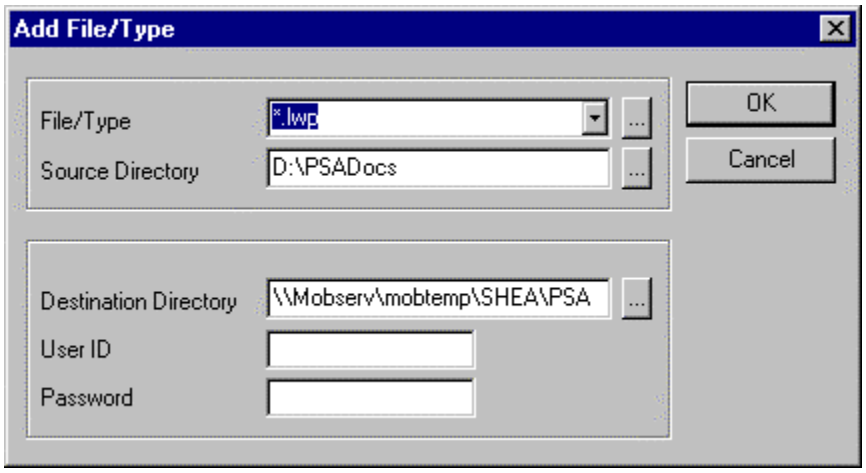

Figure 4-33. Editing an existing file replication setting.

To set up the Web page cache, the user highlights the Cache Web Pages task in the Synchronize page and clicks the Settings button.

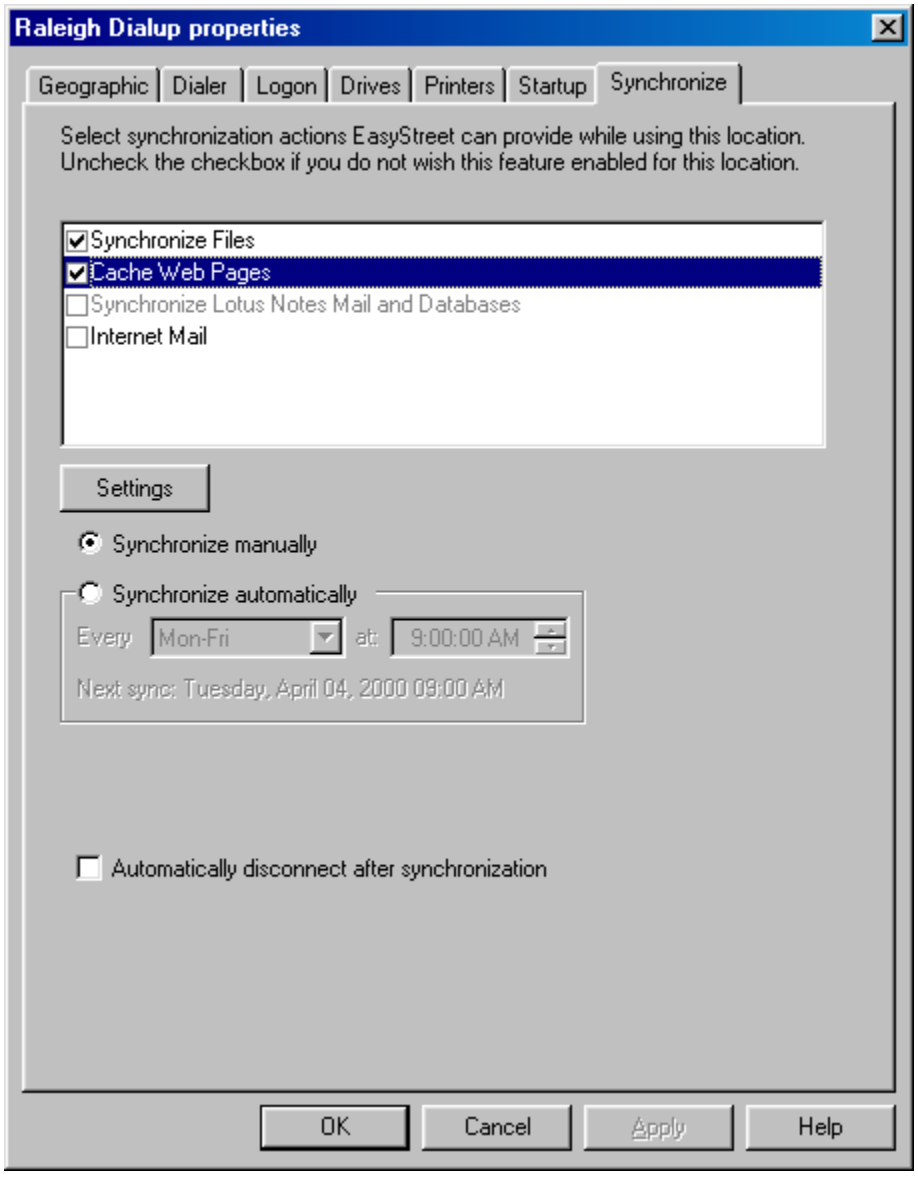

Figure 4-34. Selecting Web cache settings.

A list of Web pages to be retrieved when synchronization is performed is displayed in the Web Cache Settings dialog. The selected pages are retrieved from the specified Web sites and cached (stored) locally on the system's hard drive.

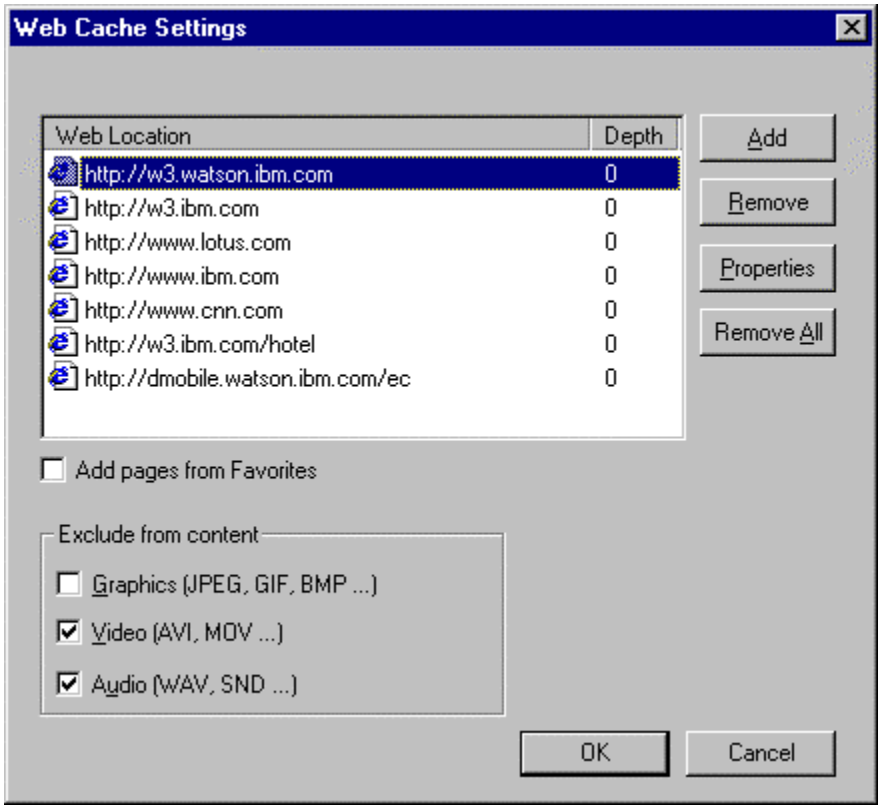

Figure 4-35. Selecting a URL from the list.

In this dialog, the user can add, remove, or edit the settings for Web page synchronization.

If the user will be connecting at this location using a phone line or wireless connection, they may wish to avoid having extra Web content such as images, streaming audio and video, or music files downloaded to the system at synchronization time. The checkboxes in the Web Cache Settings dialog are used to set these preferences on a per-location basis.

To quickly add the URLs from the browser's Favorites settings, the user can check the Add pages from Favorites box and the URLs from the user's Favorites settings will be imported into the list of Web pages to be cached at synchronization time.

To remove a URL, the user highlights the URL and clicks the Remove button.

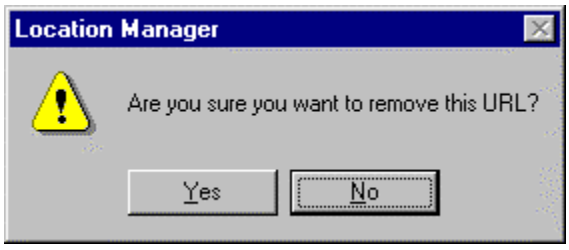

Figure 4-36. Removing a URL from the list.

To edit the settings for a particular URL, the user highlights the URL and clicks the Properties button. The Specify Web Location dialog appears. In that dialog, the user can enter the address of the page or pages to cache, as well as the update frequency and the depth of the links to cache, relative to that specific page.

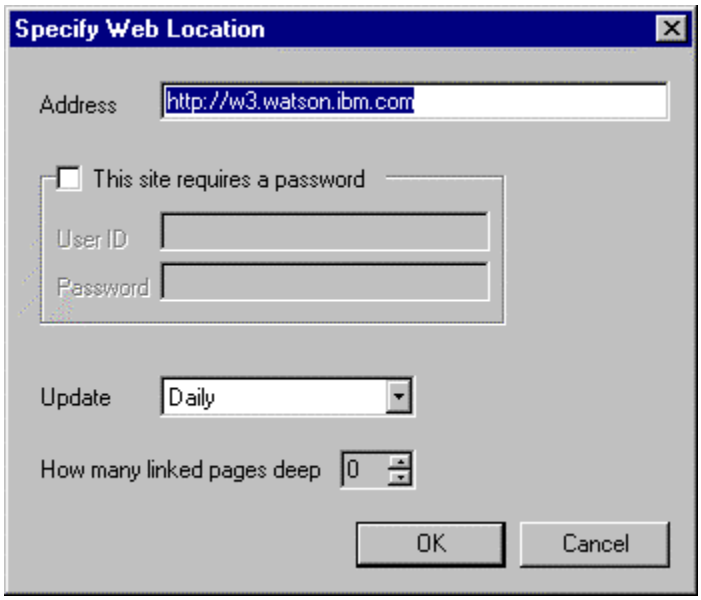

Figure 4-37. Editing the settings for a URL.

To add a URL to the list of cached pages, the user clicks the *Add* button. The Specify Web Location dialog appears where the user can enter the address, update frequency, and link depth for the new page. When done entering the data, the user clicks *OK* to save the settings or *Cancel* to quit without saving.

To set up Notes synchronization, the user highlights the Synchronize Lotus Notes Mail and Databases task in the Synchronize page and clicks the Settings button.

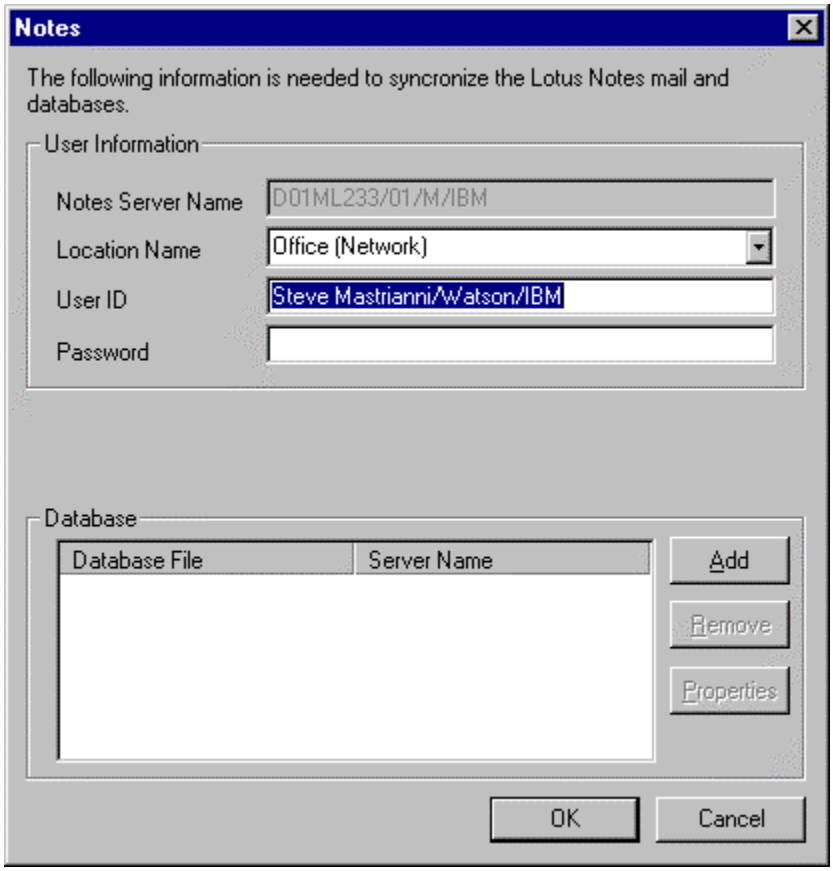

Figure 4-38. Editing Lotus Notes and Notes database settings.

If the user had previously installed Notes, the user's Notes mail server name and Notes ID should be filled in for them in the Notes synchronization dialog. The

Location Name field contains a default Notes location called “Office (Network)”. In order to perform unattended or automatic synchronization, the Notes location (not to be confused with the EasyStreet location) must be set to the location the user normally uses when connected to the LAN. This Notes location is normally called “Office”, but users can customize the names of their Notes locations. To see a list of Notes locations, start Notes and click the Office location tab on the bottom right of the Notes window.

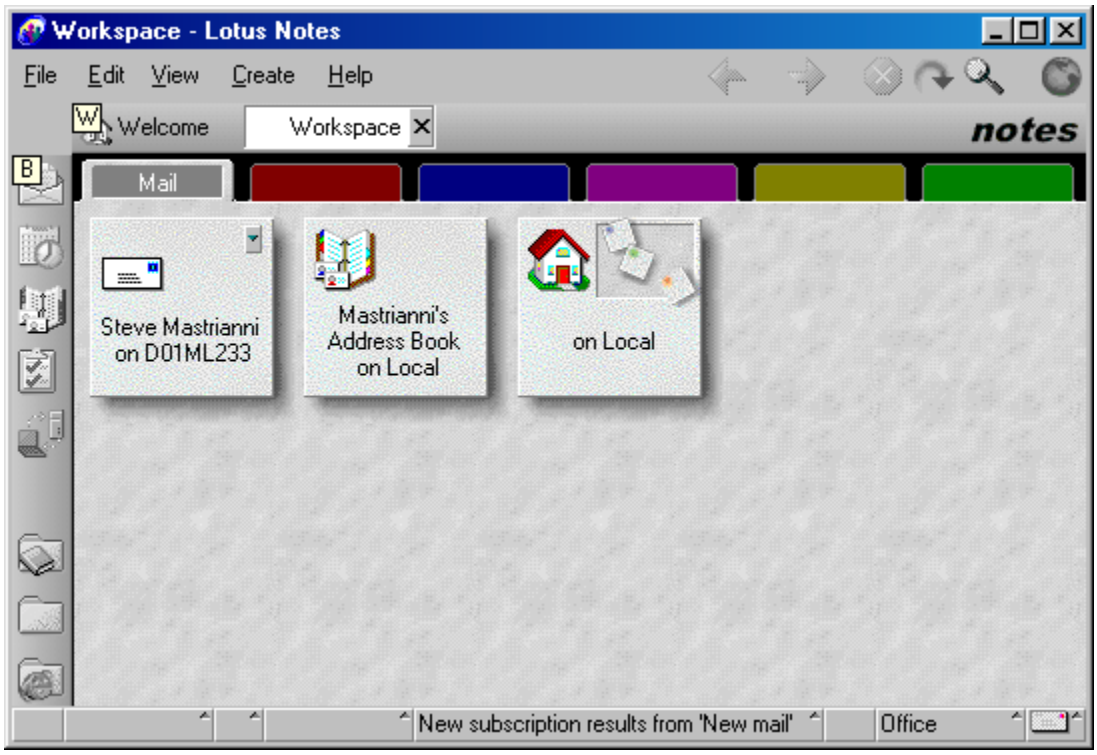

Figure 4-39. Lotus Notes location settings.

Lotus Notes® uses a special type of location called Island mode for viewing Notes mail and databases while disconnected from the network. In Island mode, the system does not attempt to connect to the dialup or local area network. This allows users to browse synchronized mail and databases without being connected. If Notes is left in the Island mode, EasyStreet cannot start the Notes replication because the Notes client is configured for off-line use. To get around this limitation, EasyStreet temporarily switches the Notes location to the location the user normally uses while connected to the network, performs the synchronization, and sets the Notes location back to the original mode.

EasyStreet can also synchronize selected Notes databases. However, the database must already exist as a local replica on your system. To add a Notes database to the synchronization list, a local replica of the database must already exist.

If a local replica already exists, the user can enable replication for a local database by clicking the *Add* button from the Notes synchronization settings page. The user can then select the database (NSF) file and the server name to synchronize with. When done, the user clicks *OK* to keep the settings or *Cancel* to discard them.

To configure EasyStreet for POP3 mail, the user selects the Internet Mail settings from the Synchronization page and then clicks the Settings button.

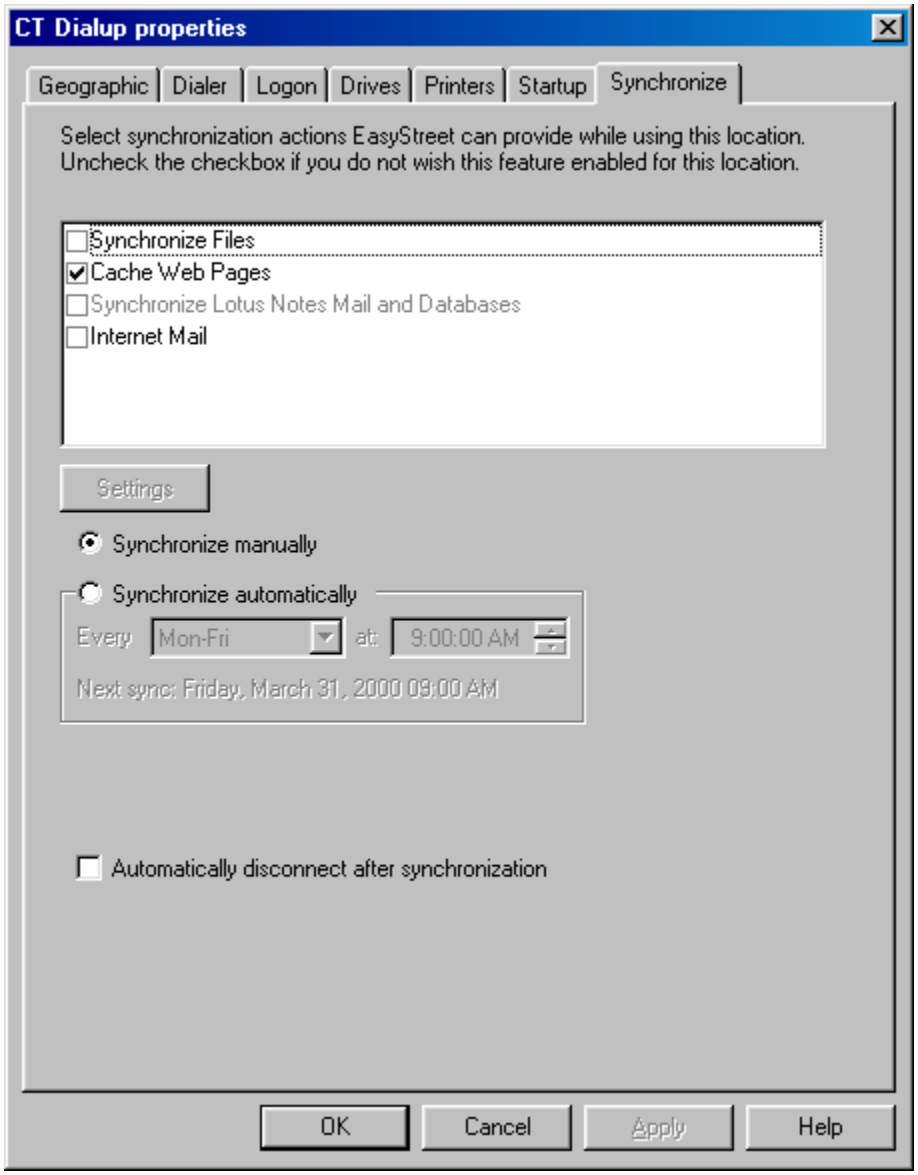

Figure 4-40. POP3 mail settings.

In the POP3 dialog, the user selects the mail client to be used for sending and receiving POP3 mail. The user then enters the name of the incoming mail server and the outgoing SMTP server, and the user ID and password for the mail account. The user then clicks *OK* to save the changes or *Cancel* to discard the changes.

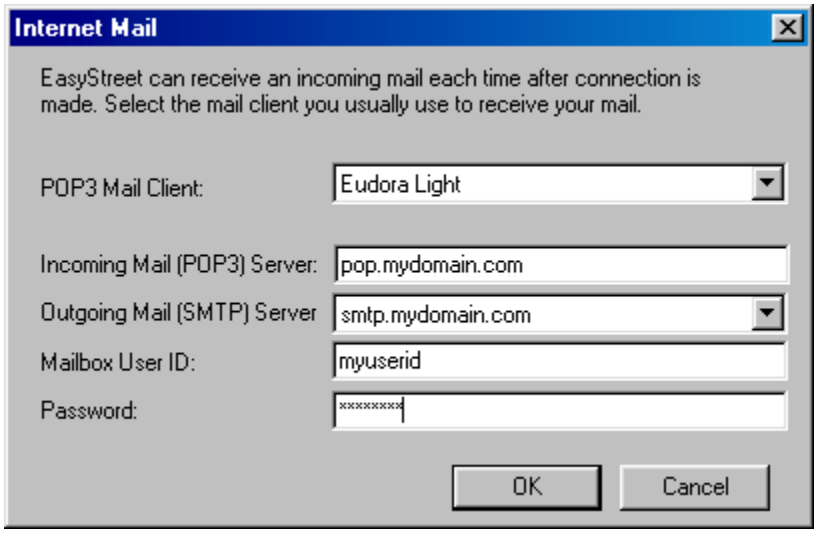

Figure 4-41. POP3 mail settings.

At the next synchronization time, any new mail on the server is retrieved, and any mail that the user has composed and sent while offline is transferred to the SMTP server to be sent.

To prevent unauthorized mail clients from using a provider's mail server to send junk email or SPAM, many providers require that the mail client first check the server for new mail before attempting to send any queued mail. If the user ID and

password are valid, the provider grants the user temporary access to the SMTP server for a short period of time, usually 30 minutes to one hour. Because of this requirement, EasyStreet always checks for mail on the server before attempting to send any queued mail.

Chapter 5. - Summary and Conclusions

Summary

Results of the Pilot

User feedback was generally positive. The test group consisted of users with a wide range of computer experience from novice to expert. Some of the testers were casual users that used their system simply to connect and browse the Web, while others were programmers and computer engineers that had a good understanding of how the application worked. Not surprisingly, the detail and depth of the user feedback varied greatly from user to user.

Some of the testers were nomadic users who used their system at the office and at home. We feel that these were poor candidates, as they would probably see the least benefit from EasyStreet. These users usually connected to a local LAN at the office and used a dialup connection at home. Since the dialup provider resolves the network configuration issues, it was not necessary to select a special location object for the home connection. IN fact, many of these nomadic users actually did not need or use the location management functions of EasyStreet. They did, however, find the file synchronization, email, and Web page cache to be very useful.

Several of the testers, however, were road warriors who spent a majority of their time on the road. For these users, EasyStreet provided the most benefit and consequently the best feedback for the pilot. For these users, the location management functions of EasyStreet proved invaluable, and many asked if they could keep the copy of EasyStreet on their system indefinitely. They also found the synchronization and Web page hoarding functions very useful, and found that modifying complex network configuration parameters using EasyStreet's configuration program was easy.

The overall response was that EasyStreet made network configuration much simpler, and that the added benefits of file synchronization, email, and Web page hoarding made the program very attractive to road warriors. Casual users, however, did not feel as strongly about the usefulness of EasyStreet.

Testers that work in the IT area expressed concern that users could modify the parameters in such a way that the system would be rendered unusable or unable to boot. There were several suggestions on how this could be prevented, such as read-only files, encrypted configuration files and disabling the ability of the user to change certain critical parameter values. The IT representatives also felt that the EasyStreet location files should be available from the Web, and that users should be required to periodically check in with a server to get automatic updates to the program and location objects.

Conclusions

Wireless Connectivity

Mobile computing will become much more pervasive in the next five years. Some preliminary studies and estimates show that by the year 2005, the number of wireless subscribers around the world will number more than one billion (see Table 5-1).

Region	1996	1997	1998	1999	2000	2001	2002	2003
Americas	54.2	71.2	91.6	108.2	131.5	157.2	174.8	189.7
Europe	37.9	60.8	74.3	77,7	193.5	116.5	126.5	138.1
Japan	26.9	38.3	48.5	61.2	80.1	95.3	100.1	103.7
Rest of World	27.6	47.8	67.0	94.0	135.0	205.0	287.0	368.6
World Total	146.6	218.1	281.4	352.1	450.1	574.1	688.4	800.1

Table 5-1. Wireless subscribers worldwide.
Source: ITU & Micrologic Research.

One of the factors limiting the use of wireless services is the relatively slow connection speed available today. In the near future, 3G wireless technologies will enable a new set of applications by providing high-speed wireless access of up to two megabits per second. The first 3G call was completed in April 2000, but the first 3G networks will be deployed in Europe and will likely not be deployed in the United States until 2001. The availability of high-speed wireless connectivity

is expected to spur a large growth in the number of wireless subscribers. That expected growth is illustrated in Table 5-2).

Region	2000	2001	2002	2003	2004	2005
Americas	0.0	0.0	0.1	1.9	5.2	20.6
Europe	0.0	0.8	2.7	4.9	8.6	12.5
Japan	0.1	0.9	2.1	4.2	7.8	12.5
Total	0.1	1.7	4.9	11.0	21.6	40.0

Table 5-2 Expected growth of 3G subscribers.
Source: Micrologic Research

As the number of subscribers increases, the capacity of the existing infrastructure will be continuously challenged to provide better coverage and more reliable connections for all subscribers. Getting connected may be difficult at times as wireless providers struggle to handle the increased traffic and a large number of simultaneous users. Getting connected and staying connected may be a challenge, so making the best of the time the user remains connected will become even more important.

Even with the advent of faster connections, the ability to prioritize content delivery will continue to be important, especially when using cellular or satellite connections. As connectivity improves, Web page designers will attempt to take advantage of that extra bandwidth by adding more and more content to Web

pages such as digital audio and real-time video. It is important to be able to filter some of that content depending on the type and speed of connection.

Wired Connectivity

While higher-bandwidth connectivity and lower costs will make wireless access more pervasive, the number of subscribers that subscribe to broadband and DSL technologies will likely grow over 400 percent in the next year alone. Long distance carriers are racing to deliver broadband local access services to small and midsize businesses, giving them the speed and reliability now available only to large enterprises on a T1 line. Businesses see these local services as cost-effective alternatives to high-speed leased lines.

DSL will likely double its market penetration before the end of 2000. AT&T, one of the largest suppliers of DLS technology, expects that access speed, now in the range of 144 thousand bits per second to 1.5 megabits per second, will be increased to beyond 7 megabits per second within the next few years.

Both DSL and cable will become pervasive in hotels, airports, business centers and public libraries in the next three years. Although no one is sure which technology will come out on top, it is clear that high-speed access will become available in most metropolitan areas before 2005. Connecting to these networks

will not always be easy, however. Although there is some movement to standardizing network access protocols, there are still many exceptions.

At the physical layer, broadband and DSL carriers have standardized on Ethernet. The availability of inexpensive network interface cards (NICs) has made Ethernet very attractive since most carriers supply users with the NIC as part of the installation cost. A 10 megabit-per-second Ethernet card can be purchased for approximately $10.00 in quantity, and 10 megabit-per-second hubs and routers can be purchased for under $50.00. Support for Ethernet NICs is standard in most every operating system, making installation easy.

TCPIP is the standard network protocol, but suppliers seem to be split on whether to use static IP addresses or DHCP. Many cable carriers still use a static IP to verify the subscriber, although some have migrated to DHCP using the machine name to verify the client. In the DSL area, many installations are still being done using static IP. Users who set up systems for outside access via telnet or FTP need servers with static IP addresses because the name of their machine cannot be resolved by an external name server.

Many NT networks still use a WINS server that allows users to resolve NetBIOS names using the TCPIP protocol. The WINS server requires a static IP address, which means that the client connecting to the network needs to know the IP address in advance.

A large number of Windows NT networks use the NetBIOS protocol to resolve the names of printers and to map network drives. NetBIOS is not available on all networks, however, because some routers or firewalls block NetBIOS. In this case, the TCPIP protocol is used to route NetBIOS traffic with the network option called *NetBIOS over TCPIP*.

While we have focused somewhat on the problem of connecting to WANs, connecting to LANs presents a bigger problem. Connecting to a local LAN requires knowledge of the existing LAN configuration, something that is not always east to obtain. We could, of course, select a default configuration such as TCPIP and DHCP. In fact, the HiDS algorithm does just that if no other information is available. But in many cases, this won't work because the network requires a fixed IP address and requires that the client know the IP address of the WINS server and gateway, the name of the domain.

Where it really gets difficult is when a user travels from one site to another. Even in the same company, no two LAN configurations are identical. They are usually often configured by a system administrator or IT staff member, and often reflect the personal bias or background of the LAN administrator. One site might require a WINS server address, static IP, and socks server name while another site belonging to the same company might require DHCP. Changing between these modes is not easy.

In the current operating systems such as Windows, only one set of network configuration parameters can be saved at one time. If the user needs to connect to a different type of network, he or she must manually change several parameters in the operating system. Changing these parameters could render the system unusable, even unbootable if the wrong parameters are changed or changed incorrectly. While some experienced programmers and engineers don't find this difficult, it is nearly impossible for the more causal user to do. It is, therefore, absolutely necessary to have a program such as EasyStreet which makes changing these settings easy.

Access to Data

For mobile and nomadic users, access to their data is the most important aspect of their mobile computing environment. Getting connected and staying connected is critical, yet it is essentially only the enabling technology that allows users to access their data. Users should be able to access their data from any physical location and from any type of network providing they have the proper access information.

For nomadic users, this involves connecting to their network from a remote location such as a hotel or airport. In the past, this has usually meant dialing in to their network from an analog phone. Recently, however, hotels, airports, and

business centers have begun to offer high-speed access to the Internet using cable, DSL, and in-house servers. Many of these hotels now offer rooms with Ethernet jacks that are connected to the hotel's server.

Users with Ethernet adapters can quickly get high-speed Internet access by plugging their laptop into the RJ-45 receptacle, providing the network configuration has been se up correctly on the user's machine. A large number of these installations have standardized on TCPIP with DHCP enabled. Users that normally connect to a TCPIP network with DHCP will have no problem at all connecting to the same type of network in the hotel. However, users that normally connect to their network with a fixed IP address, fixed gateway or fixed WINS server will not be able to get connected without making some major network parameter changes.

With EasyStreet, this so simple that the user need not know any details of the network configuration or what parameters need to be changed. The information for the network configuration is in the EasyStreet location object. The user simply selects a location object for the physical location when the system is started, and the network configuration is updated automatically, resulting in a positive connectivity experience for the user.

This is made even easier if the various physical locations agree on a network configuration and protocol. For example, if all Holiday Inns agree that the network

configuration should be TCPIP with DHCP, EasyStreet need only include a single location object called “Holiday Inn”. When the user arrives at a Holiday Inn anywhere in the country, they simply select the “Holiday Inn” location and are immediately connected to the Internet via the hotel’s network.

Until all the differences in network configuration, protocols, and access are standardized, an application such as EasyStreet will be required make trouble-free access to data from any location possible.

Appendix A - Glossary

100Base-T -- An IEEE 802.3 extension for providing Ethernet transmission at 100 Mbps on twisted pair and powered signal-regenerative hubs.

10Base-T -- Ethernet IEEE 802.3 standard for transmission at 10 Mbps specifically for twisted-pair wiring and connectors and signal-regenerative powered hubs.

10Base2 -- Ethernet standard for baseband networks with transmission rates of 10 Mbps, using co-axial cable segment lengths of 2*100 meters (200 meters).

10Base5 -- Ethernet standard for baseband networks with transmission rates of 10 Mbps, using co-axial cable segment lengths of 5*100 meters (500 meters).

A Band -- A non-wireline radio frequency spectrum. *See also* B Band.

Access Charge -- A flat monthly fee charged a subscriber for the use of a cellular system (whether the subscriber makes or receives any calls or not).

Access Number -- The phone number that must be dialed by someone calling you when you are roaming outside of the National Network, prior to dialing the number of your phone. The access number gives the caller access to the facilities of the system in which you are roaming.

Adjacent Cell -- Two cells are adjacent if it is possible for an MES to maintain continuous service while switching from one cell to another.

ADPCM -- Adaptive Differential Pulse Code Modulation.

Air Link Interface -- The network interface between a MES and the CDPD service provider network.

AMPS (Analog Mobile Phone Service) -- The official name for the first commercial cellular system, which used 666 channels (A Band 333 and B Band 333). The standard for the analog cellular telephone service in use in North America.

Analog Signal -- A transmission in which information is represented as physical magnitudes of electrical signals.

ANI -- Automatic Number Identification

APCO 25 (Association of Public Safety Communications Officials) -- A set of standards for private radio networks designed specifically for public safety (police, fire, emergency services) applications in North America

ARQ (Automatic Repeat Request) -- Communication method where the receiver detects errors and request retransmissions.

AUI (Attachment Unit Interface) -- An IEEE 802.3 cable connecting the MAU (Media Access Unit) to the networked.

Attenuation -- Loss of communication signal energy.

ATMD (Asynchronous Time Division Multiplexing) -- A method of sending information in which normal time division multiplexing (TDM) is used, except that to, slots are allocated as needed rather than specific transmitters.

AUC -- Authentication Center

Bandwidth -- 1. The range (band) of frequencies that are transmitted on a channel. The difference between the highest and lowest frequencies is expressed in hertz (Hz) or millions of hertz (MHz). 2. The range frequencies on the electromagnetic spectrum allocated for wireless transmission. 3. The wire speed of the transmission channel.

Base Station -- The low power transmitter/receiver and signal equipment located in each cell in a cellular service area.

Baseband -- A transmission channel that carries a single communications channel, on which only one signal can transmit at a given time.

BCHO (Base Controlled Handoff) -- Cell transfer initiated by the network.

BOC (Bell Operating Companies) -- The local telephone companies that existed prior to deregulation, under which AT&T was ordered by the courts to divest itself in each of the seven U.S. regions. *See also* RBOC.

Block A -- The block of 800-MHz cellular radio frequencies assigned to the non-wireline or Block A carrier.

Block B -- The block of 800-MHz cellular radio frequencies assigned to the wireline or Block B carrier.

Bluetooth – a wireless technology as well as a specification for small-form factor, low-cost, short range radio links between mobile PCs, mobile phones and other portable devices.

Broadband – A transmission channel that allows many simultaneous channels to transmit and receive at the same time at a high rate of speed. The most common type of broadband system is the one used by cable television companies to deliver cable television signals and cable modem Internet access.

BTS -- Base Transceiver Station

Call Setup Time -- The time required to establish a switched call between DTE and devices.

CCITT (Consultative Committee for International Telephone and Telegraph) -- An international organization that makes recommendations for networking standards like X.25, X.400, and facsimile data compression standards. It is wow called the International Telecommunications Union Telecommunication Standardization Sector; and abbreviated as ITU, ITU-T, or ITU-TSS.

CDMA -- Code-Division Multiple Access.

CDPD (Cellular Digital Packet Data) -- Uses idle moments on voice channels to send pure data over the channel, without affecting quality of voice transmissions.

Cell -- The basic geographic unit of a cellular system and the basis for the generic industry term "cellular." A geographical area is divided into small "cells", each of that is equipped with a low-powered radio transceiver. A computer at the Mobile telephone switching office monitors the movement and transfers (or hands off) the phone calls to another cell and another radio frequency as needed.

Cell Splitting -- Dividing one cell into two or more cells to provide additional capacity within the original cell’s region of coverage.

Channel -- An individual communication path that carries signals at a specific frequency. The term also is used to describe the specific path between large computers and attached peripherals.

Channel Bandwidth -- The frequency range of a RF channel. In a CDPD, it is 30 KHz.

Channel Hop -- The process of changing the RF channel supporting a channel stream to a different RF channel on the same cell.

Channel Hopping -- A radio frequency transmission method whereby transmissions *hop* from one channel to another. The channels are visited in a predefined order specified by a hopping sequence. Typically this uses the ISM band from 2.4000 to 2.4835 GHz with 85 one-megahertz channels or "hops," but at least 50 different frequencies must be used by FCC regulation. CDPD uses

frequency hopping on analog cellular systems to take advantage of unoccupied voice channels.

CHAP -- Challenge handshake authentication protocol.

CID -- Caller ID

Circuit Switching -- An open-pipe technique that establishes a temporary dedicated connection between two points for the duration of the call.

Class of Service -- An indication of how an upper layer protocol wants a lower-layer protocol to treat messages in terms of priority, assignment on physical bandwidth pipes, routing, etc.

Code Division Multiple Access (CDMA) -- 1. A division of the transmission spectrum into codes, effectively scrambling conversations. 2. Wireless transmission technology that employs a range of radio-frequency wavelengths to transport multiple channels of communication signals.

CODEC -- Coding/Decoding Device

CSU (Channel Service Unit) -- A digital interface device that connects end-user equipment to the local digital telephone loop.

CSDS (Circuit Switched Data Service) -- *Developed* for delivery vehicles to track packages (used by UPS). Services that carry data over conventional cellular where circuits are switched from call to call.

CSMA/CD (Carrier Sense Multiple Access With Collision Detection) -- A communications protocol in which nodes contend for a shared communications channel. Simultaneous transmission from two or more nodes results in a collision. When a collision is detected, the transmission is restarted at a random time.

CT-2 -- Digital Cordless Telephony (2nd generation)

CT-3 -- Digital Cordless Telephony (3rd generation)

CTIA (Cellular Telecommunications Industry Association) -- The organization created, in 1981, to promote the cellular industry, address the common concerns of cellular carriers and serve as a forum for the exchange of nonproprietary information.

Crosstalk -- A technical term indicating that stray signals from other wavelengths, channels, communication pathways, or twisted-pair wiring have

polluted the signal. It is particularly prevalent in twisted-pair networks or when telephone and network communications share copper-base wiring bundles.

CSMA/CA -- Carrier Sense Multiple Access With Collision Avoidance.

CSMA/CD -- Carrier Sense Multiple Access With Collision Detection.

CSU -- Channel Service Unit.

CT2/Telepoint --A U.K. system, which allows callers with CT2 phones who are within 200 yards of a base station (or site) to make outgoing calls.

CTI -- Computer Telephone Integration.

CTIA -- Cellular Telecommunications Industry Association.

DACS -- Digital access and cross-connect systems.

Data Compression -- A reduction in the size of data by exploiting redundancy. Many modems incorporate MNP5 or V.42bis protocols to compress data before it is sent over the phone line.

dB (Decibels) -- A value expressed in decibels is determined as 10 times the logarithm of the value taken to base 10.

DCE -- 1. Data Communications Equipment.

Dead Spot -- A location in a radio/cellular system where, for one reason or another, signals do not penetrate.

DECT -- Digital European Cordless Telephone

DES (Data Encryption Standard) -- An encryption/decryption algorithm defined in FIPS Publication 46. The standard cryptographic algorithm developed by the National Institute of Standards and Technology

Decompression -- The restoration of redundant data that was removed through compression.

DECT (Digital European Cordless Telephone) -- The specs for future European cellular, as yet not fully defined.

DSU (Data Service Unit) -- A device used in digital transmission for connecting data terminal equipment (DTE), such as a Router, to a digital transmission circuit (DTC) or service.

DTE (Data Terminal Equipment) -- A computer terminal that connects to a host computer. It may also be a software session on a workstation or personal computer attached to a host computer.

DTMF -- Dual Tone Multi Frequency

Dual-Mode -- New cellular phones that work with both digital and analog switching equipment.

Dual-NAM -- Allows user to have two phone numbers with separate carriers.

Duplex -- 1. The method in which communication occurs, either two-way as in full duplex, or unidirectional as in half duplex. 2. Cellular phones that use separate frequencies for transmission and reception, allowing both parties to talk and listen at once.

EMI (Electromagnetic Interference) -- Interference by electromagnetic signals that can cause reduced data integrity and increased error rates on transmission channels. Signal noise pollution from radio, radar, fluorescent lights, or electronic instruments.

Encryption --The processing of data under a secret key in such a way that only a recipient in possession of a secret key can determine the original data.

ETC (Enhance Throughput Cellular) -- AT&T Paradyne protocol for data transmission over analog cellular connections consisting of enhancements to V.42 and V.32bis for compression, error detection, and error correction.

ECC (Enhanced Control Cellular) -- Proprietary protocol from Motorola for cellular modems.

Erlang -- 1 hour, 300 seconds, and 36 CCs. If a channel is occupied (used) constantly for 1 hour, that circuit has carried 1 Erlang of traffic.

ESMR -- Extended Specialized Mobile Radio

Fading -- The combination of out-of-phase multiple signals that results in a weaker or self-canceling data signal.

FDDI (Fiber Data Distributed Interchange) -- FDDI provides 125 Mbps signal rate with 4 bits encoded into 5-bit format for a 100-Mbit per second transmission rate.

FDMA (Frequency Division Multiple Access) -- The analog communications technique that uses a common channel for communication among multiple users allocating unique time slots to different users.

FES (Fixed End System) -- The non-mobile communication system (and software) that handles OSI transport and higher layers of CDPD transmission.

Firewall -- A device, mechanism, bridge, Router, or gateway that prevents unauthorized access from outside the firewall.

Frequency Hopping -- A radio frequency transmission method. Typically this uses the ISM band from 2.4000 to 2.4835 GHz with 85 one-megahertz channels or "hops." CDPD uses frequency hopping on analog cellular systems to unoccupied voice channels. Transmissions hop from one channel to the other, staying only 1/10 of a second on any given channel. The channels are visited in a predefined order specified by a hopping sequence.

Geosynchronous Orbit -- Orbit taken by satellites where the satellite's orbit velocity matches the rotation of the earth, causing the satellite to remain stationary relative to a position on the earth's surface.

GIS (Geographic Information System) -- Generally refers to a database of geographical data. In some circles, it refers to Graphics database.

GLS (Global Locationing System) -- A triangulation system used to locate a vehicle and convey that information to a central management facility.

GPS (Global Positioning System) -- A satellite-based triangulation system used to ascertain current location.

GSM (Global System For Mobile Communications) -- The pan-European digital cellular system standard.

Handoff -- The transfer of responsibility for a call from one cell site to the next.

Hopping Sequence -- The pre-set order in which frequency-hopping RF transmissions are distributed over the 82 channels of the assigned ISM band.

Host -- Any computer, although typically a mainframe, mid-sized computer, mini-computer, or LAN server, servicing users and their processing at the centrally-based processor but distributing the results to terminal based or client connections.

IEEE (Institute for Electrical and Electronic Engineers) -- A membership-based organization based in New York City that creates and publishes technical specifications and scientific publications.

IEEE 802 -- An Institute of Electrical Engineering standard for interconnecting of local area networking computer equipment. The IEEE 802 standard describes the physical and link layers of the OSI reference model.

IEEE 802.1 -- A specification for media-layer physical linkages and bridging.

IEEE 802.3 -- An Ethernet specification derived from the original Xerox Ethernet specifications. It describes the CSMA/CD protocol on a bus topology using baseband transmissions.

IEEE 802.4 -- Broadband and base band bus using token passing as access method, and physical interface specifications.

IEEE 802.5 -- A token ring specification derived from the original IBM Token-Ring LAN specifications. It describes the token protocol on a star/ring topology using baseband transmissions.

IEEE 802.6 -- A token bus specification for metropolitan area network with star/ring topology using baseband transmissions.

IEEE 802.11 -- A physical and MAC layer specification for wireless network transmission based on direct and frequency hopping, SST, and infrared at transmission speeds from 1 to 4 Mbps.

IEEE 802.12 -- A specification for wireless network transmission based on SST.

IMTS (Improved Mobile Telephone Service) -- Cellular telephone predecessor that uses a single central transmitter and receiver to service a region.

In-Band Signaling -- Transmission within a frequency range normally used for information transmission. Contrasted with out-of-band signaling, which uses frequencies outside the normal range of information-transfer frequencies.

IP -- Internet Protocol.

IPX -- Internet Packet Exchange (Novell Netware LAN protocol)

IS 54 -- Interim Standard developed by CTIA for introduction of TDMA in conjunction with FDMA.

IS-41 -- TIA cellular standard for seamless roaming with inter-system handoff, call delivery, validation, and authentication.

IS-54 -- TIA cellular standard defining the air interface to TDMA and digital handsets to base station communications.

IS-95 -- TIA cellular standard defining the air interface to CDMA and digital handsets to base station communications.

ISDN -- Integrated Services Digital Network

ISM -- Instrumentation, Scientific, and Medical band.

ISO -- International Standards Organization.

Ka-band -- A high-bandwidth satellite wireless communication frequency using the 30 GHz spectrum.

Lease Line -- A dedicated Common carrier circuit providing point-to-point or multipoint network connection, reserved for the permanent and private use of a customer.

LEO -- Low-Earth Orbit Satellite.

Local Loop -- The line from a telephone subscriber's premises to the telephone company CO.

MAN (Metropolitan Area Network) -- A network that spans buildings, or city blocks, or a college, or corporate campus. Optical fiber repeaters, bridges, routers, packet switches, and PBX services usually supply the network links.

MASC (Mobitex Asynchronous Communication) -- Asynchronous link level protocol based on Mobitex.

MCSS (Mobile Communications Server Switch) -- A hardware/software configuration that provides communications connection and message switching functionality. It sits between the wireless network and information servers.

MCP/1 -- Mobitex Compression Protocol

MDBS (Mobile Data Base Station) -- The hardware used by a cellular provider to convert the data streams into a valid signal and route cellular switched data calls through the wired phone network or to the cellular destination.

MES (Mobile End System) -- The portable wireless computing device that can roam from site to cell while communicating with the MDBS via CDPD.

MHX- (Mobitex Main Hierarchical Exchange) -- Part of Mobitex network hierarchy. Each MHX exchanges information with other MHX's.

MIB (Management Information Base) -- an SNMP term

Microwave -- Electromagnetic waves in 1 to 30 GHz range

MIN -- Mobile Identification Number

MMTF (Mobile Management Task Force) -- A vendor organized body that has undertaken to create SNMP based MIB for mobile network management

MPAK -- Mobitex packet that is routed through Mobitex network. A 512 octet of user data.

MOX (Mobitex Area Exchange) -- A node in the internal Mobitex network.

MNP (Microcom Network Protocols) -- A set of modem-to-modem protocols that provide error correction and compression.

MNP10 -- Microcom Network Protocols for cellular or wireless transmission applying compression, error detection and error correction, data rate fallback, and readjustment.

MNP5 -- Microcom Network Protocols with simple data compression. Dynamically arranges for commonly occurring characters to be transmitted with fewer bits by encoding long runs of the same character.

MPT/1 -- Mobitex Transport Protocol

MPAK -- Mobitex Packet

MSA (Metropolitan Statistical Area) -- The 30 U.S. urban areas (markets) as defined by the FCC, using SMSA (Standard Metropolitan Statistical Area) data. All are licensed for two cellular operators, and almost all have both operators on the area. MSA's comprise 76% of the U.S. population, but only 22% of its land surface area.

MTSO (Mobile Telephone Switching Office) -- The cellular system's switching computer, located between a cell site and a conventional telephone switching office.

NAMPS (Narrow Band Advanced Mobile Phone System) -- Using a radio frequency transmission on a single, preset frequency.

Narrow Band -- PCS frequency in 900-931 MHz range for 2-way paging.

NDIS (Network Device Interface Specification) -- A Microsoft network interface specification for operating system and protocol independent device drivers supported by IBM LAN Manager and Microsoft Windows NT.

NIC (Network Interface Card) -- The network access unit that contains the hardware, software, and specialized PROM information necessary for a station to communicate across the network.

NOS (Network Operating System) -- A platform for networking services that combines operating system software with network access.

ODI (Open Data-link Interface) -- A protocol that supports media- and protocol-independent communication by providing a standard interface allowing network layer protocols to share hardware without conflict. Presently used in PC software, mostly.

OLTP -- On-Line Transaction Processing

PCCA -- Portable Computer and Communications Association. A non-profit association of vendors to develop and promote software and hardware for mobile computing applications

PC Card Standard -- Latest PCMCIA specification PCMCIA 5.0. Adds support for low-voltage 3.3 volt operation, DMA, multi-function capability, and CardBUS - that provides higher performance, bus mastering and 32-bit data path.

PCMCIA (Personal Computer Memory Card International Association) -- A standard for a computer plug-in, credit card-sized card that provides about 90 percent compatibility across various platforms, BIOS, and application software.

PCS (Personal Communications Services)/ PCN (Personal Communications Network) -- A term used to describe emerging wireless/portable network technology where subscribers carry their own personal communication numbers with them, and the system locates them wherever they are.

POP -- Point-Of-Presence

POPS -- One unit of population. The POPS concept is used to measure relative market sizes.

POTS -- Plain Old Telephone Service.

PSTN (Public Switched Telephone Network) -- The telecommunications network traditionally encompassing local and long distance landline carriers and now also including cellular carriers.

PTC -- Portable Transaction Computer - A term used in manufacturing, distribution and warehousing industry to denote portable computers equipped with scanners, bar code readers or a pen

RJ-11 -- Standard 4-wire connectors for phone lines.

RJ-22 -- Standard 4-wire connectors for phone lines with secondary phone functions (such as call forward, voice mail, or dual lines).

RJ-45 -- Standard 8-wire connectors for networks. Also used as phone lines in some cases.

Roaming -- The ability to access a network anywhere and move freely while maintaining an active link through a wireless connection to a network. Roaming usually requires a handoff when a node (user) moves from one cell to another.

Router -- A device that interconnects networks that are either local area or wide area.

SMR -- Specialized Mobile Radio.

SMP (Symmetric Multiple Processing) -- processor hardware architecture that allows multiple processors to share processing workload, using common memory.

SNMP -- Simple Network Management Protocol.

SPX -- System Packet Exchange, a protocol used in Novell's Netware network operating system.

SST -- Spread-Spectrum Technology, used in wireless LANs.

SS#7 -- Signaling System # 7 - channel for network control

T-1 -- Bell technology referring to a 1.544 Mbps communications circuit provided by long-distance carriers for voice or data transmission through the telephone hierarchy.

T-3 -- An AT&T standard for dial up or leased line circuits with a signaling speed of 44.736 megabits per second.

TCP/IP -- Transaction Control Protocol/Internet Protocol.

TDMA -- Time Division Multiple Access.

TIA -- Telecommunications Industry Association

V.32 -- An international standard for synchronous and asynchronous transfer of data of up to 9,600 bps over dial-up telephone lines.

V.32bis -- An international standard for synchronous and asynchronous transfer of data of up to 14,400 bps over dial-up telephone lines.

V.34 -- An international standard for synchronous and asynchronous transfer of data of up to 28,800 bps over dial-up telephone lines.

V.42 -- An international error correction protocol that uses Link Access Procedure Modem (LAP-M) as the primary protocol, and MNP2-4 as back-up protocols.

V.42bis -- An international data compression protocol that can compress data by as much as 4-to-1.

VSAT -- Very-Small-Aperture Terminal.

WAN -- Wide Area Network.

Index

Bibliography

[Alzo94] Alzone, M., Hovey, R., Liau, W., Luss, A., Nagpal, A., Schiller, S., Winter, R., and Zavery, A. “A Framework for Ubiquitous Mobile Computing” - Masters Thesis, October 1994.

[Ande97] Anderlind, E., "Resource Allocation in Multi-Service Wireless Access Networks". Dept. of Signals, Sensors and Systems, Royal Institute of Technology, Stockholm, Sweden, October 1997.

[Badr98] Badrinath, R. and Welling, G. “Event Delivery Abstractions for Mobile Computing”, Rutgers University, June 1998.

[Bure97] Buretta, M. *Data Replication*. New York: John Wiley, 1997.

[Cho96] Cho, G., "Location and Routing Optimization Protocols Supporting Internet Host Mobility". Newcastle University, Newcastle, UK, May 1996.

[Domm98] Dommety, G., "Efficient Techniques to Support Mobile Hosts in Wireless Networks". Dept. of Computer and Information Science, Ohio State University, Columbus, Ohio, USA, July 1998.

[Good97] Goodman, D. *Wireless Personal Communications Systems*. Massachusetts: Addison-Wesley, 1997.

[Hela99] Helal, A., Haskell, B., Carter, J., Brice, R., Woelk, D., and Rusinkiewicz, M. *Any Time, Anywhere Computing – Mobile Computing Concepts and Technology*. Boston: Kluwer Academic Publishers, 1999.

[Ho96] Ho, J. S. M., "Mobility Management for Personal Communications Networks," Dept. of Electrical and Computer Engineering, Georgia Institute of Technology, May 1996.

[Huur97] Huurdeman, A. *Guide to Telecommunications Transmission Systems*. Massachusetts: Artech House, Inc., 1997.

[Faiz95] Faiz, M., Zaslavsky, A. “Database Replica Management Strategies in Multidatabase Systems with Mobile Hosts”. Accepted for presentation at the 6th International Hong Kong Computer Society Database Workshop: Database Reengineering and Interoperability, Hong Kong, March 1995.

[Jaya98] Jayaram, R, "Quality-of-Service Provisioning and Resource Reservation Mechanisms for Mobile Wireless Networks". University of North Texas, Denton, Texas, USA, August 1998.

[Joa99] Joa-Ng, M., "Routing Protocol and Medium Access Protocol for Mobile Ad Hoc Networks". Polytechnic University, NY, USA, June 1999.

[Kist93] Kistler, J. “Disconnected Operation in a Distributed File System” - PhD Thesis, May 1993.

[Kris99] Krishnamurthi, G., "Location Management and Fault Tolerance Issues in Mobile Networks". Iowa State University, Ames, IA, May 1999.

[Kris96] Krishna, P., "Performance Issues in Mobile Wireless Networks". Texas A&M University, College Station, Texas, USA, August 1996.

[Lai95a] Lai, S.J., Zaslavsky, A.B. “Behavioural Discovery of Network Protocols Functionality for OSI Model Conformance Testing”. Accepted for presentation at the IASTED International Conference on Modeling &Simulation, Pittsburgh, USA, April 1995.

[Lai95b] Lai, S.J., Zaslavsky, A., Martin, G.M., Yeo, L.H. "Cost-Efficient Adaptive Protocol with Buffering for Advanced Mobile Database Applications". Accepted for presentation at the 4th International Conference on Database Systems for Advanced Applications, DASFAA'95, 10-13 April 1995, Singapore.

[Lai95c] Lai, S.J., Zaslavsky, A., Martin, G.M. (1995), "A Simulation Model of Adaptive Protocols in Distributed Computing Systems with Mobile Hosts". Accepted for presentation at the 28th Annual Simulation Symposium, Arizona, Phoenix, April 1995.

[Lam97] Lam, D., "Location Management Techniques and Teletraffic Modeling for Large Wireless Communications Networks". Stanford University, CA, USA, November 1997.

[Liu96] Liu, G., "The Effectiveness of a Full-Mobility Architecture for Wireless Mobile Computing and Personal Communications". Royal Institute of Technology, Stockholm, Sweden, March 1996.

[Mast97] Mastrianni, S., Chang, H., Tait, C., Shapiro, M., Housel, B., Lindquist, D. “Web Browsing in a Wireless Environment: Disconnected and Asynchronous Operation in ARTour Web Express”. Proceedings of the third annual ACM/IEEE international conference on Mobile computing and networking, Budapest, Hungary, 1997.

[Mumm94] Mummert, L., and Satyanarayanan, M. “Large Granularity Cache Coherence for Intermittent Connectivity”. April 1994.

[Nels98] Nelson, G. J., "Context-Aware and Location Systems". University of Cambridge, United Kingdom, January 1998.

[Nobl94a] Noble, B., and Satyanarayanan, M., “An Empirical Study of a Highly Available File System”. February 1994.

[Nobl94b] Noble, B., Satyanarayanan, M., Kumar, P., Price, M. “Application-Aware Adaptation for Mobile Computing”. In Proceedings, 6 ACM SIGOPS European Workshop, September 12-14, 1994.

[Perk97] Perkins, C. *Mobile IP*. Massachusetts: Addison-Wesley, 1997.

[Perk98] Perkins, C. *Mobile IP Design Principles and Practices*. Massachusetts: Addison-Wesley, 1998.

[Pope96] Pope S., "Application Support for Mobile Computing". Computer Laboratory, University of Cambridge, U.K, October 1996.

[Ramj97] Ramjee, R, "Supporting Connection Mobility in Wireless Networks". Department of Computer Science, University of Massachusetts, Amherst, MA, USA, May 1997.

[Ratn98] Ratner, D. H., "Roam: A Scalable Replication System for Mobile and Distributed Computing". University of California at Los Angeles, CA, USA, January 1998.

[Saty93] Satyanarayanan, M., Kistler, J., Mummert, L., Ebling, M., Kumar, P., Lu, Q. “Experience with Disconnected Operation in a Mobile Computing Environment”. Carnegie Mellon University. June 1993.

[Saty94a] Satyanarayanan, M., Noble, B., Kumar, P., and Price, M. “Application Aware Adaptation for Mobile Computing”. 1994.

[Saty94b] Satyanarayanan, M. and Mummert, L. “Long Term Distributed File Reference Tracing: Implementation and Experience”. November 1994.

[Saty94b] Satyanarayanan, M., and Kumar, P., “Flexible and Safe Resolution of File Conflicts”. November 1994.

[Saty94d] Satyanarayanan, M., “Fundamental Challenges in Mobile Computing”. Fifteenth ACM Symposium on Principles of Distributed Computing. May 1996, Philadelphia, PA

[Saty94e] Satyanarayanan, M., “Mobile Information Access”. IEEE Personal Communications, Vol. 3, No. 1, February 1996.

[Sen97] Sen, S. K. "Efficient Resource Utilization Techniques in the Bandwidth and Location Management Problems in Mobile Computing". University of North Texas, Denton, Texas, USA, December 1997.

[Solo98] Solomon, J. *Mobile IP*. New Jersey: Prentice Hall, 1998.

[Tann96a] Tannenbaum, A. *Computer Networks, Third Edition*. New Jersey: Prentice Hall, 1996.

[Tann96b] Tannenbaum, A., Woodhull, A. *Operating Systems, 2nd Edition.* New Jersey: Simon and Schuster, 1996.

[Tann97] Tannenbaum, A., Woodhull, A. *Operating Systems Design and Implementation*. New Jersey: Prentice Hall, 1997.

[Wese97] Wesel, E. *Wireless Multimedia Communications*. Massachusetts: Addison-Wesley, 1997.

[Yeo94a] Yeo, L.H. and Zaslavsky, A. "A Layered Approach to Transaction Management in Multi-database Systems". In Proceedings of the 5th International Hong Kong Computer Society Database Workshop: Next Generation Database Systems, 1994.

[Yeo94b] Yeo, L.H. and Zaslavsky, A. "Submission of Transactions from Mobile Computers in a Cooperative Multi-database Processing Environment". Proceedings IEEE/CS 14th International Conference on Distributed Computing Systems, Poland, 1994.

[Yeo94c] Yeo, L.H. and Zaslavsky,A. "Simulation Model for Managing Mobile Workstations in Distributed Computing Environment". Australian Telecommunications Networks and Applications Conference (ATNAC'94), December, Melbourne, Australia, 1994.

[Zasl95] Zaslavsky, A.B., Yeo, L.H., Lai, S.J. “Simulation Model of Transaction Management in Mobile Computing Environment Using Coloured Petri Nets”. Accepted for presentation at the IASTED International Conference on Modelling & Simulation, Pittsburgh, USA, April 1995.

[Zhao97] Zhao, X., Baker, M. “Flexible Connectivity Management for Mobile Hosts". Stanford University, September 1997.

[Zhao98] Zhao, X., Castelluccia, C., Baker, M. “Flexible Network Support for Mobility”. In Proceedings of the Second Annual International Conference on Mobile Computing and Networking. November 1998.