Windows CE handheld systems for the corporate mobile work force.

Steven J. Mastrianni

Unionville, Connecticut

Last Updated 3/30/2005 4:22:52 PM

Abstract

In the past few years, we've witnessed a remarkable metamorphosis of the personal computer. The once clunky, monotone eyesore that frequently occupied an entire desktop has undergone a major cosmetic facelift. Personal computers have been slimmed down, sculpted to look like furniture, and made available in designer colors. Once the office fixture of the executive or financial consultant, the PC has become an integral part of every occupation. It is difficult to find any job that does not require some sort of computer knowledge or skill to perform.

A major factor that has made this metamorphosis possible is the ever-decreasing size of the personal computer. Advances in miniaturization and manufacturing technologies have allowed computer manufacturers to squeeze more and more electronic components into a smaller space. Just a few years ago, CPUs that contained a few thousand transistors were viewed as state-of-the-art. Today, CPUs with a million transistors or more are commonplace. New technology that allows the conductive paths inside the microprocessor to be closer together has allowed the microprocessor to not only run faster, but also cooler. The discrete support components such as resistors and capacitors have also been getting smaller. Current technology allows dozens of these components to occupy the same space as only one of their predecessors.

The result of all of this technical innovation has been the introduction of a family of powerful computers that can fit in the palm of one's hand. In 1995, the first of these handheld systems was introduced by Palm Computing. Dubbed the PalmPilot™ because of its palm-sized form factor, it gained immediate popularity because of its relatively low cost, long battery life and ease of use. Sales figures indicate that over two million PalmPilots have been sold. Microsoft®, meanwhile, began to take notice of the popularity of these handheld devices and introduced their own version of a palm-size PC. Initially dubbed the Palm-size PC[1], it was the result of Microsoft's effort to standardize the hardware and software for this new class of device. Sales of the Palm-size PC have been disappointing, however, despite that fact that Microsoft is behind it. Is this the sign of a failure, or is it just the beginning of a new technology that is just crossing the chasm? A third company called Psion has also introduced two handheld devices called the Psion 3 and Psion 5. The Psion enjoys a loyal following in Europe but uses a non-standard hardware and software architecture. Companies eager to deploy these mobile devices are faced with the problem of choosing which type of device to embrace. Which handheld makes the most sense? The PalmPilot is inexpensive and supported by hundreds of third-party applications. The Palm-size PC and its clamshell-sized cousins are backed by hundreds of hardware and software vendors, including the venerable Microsoft. The Psion is very popular despite its proprietary architecture.

[1] 3COM Corporation sued Microsoft claiming that PalmPC infringed on their PalmPilot trademark. On April 8, 1998, Microsoft agreed to drop the term PalmPC and used the term "Palm-size PC" instead. In this document, we use Palm-size PC to indicate palm-sized computers that run the Windows CE operating system.

We believe that the Windows® CE-based devices offer a clear advantage over the PalmPilot and the Psion. We feel that the designers of the Palm Pilot were more interested in profits, and in their attempt to keep the price low, and battery life long, they made some crucial errors in judgement. These errors in judgement, we believe, will ultimately doom the Palm Pilot to nothing more than a tick mark on the computer history timeline. We also believe that the makers of the Psion made a big mistake by choosing relatively unknown software architecture and failing to standardize the hardware.

Trademarks and Copyrights

Table of Contents

Chapter 1. - Introduction 5
Problem 5
Purpose 7
Importance of Study 7
Rationale of Study 8
Overview of the Study 10
Definition of Terms 11
Chapter 2. - Review of Related Literature 17
The PalmPilot 21
The IBM WorkPad 23
Palm Pilot Hardware 24
PalmPilot Operating System 27
PalmPilot User Interface 28
PalmPilot Input 29
PalmPilot Synchronization 30
The Psion 33
Psion Hardware 34
Psion Operating System 35
Psion User Interface 38
Psion Input 38
Psion Synchronization 39
The Palm-size PC 40
Palm-size PC Hardware 42
Palm-size PC Operating System 44
Palm-size PC User Interface 47
Palm-size PC Input 48
Palm-size PC Synchronization 49
Chapter 3. - Methodology 53
Approach 53
Validity of Data 54
Originality and Limitations of Data 54
Summary 55
Chapter 4. - Analysis of the Problem 57
How We Tested 57
Operating System Evaluation and Comparison 60
PalmOS 60
Psion EPOC32 69
Windows CE 74
Development Tools Evaluation 80
PalmPilot Development Tools 80
Psion Development Tools 83

Palm-size PC Development Tools ... 86
Handheld Applications ... 89
PalmPilot Standard Applications ... 89
PalmPilot Shareware and Commercial Application Highlights ... 90
EPOC32 Standard Applications ... 95
EPOC32 Shareware and Commercial Application Highlights ... 97
Windows CE Standard Applications ... 99
Windows CE Shareware and Commercial Application Highlights ... 101
Newsworthy ... 104
PalmPilot News ... 104
EPOC32/Psion News ... 107
Windows CE News ... 111
Connectivity ... 116
PalmOS Connectivity ... 116
Psion Connectivity ... 117
Windows CE Connectivity ... 120
Chapter 5. - Summary and Conclusions ... 125
Summary ... 125
Operating System Wrap-up ... 126
Developer Tools Wrap-up ... 129
Applications Wrap-up ... 132
Connectivity Wrap-up ... 135
Conclusions ... 140
Appendix A – Interview Ratings ... 143
Appendix B - Handheld PC Comparison ... 147
Appendix C – Handheld Resources ... 149
PalmPilot Development Tools ... 149
PalmPilot Software and Downloads ... 149
PalmPilot Commercial Applications ... 151
PalmPilot Accessories and Add-ons ... 151
PalmPilot Books ... 152
PalmPilot Newsgroups ... 152
PalmPilot Online Publications ... 153
PalmPilot Press Clips ... 153
PalmPilot User Groups ... 155
Psion Development Tools ... 155
Psion Software and Downloads ... 155
Psion Commercial Applications ... 156
Psion Accessories and Add-ons ... 157
Psion Books ... 157
Psion Newsgroups ... 157
Psion Online Publications ... 157
Psion Press Clips ... 158
Psion User Groups ... 158
Windows CE Development Tools ... 158
Windows CE Software and Downloads ... 159

Windows CE Commercial Applications 160
Windows CE Accessories and Add-ons 161
Windows CE Books 161
Windows CE Newsgroups 162
Windows CE Online Publications 162
Windows CE Press Clips 163
Windows CE Palm-size PC Suppliers 164
Windows CE User Groups 164
Index 165
References 168

Chapter 1. - Introduction

Problem

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. There are several types of mobile computing devices to choose from, each with their strengths and weaknesses.

Each class of device has its own architecture and therefore a different set of hardware and software development tools. Each class of device uses a different operating system and provides different levels of software support for such things as Email, Web browsing, internet access and communications protocols. The devices differ in the size, resolution, and color depth of the display, as well as the methods of data input. Some devices provide voice recording and the ability to control the device with voice commands.

The challenge is to choose the device with the correct balance of features that allows the user to be the most productive. Since this means providing applications to take advantage of these features, the system should support a good set of software development tools. We believe that the Windows CE-based Palm-size PC devices offer the best set of development tools.

Purpose

The purpose of this study is to evaluate the three main classes of handheld computers, and to determine which one makes the best choice to use as a platform to deploy solutions to customers. These solutions are predominately vertical solutions to enterprise customers, and very often involve large amounts of custom software to implement.

Importance of Study

This study will help determine which handheld device technology to embrace and subsequently license. The final decision could determine the company's handheld product offerings for the next five years. This study will provide the technical data to support the decision, but there are certainly more factors that will be considered.

Rationale of Study

There are several major classes of handheld systems to choose from. For purposes of this study, we have classified them into three major families of devices, the Windows CE-based systems, the PalmPilot, and the Psion devices. Each platform has its strengths and weaknesses, and it's easy to find individuals or companies that support a particular strategy for the wrong technical reasons. Perhaps someone has used one of the devices for some time and loves it. Or perhaps another user had some bad experiences with one of the platforms. Neither of these reasons provides a technical basis for accepting or rejecting a particular device or class of devices, however. What this report does is to separate the technical details from the emotional ones so that the correct technical decisions can be made.

To keep the focus of this research narrow and focused, we have eliminated from our research those devices that we considered not important to our investigation due to their lack of significant market penetration. These included such devices as the Franklin RexPro [www.franklin.com] and the Royal daVinci [www.royal.com].

Overview of the Study

In this report, we will examine the features of the Palm-size PC, the Psion, and PalmPilot, both in terms of the developer and the end user. We will examine in depth the details of their software, operating systems, memory architecture, user interface design, communications capabilities, and display technology. We will look at the software tools available for each platform, and examine how easy or how difficult it is to write software for them by writing a few sample programs for each platform. We'll take a look at the types of applications available for each class of device, as well as the types of applications likely to be available in the near future. Finally, we will look at current market data and sales figures to see what the future might hold for these handheld systems.

Definition of Terms

API. Application Programming Interface. Defines what external functions are available to a program and how to call them.

Arbitrate. In programming terms, the ability of the device driver to act as a “traffic cop” to grant or deny hardware access to an application.

Abstract (verb). In programming terms, to “hide the details”. A device driver abstracts or “hides the details” of the low-level device architecture from the application software.

Baud. A term used to specify the speed of data transmission in bits per second.

Device Driver. The software program that acts as the interface between an application program and a hardware device. The device driver is responsible for converting high-level requests from the application program into low-level commands that the hardware can understand.

Dynamic Linking. An architecture that allows external references to be resolved at program load time. This results in smaller executables but a slightly longer initial load time.

File System. The internal subsystem in an operating system that controls access to file-oriented devices as well as other hardware devices.

FTP. File Transfer Protocol. A common protocol used to transfer files from one computer to another.

Gnu Tools. Tools including compilers, assemblers, linkers and Make programs written and supplied by the Free Software Foundation and available on many Web sites.

Hardware. The physical electrical components of the computer system or device interface.

HTTP. Hypertext Protocol, the primary protocol used today to send and receive Web pages.

IrDA. Infrared Data Association. An industry-sponsored group that sets the standards for infrared data transmission and reception.

JVM. Java Virtual Machine. A program that is capable of interpreting and executing Java byte codes.

Kernel Mode. A privileged mode of the operating system usually reserved for the operating system itself or device drivers.

OEM. Original Equipment Manufacturer. A company that builds and supplies computer hardware.

Palm-size PC. For the purposes of this report, the class of small, handheld devices that run the Windows CE operating system.

PC. For purposes of this report, an Intel x86-based IBM compatible personal computer.

PDA. Personal Digital Assistant. A generic reference to a small, handheld computer that usually contains applications useful for scheduling one's time, keeping track of expenses, and planning one's day.

Protect Mode. A specific mode of operation for the Intel 80x86 series of processors that provides for hardware-based memory protection and a 32-bit flat memory architecture.

RAM. Random Access Memory. For a handheld device, the temporary memory that holds the user's programs and data. The contents of RAM are maintained only while power is present, either from an AC adapter or batteries. If the power is lost, the contents of RAM are lost. To prevent losing the contents of RAM while the batteries are being changed, most handheld devices have enough residual power to maintain the contents of memory for one to two minutes.

RAS. Microsoft Remote Access Services. Allows other computers and systems to dial in or connect to a particular system.

Real Mode. A specific mode of operation for the Intel 80x86 series of processors that provides a one megabyte memory size with no memory protection. MS-DOS runs in real mode.

ROM. Read Only Memory. For a handheld device, the memory used to store the applications and data that are shipped with the device. The contents of ROM are maintained even if power is lost. When the handheld device is reset after a power loss (and loss of RAM memory), applications and data are loaded into RAM from ROM to reinitialize the device.

Synchronization. For handheld devices, the operation performed between the handheld device and the desktop or server that insures that files and data are up-to-date. Changes to data in the handheld are reflected to the corresponding copy on the desktop or server. Changes on the desktop or server are reflected on the handheld. The data and time stamps on files are generally used to resolve conflicts.

TAPI. The Microsoft Telephony API for Windows. Provides services for connecting with modems.

TCPIP. A connectionless communications protocol used primarily for connecting to networks and the Internet.

UART. Universal Asynchronous Receiver Transmitter. A device to handle the sending and receiving of serial data.

User Mode. A mode of operation where a program runs with the limited ability to access an operating system or attached devices. User mode operation protects the program from damaging other programs or the operating system.

Win32. Microsoft's standard Windows API strategy for the Windows operating systems.

Winsock. The Windows Sockets API. A standard set of APIs for networking in the Windows world.

Chapter 2. - Review of Related Literature

All mobile devices share a common set of constraints that have a direct effect on their usefulness. The way in which the system hardware and software handles or manages these constraints is an important factor in choosing the right mobile device. This does not mean, however, that the system with the longest battery life or the brightest display is the right choice. Each of the features must be evaluated individually and then collectively to determine which system provides the best mix of features.

First, all of the mobile devices, by definition, use some type of battery power that has a limited life. Some of these devices use rechargeable nickel-cadmium or lithium-ion battery modules, while others use less expensive AA or AAA alkaline cells. But whatever the power source, it is clear that maximizing the time that the device can be used before requiring a charge or battery replacement is very important. The battery life, while certainly affected by usage patterns, is managed by the operating system and system support applications, and to some extent, affected by the hardware design. However, applications intended for mobile computing will need to take into account these constraints so the best possible use is made of existing resources [3]. In this type of environment, the most appropriate strategy is to shut off the processor when all the processes are blocked, and turn the processor back on when an external event occurs [20].

A second important factor for mobile devices is connectivity, specifically, the ability to access critical data regardless of location [3]. 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 [45]. Connectivity will be the single biggest differentiator between the universe of devices today and the universe of devices 10 years from today [26].

A third 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 [33].

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 [33]. 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 [4].

The overall goal is to minimize the need for active intervention by users to cope with the consequences of mobility [27]. 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 [8].

[2] By weakly connected, we mean slow connections such as a 9,600 baud cellular connection or intermittent connection to some type of low-bandwidth access point.

Each of the family of devices we reviewed enjoys somewhat of a cult following. While PalmPilot users love their PalmPilots, they tend to be the “least connected” of the users we interviewed. They were generally not corporate users, nor did they use their PalmPilots in a corporate or professional capacity. Rather, they were casual computer users, less interested in serious computing and more interested in scheduling the kid’s soccer games and appointments with the pediatrician. Psion users were concentrated in the United Kingdom, which makes sense since all development and support of the Psion is located there. Windows CE users, however, were more interested in connecting to corporate databases, begin able to get to critical and sometimes confidential information while on the road. While we understand the need for mobile devices in the consumer and end-user markets, we decided to base our study on the best mobile client for corporate and professional users. This particular set of users has more demanding requirements, such as secure data transmission and reception, encryption, password protection, access to sensitive corporate data, and secure e-business transactions.

The PalmPilot

In 1992, Jeff Hawkins fashioned a small block of wood in the shape of a handheld calculator. During the year, he carried that block of wood everywhere, jotting notes on small pieces of paper and pretending to use it to manage his busy schedule. He drew buttons on the wood block, and noted the things that should be displayed based on the small screen area. He counted the number of steps necessary to perform certain functions, and which functions were performed most frequently. As soon as he was comfortable with the functionality, he began an effort to raise the capital necessary to begin production on the new device. In 1996, after months of fine-tuning and rework, Hawkins and his company, Palm Computing®, released the first model of the PalmPilot handheld organizer called the Pilot 1000 [1].

The PalmPilot quickly became a hit with users and developers. Its simple, easy-to-use interface, long battery life and snappy response made it immediately popular with early adopters. The PalmPilot was supplied with several useful applications including calendar, to-do list, and memo pad and address book applications. The first model to be released was the PalmPilot 1000. The PalmPilot 1000 was delivered with only 128 kilobytes of RAM, much too small to be practical. The introduction of the PalmPilot 1000 was followed by a larger, faster model called the PalmPilot 5000. The PalmPilot 5000 had 512 kilobytes of RAM, enough to store an acceptable amount of data. Both had 512 kilobytes of ROM. But the real success came with the introduction of the PalmPilot Professional [21]. The PalmPilot Professional was shipped with one megabyte of RAM and one megabyte of ROM, enough memory to hold some of the most demanding organizer applications such as expense report processing, email and wireless communications[3].

In 1995, the 3COM company [www.3com.com], known for its networking and modem products, purchased Palm Computing [1]. Palm Computing's founder, Jeff Hawkins, was appointed a vice president of the Palm Computing division of 3COM Corporation.

In 1998, 3COM introduced the PalmPilot-III, with two megabytes of RAM and two megabytes of ROM, enough memory to large amounts of data. Third-party developers began shipping applications such as the full text of the King James Bible.

[3] Palm Computing and 3COM were never interested in providing additional features such as paging or two-way communications, but instead licensed the PalmPilot operating system and hardware to Motorola.

The IBM WorkPad

Shortly after the PalmPilot Professional was introduced, IBM saw an opportunity to offer a handheld device to complement their line of mobile computing systems. IBM [www.ibm.com] licensed the PalmPilot Professional from 3COM, painted the case black and renamed the device the IBM WorkPad[4]. A few changes were made to the PalmPilot splash screen, but the WorkPad was otherwise the exact copy of the PalmPilot. In fact, IBM never touched a WorkPad or manufactured any piece of it. The WorkPad was configured and drop-shipped directly from 3COM's manufacturing facility.

[4] In this report, we use the name PalmPilot to refer to the PalmPilot and IBM WorkPad devices.

Palm Pilot Hardware

At the heart of the PalmPilot is the Motorola 68328 DragonBall [http://mot-sps.com/sps/General/chips-nav.html] processor running at a 16MHz-clock rate. The PalmPilot exports the DragonBall's serial port through a small connector on the base of the PalmPilot. The serial port acts as the pipeline for all external connections to the PalmPilot. While a few vendors have designed special add-on hardware for the PalmPilot, the only way to access the PalmPilot externally is through the serial port. The serial port also serves as the physical connection to the PalmPilot cradle from the host computer, providing the communications channel on which to perform the HotSync® operation. The serial port is compatible with a standard PC serial port except that it is not buffered (has no FIFO). Because it is not buffered, the throughput rates are somewhat lower than the serial ports on a standard PC. 3COM will only support rates of up to 65K baud. Faster speeds are prone to bit rate errors and overruns.

The PalmPilot uses a small black and white backlit liquid crystal display with a resolution of 160 by 160 pixels. Officially, the PalmPilot supports only 1 bit per pixel, so the pixel can only be on or off. This means that the PalmPilot has no gray-scale support for the display, making it less useful than other competing products[5]. The screen is touch-sensitive to allow tapping on the screen with the plastic stylus to select programs and menu items.

[5] Recently, 3COM disclosed that the PalmPilot hardware does support up to 4 bits per pixel, but the feature is not yet supported in the operating system.

The PalmPilot has seven buttons on the bottom section of the case that are used to select common functions. There are buttons for To Do, Date Book, Address Book, and Memo Pad applications, as well as up and down arrows and the power switch.

The PalmPilot is powered by two AAA batteries that fit under a small door on the rear of the unit. 3COM claims that the batteries last 8-12 weeks, but in practice we did not observe this. In fact, our testing showed that the average life of the batteries in the PalmPilot was less than 30 days depending on the type of user and the number of times the unit was turned on and off during that period. This is still better than other handheld devices that have a significantly shorter battery life. In fact, everything about the PalmPilot was designed around battery usage. The system shuts itself off automatically to save battery power if it is not used for a minute or two. Every circuit and component was measured and calculated to save battery power. Even the operating system was designed with only two threads of execution so that the processor would not waste a single cycle. The PalmPilot has no charging unit or AC supply, so the PalmPilot uses its share of AAA batteries.

3COM has established relationships with several hardware vendors that supply add-on hardware for the PalmPilot. A company called Symbol Technologies [www.symbol.com] supplies the PalmPilot with a built-in bar code scanner that is used in supermarkets and warehouses for reading the various types of industry bar codes. Several companies, including Motorola, have produced add-on one-way and two-way pagers as well as dial-up connections for web browsing or email. 3COM does not provide any specialized hardware opting instead to license the PalmPilot to other companies that wish to do so.

One item that 3COM does supply and support through the operating system is a modem attachment. The modem snaps on to the bottom of the PalmPilot, connecting to the PalmPilot through the serial port connector. Because it uses the serial port connector, the PalmPilot cannot be used for communications while in the cradle. Using a small utility program in the PalmPilot, users can dial their desktop machine and perform a HotSync over the phone connection much the same as the HotSync operation is performed using the serial connection on the cradle [36]. In fact, the PalmPilot synchronization software is not aware if the synchronization is being done locally through the serial cable or remotely through a telephone connection.

PalmPilot Operating System

The PalmPilot uses a small real time operating system (RTOS) called AMX from Kadak Products Ltd. of Canada [www.kadak.com]. Although the AMX kernel is a real-time preemptive system, the version used in the PalmPilot, called PalmOS, has been modified extensively to provide quick operation and improved battery life. The kernel was modified to include only two simultaneous threads, one for the currently running program's event handling and one for the HotSync application. Applications created for PalmOS cannot be multithreaded.

PalmOS itself occupies only about 300 kilobytes, and uses a single address space [30]. This allows programs to refer to objects directly without going through some type of translation, but it also means that any application can write over any byte of memory and potentially destroy itself. PalmOS stores its data in small databases, and works primarily with small RAM allocations of 64 kilobytes or less called chunks [29]. PalmOS is also capable of executing programs in ROM, eliminating the need to load the entire application in RAM before execution.

To conserve battery life, PalmOS puts the system into a sleep mode if no activity is sensed in a given period of time. While in this state, the system maintains the current contents of memory, but turns off the display. The power to the processor is removed and the processor clock stopped. Power is restored when the user pushes one of the hardware buttons. While the PalmPilot is on but senses no activity, the processor is temporarily halted. This is called the sleep mode. The system exits the sleep mode and enters the running mode when it senses some user input.

PalmPilot User Interface

The PalmPilot uses a non-standard user interface with a limited number of controls. Because of the limited screen size, some of these controls can be difficult to identify, especially in sunlight. To conserve screen space, PalmOS limits the size of program icons making it very difficult to display a meaningful icon as part of a program.

The PalmPilot uses a proprietary user interface specific to the PalmPilot display size and configuration. Applications written for the PalmPilot are closely coupled to the user interface. The basic object for displaying data is called a Form. Data items to be displayed are kept as allocated memory that is part of the Form. Objects that are assigned to the Form come and go with the Form, so data must be saved and loaded with each transition to another Form. Because this relationship is so closely tied to the form and data type, almost no part of any PalmPilot application can be salvaged when attempting to port the application to another platform.

The PalmPilot user interface uses events to notify the application that the user performed an operation such as tapping the screen [31]. PalmPilot events are detected by the system and passed to the application's main input loop. There, an application decides if it should handle the event or pass it on to the system for default handling [23]. Since there is only one application running, it is not possible for a particular application to receive events destined for another application. PalmPilot application programs can (and frequently do) use the event functions to send messages to themselves, simulating user input from the pen or keyboard.

PalmPilot Input

Besides using the serial port, data can be entered into the PalmPilot using a small keyboard displayed on the screen or by using the PalmPilot's built in handwriting recognition language called Graffiti. The user strokes characters with small gestures on the PalmPilot's Graffiti area, and these strokes and gestures are converted into plain text in real time. While some of these gestures mimic the actual writing of the character desired, some do not, making Graffiti somewhat difficult to master. As with any repetitive motion operation, there is always the possibility of straining one's hand or wrist. For those individuals that can't learn Graffiti (or don't want to), PalmOS supplies a pop-up keyboard display allowing users to tap the keyboard keys to select a character. Both of these methods can be somewhat cumbersome if the user has a large amount of data to be entered.

For this reason, it is better to enter the data on the desktop machine or server and have the data transferred to the PalmPilot during synchronization.

PalmPilot Synchronization

PalmPilot applications share their data with a host computer, usually the desktop machine. Data for the applications is sent to and from the PalmPilot via a communications cable connected between the host computer and the PalmPilot. The PalmPilot is placed in an inexpensive holder referred to as the cradle, which not only holds the PalmPilot stationery but acts to connect the PalmPilot to the communications cable. The cradle contains contacts that connect the serial cable to several small contacts located at the base of the PalmPilot.

Data that resides on the PalmPilot and the desktop is synchronized using a program stored in the desktop system and a companion program located in the PalmPilot. Changes in desktop data are reflected on the PalmPilot, while changes to data in the PalmPilot are also reflected back on the desktop. This basic form of data sharing or synchronization is a fundamental element of all handheld systems. The PalmPilot user presses a button located on the cradle to initiate the synchronization. On the PalmPilot, this operation is called the HotSync operation. Synchronization is the process of merging two databases: one on a PC and one on the Pilot. The HotSync application manages this process, and calls on special programs called conduits to perform the actual data manipulation for each database to be synchronized. There is one conduit assigned to each unique pair of PC and Pilot applications (Stack).

The HotSync operation transfers data in both directions at up to 65K baud, although the actual throughput is much lower[6]. It should be noted that synchronization is used to transfer both programs and data. In fact, the same mechanisms are used to transfer programs to the PalmPilot as well as data[7].

The PalmPilot HotSync program allows the user to customize flow of data between the desktop and the PalmPilot. The user can select Synchronize the Files, Desktop Overwrites PalmPilot, PalmPilot Overwrites Desktop, or No Operation. These settings give the user a great deal of flexibility in customizing the HotSync operation. In practice, however, most users never use these features. One drawback of the PalmPilot synchronization architecture is that the HotSync operation must be initiated manually by the user. If the user fails to initiate the HotSync operation, the data in the PalmPilot will not be transferred to the desktop system, and data on the desktop system will not be transferred to the PalmPilot. If the PalmPilot is dropped or the batteries die, the data stored in the PalmPilot is lost.

[6] Even with the baud rate set to 65K baud, the serial link protocol kept the aggregate throughput to under 19.2K baud.
[7] In fact, programs and data are both stored as databases in the PalmPilot, but each as a different type.

The PalmPilot synchronization is optimized to provide the highest throughput to compensate for its poor serial performance. Databases stored on the Pilot are kept with a bit for each field in the database. During program operation, any changes made to a particular field will cause the dirty bit for the field to be set. Later during synchronization, the bit for each field is checked to see if the field has been changed. Only fields that have been changed are transferred to minimize synchronization time.

The Psion

Psion released its first electronic organizer, the Organiser II in 1986. The Organiser II sold over one million units during its lifetime. It featured an 8-bit CPU that was arguably well ahead of its time. The Organiser II could be programmed in assembly language or a new high-level language called OPL for Organiser Programming Language. This BASIC-like language allowed novice programmers as well as professionals to write programs for the Organiser II.

In 1991, five years after the introduction of the Organiser II, Psion [www.symbian.com] introduced the Psion Series 3; a new handheld organizer based on a 16-bit microprocessor. Because it was based on a 16-bit processor, the device was named SIBO, for 16-bit organizer. Psion believed that the introduction was so revolutionary that they named the new operating system for the Psion 3 EPOC. The EPOC32 name has remained with operating system for the entire Psion line of handheld devices including the newly introduced Psion 5.

Psion Hardware

The Psion Organiser II and Series 3 handheld systems were both based on the Intel x86 architecture. Although this made Psion dependent on the x86 processor, it allowed thousands of third-party applications to be written for the Organiser II and Psion 3. Over 1.5 million Series 3 machines have been sold [11]. In 1997, Psion introduced the Psion Series 5 systems based on the Advanced Risc Machines ARM7100 32-bit microprocessor[8]. To conserve battery life with this powerful processor, the clock rate was set at a fairly slow 18.432MHz, and the size of ROM capped at six megabytes[9]. Instruction size is fixed at the standard RISC size of 4 bytes per instruction [22]. The memory path was also decreased in size to save power. The Series 5 comes equipped with a serial port and serial infrared transceiver.

[8] The ARM processor core is actually embedded in an ASIC package with memory and appropriate glue logic.
[9] The ARM7 processor is capable of running from DC to 200MHz.

Psion Operating System

The Psion 5 system is powered by the 32-bit version of the EPOC operating system called EPOC32. To build the operating system, Psion spun off a separate company called Psion Software in 1996. The new company's mission was to write a new 32-bit operating system for the Psion devices, as well as to provide a portable, 32-bit operating system for other types of processors. Psion wanted the EPOC system to become an open system that it could license to other OEMs. After the release of EPOC32, Psion Software became a separate company called Symbian, Ltd [www.symbian.com] that is owned by Psion, Nokia, and Ericsson.

EPOC32 was designed from inception as a full 32-bit operating system with no 16-bit code. It is written primarily in C++, although a few small sections are written in assembler. EPOC provides a standard C runtime library that provides a POSIX-like, non objected-oriented layer over the many base and communications functions. The foundation of EPOC is object-oriented, surrounded by non object-oriented layers. This is the reverse of Microsoft's MFC, which supplies object-oriented layers over non object-oriented code. Programmers can use C++, Java, or Psion's legacy language, OPL, to write programs for the Series 5. There is also a set of Gnu tools available to write code for the Psion machines.

EPOC32 is a full function operating system, with many of the features found only in the most powerful desktop operating systems such as Windows. EPOC uses the thread as the instance of execution. A process is a collection of one or more threads. The thread priority can be modified to achieve the desired result. Threads share the priority of the process, so increasing the priority of the process causes an increase in the priority of the threads associated with that process. Each thread contains its own heap and thread-local storage. EPOC32 supports full memory protection for processes and threads.

EPOC32 provides support for shared resources, which are controlled by mutex semaphores. The EPOC32 kernel runs in a privileged mode while applications run unprivileged. EPOC32 uses servers to provide services to applications and to other servers, thus is designed as a client-server architecture. Using these servers, applications can access system resources while in unprivileged mode without having to switch to a privileged mode or use expensive mutex semaphores [11]. The kernel server synchronizes access to kernel-owned resources such as processes, allocated memory and timers[10]. A single thread owned by the operating system acts as the idle thread, and is run when no other threads are waiting to run [13]. When the idle thread executes, it puts the processor in a low-power state to conserve battery power of the two AA cells.

[10] This message-passing architecture is the type of architecture usually found in high-end operating systems such as Mach.

EPOC32 contains a file system server capable of manipulating drives, volumes, directories, and files. It can support up to 26 logical disk drives per memory device, usually a compact flash card. Device drivers are architected with an in-kernel and out-of-kernel section, much like the layered device driver models of modern operating systems. The top section provides access to the device from user mode. The lower section provides direct hardware support including interrupt and timer handling, and passes those events up the user mode section as a message. This type of message-passing can be detrimental to performance, so EPOC32 uses an extremely lightweight form of message passing to minimize interrupt and timer latency [40]. Because this architecture does impact interrupt latency, EPOC32 provides the Delayed Function Call, or DFC to put off processing while continuing to service interrupts. This service, akin to the Windows Delayed Procedure Call, or DPC, allows processing of the data to occur when interrupts have subsided. This insures that no interrupts are missed while processing large amounts of data.

EPOC32 contains a stream store for managing data streams. The stream store facilitates the implementation of a database management system, which uses a permanent file store to provide a relational database. The database can contain an arbitrary number of tables, and supports **commit** and **revert**. The revert feature provides the ability to recover from an operation that was interrupted, and is similar to a transaction-oriented database program's **rollback** capability.

Psion User Interface

The Psion user interface is based on a proprietary graphical user interface engine called EIKON. Because EIKON is designed as a user mode service, it can be easily removed for those systems that don't require a user interface, such as telephones and embedded systems. EIKON can also be removed and replaced by another user interface without affecting applications or the rest of the EPOC32 system. Applications call an EIKON DLL that calls the user interface component. If the user interface component changes, the application still writes to the same DLL as if the user interface component was unchanged. The user interface supports input from the both the pen and the keyboard. EIKON supports the zooming of the displayed data to suit the lighting conditions, and includes the standard toolbar and toolband components [11].

Psion Input

Input to the Psion is provided via the touch screen or miniature QWERTY-like keyboard. Most operations require a double-tap to execute. There are nine reserved touch spots for System, Word, Sheet, Data, Agenda, Time, Calc, Sketch and Extras. The touch spots are simply reserved areas on the 5.5" x 2.25" liquid crystal display, which are silk-screened with icons that identify the operation to be performed. In addition to these dedicated areas along the bottom of the display, the Psion has five functions on the left side of the display for zooming in and out, controlling the audio volume, cut and paste, and file menu.

Psion Synchronization

The Psion uses its own product called EPOC32 Connect to perform synchronization between the handheld device and the desktop system. EPOC32 Connect runs on Windows 95, Windows 98 and Windows NT. EPOC32 Connect provides integration into the Windows Explorer and desktop for easy control of EPOC32 machines attached to the desktop system. EPOC32 Connect performs file transfer, backup and restore, and can handle multiple clients. Several standard database formats are supported such as Microsoft Word and Lotus WordPro. EPOC32 Connect allows users to print from any EPOC32 application using a printer attached to the desktop system.

EPOC32 Connect abstracts the details of the database format by providing a library of synchronizer and converter engines, which are accessed through a common object model (COM) wrapper. By using the COM interface, developers can write these components in C, C++ or Visual Basic. Programmers can configure converters, filters, and synchronizers quickly without having to rewrite the code each time. The lower level transport uses Psion’s Psion Link Protocol, or PLP to provide connections, RPCs and other types of services. It has a full TCP/IP stack and network interface components.

The Palm-size PC

Microsoft began building the software for a handheld computer in 1992. The effort was focused around two products; a small, pen-driven device dubbed the WinPad and a set-top box called Pulsar for an interactive television system. The Pulsar operating system was a new object oriented operating system built around wireless communication networks [26], while the WinPad operating system was based on the old Windows 3.1 architecture. Both projects were eventually killed. Meanwhile, a small group in Microsoft created a new operating system that could run on various types of handheld and embedded systems to fill the void left by the cancellation of the WinPad and Pulsar projects. Dubbed Pegasus, the operating system had a user interface that looked liked Windows and supported a large portion of the Win32® application programming interfaces. The system was designed to be modular and to work on many different platforms.

In 1996, Microsoft released the first version of the new operating system with limited success. In 1997, version 2.0 was released on a wide variety of handheld devices. One of the most popular units to sport the new operating system was the Hewlett Packard 620LX. In 1998, several other hardware vendors including Casio, Compaq, Everex, Hewlett Packard, Hitachi, LG Electronics, NEC and Philips shipped version 2.0 products. Version 2.0 included many new features including support for color displays, high-speed communications, voice recording and playback, voice control, and higher resolution displays. At the 1997 Spring Consumer Electronics Show, Microsoft unveiled plans for a new Windows CE-based system for the automobile called the AutoPC. Consumer electronics manufacturer Clarion has signed up to provide the first of these units, which feature voice command and control.

Palm-size PC Hardware

To insure the success of the Palm-size PC, Microsoft and its hardware partners got together to specify the hardware platform on which to run Windows CE. Microsoft specified a minimum configuration, but allowed each manufacturer to add other capabilities and features [5]. Some of the specifications are:

- A pocket-sized form factor not to exceed 7” x 4” x 1”.
- Powered by two AA batteries.
- Weight less than one pound.
- A QWERTY keyboard in hardware or software
- An LCD touch screen with at least 480 x 240 with 4 grayscales and 2 bits per pixel
- A stylus that is used like a mouse
- A minimum of four megabytes ROM and two megabytes RAM
- An infrared port
- A serial port

- A PCMCIA slot
- A built-in audio device

Windows CE currently supports the Hitachi [www.hitachi.com] SH-3 and SH-4, MIPS 4101 and 4111, the Philips TwoChipPIC, and Intel x86 RISC processors [38]. Just recently, Microsoft announced Windows CE support for the ARM, StrongARM and PowerPC processors. Moving Windows CE to other platforms should be easy due to the modular design of Windows CE.

The Palm-size PC specifications require a minimum of four megabytes of ROM to hold the operating system and bundled applications. The minimum configuration of two megabytes of RAM is used to hold data and running programs. Windows CE automatically detects a change in memory size, so inserting a memory device causes the extra memory to be immediately available. Palm-size PC displays have a resolution of at least 480 x 240, and 2 bits per pixel for 4 gray scales. The backlit LCD panel is touch sensitive.

Palm-size PC Operating System

Windows CE is the operating system that runs on the Palm-size PC, and represents Microsoft's third effort to provide a slimmed-down operating system suitable for embedded and mobile devices. The first efforts to produce this operating system were based on the requirements for a small operating system for personal digital assistants, or PDAs. This new operating system was called Pulsar. Another operating system was being developed in parallel with Pulsar for a device called the WinPad. Pulsar was a 32-bit operating system based on object oriented technology, while the WinPad operating system was a 16-bit system based on the old Windows 3.1 APIs. While both groups were debating their future, a small team began writing a brand new operating system that would satisfy the needs of each group. Eventually, the Win32 API was added to the operating system. The operating system concentrated on reducing the overhead of system functions, especially the inter-process communications or IPCs while still including the Win32 APIs. Windows 95 could not be used because it didn't fit the Win32 model, and Windows NT was just too big and slow. So after considerable tuning, the new operating system was accepted by Microsoft management. Microsoft called the new base operating system **Nk** for New Kernel.

Windows CE is written almost entirely in C, with a small amount of assembly language for each specific processor platform. Microsoft supplies the processor-specific code for each supported processor, so original equipment manufacturers, or OEMs, don’t have to write any code to use Windows CE on their particular processor.

Windows CE supports preemptive multitasking using threads. It provides preemptive priority-based scheduling based on the Win32 process and thread model [25]. To prevent thread starvation, Windows CE dynamically adjusts thread priorities and provides round-robin scheduling for threads of the same priority. Windows CE provides a set of synchronization objects normally found in larger operating systems such as mutex semaphores, critical sections, wait objects and named event objects. Microsoft spent considerable time tuning these components to provide deterministic operation. Memory is based on a 32-bit flat model, with 32 megabytes allocated to each Windows CE process. Windows CE can support up to 32 simultaneous processes, and the number of threads is limited only by available physical memory [32]. Like Windows 95 and Windows NT, Windows CE provides most of the system services in reentrant, shareable dynamic link libraries, or DLLs. This allows applications to take up less space and make more efficient use of memory [6,7,12].

Windows CE implements full memory protection, and provides a form of demand paging for RAM and ROM. The heap size in Windows CE is limited only by the size of the available memory. The kernel handles mapping from virtual to physical addresses. Memory protection violations, such as writing outside processes' address space, will generate an exception. Processes can handle those exceptions by registering an exception handler for the process.

Windows CE allows memory mapping, which permits multiple processes to share the same physical memory. Memory mapping results in very fast data transfer between cooperating processes, or between a driver and an application [25]. Approximately 1 GB of virtual address space, distinct from that used for the program storage, is allocated for memory mapping [24].

Windows CE allows the programmer to install and support their own interrupt handlers and interrupt service threads [19]. Interrupt latencies are consistent and deterministic. Windows CE supports FAT volumes and uses transactions for all file system operations so that operations can be undone in the event of a failure. Block device drivers can be installed to support block devices such as additional flash RAM or disk drives.

The Windows CE system supports full TCPIP, FTP, HTTP and Winsock 1.1 sockets. Windows CE also has built-in support for IrDA infrared devices, point-to-point protocol, or PPP and remote access services or RAS. It also has a full set of telephony or TAPI APIs to provide support for modem connections [18].

Palm-size PC User Interface

The Palm-size PC user interface supports a keyboard and stylus as input devices. The graphics engine provides support for color and grayscale display devices with up to 32 bits per pixel of color depth, palette management, TrueType and raster fonts, and advanced features such as shape drawing and bit block transfers. The graphics subsystem supports the full UNICODE standard, allowing Windows CE devices to be easily modified for non-United States markets.

The user interface architecture is based on the familiar Windows-style message interface. Threads communicate with the main program or other threads using several methods including sending Windows messages. Threads can also use memory-mapped files, a method for fast data transfers between processes and threads. With memory-mapped files, the data to be shared is written to or read from a file object that resides in memory. Access to the file object is very fast because the data is stored in memory and not on a physical disk.

Palm-size PC Input

The touch-sensitive screen and a QWERTY-like keyboard provide input to the Palm-size PC. Some Windows CE-based handhelds and palmtops have a reduced size keyboard with the functionality of a full-sized keyboard. Other Palm-size PCs, specifically the smaller, palm-sized units use a software-controlled keyboard. The operating system displays a keyboard image on the display screen and the user taps each key to simulate an actual keystroke.

Most Palm-size PCs have some method of voice command and control as well as voice recording. Recording is performed at a low frequency, 3-6KHz, to save memory space. The voice data is stored in PCM format for later playback[11]. The Palm-size PC stores each recording as a file, and most can hold about 63 minutes of recording. In addition to voice recording and playback, the Palm-size

[11] PCM is an analog footprint of the voice input. Although it can be played using a sound device, the actual text of the recording cannot be recreated.

PC can be configured to respond to voice commands. The Palm-size PC allows the operator to record from 30 to 50 commands that are linked to user events.

Address book entries can be looked up using the voice command. There is no actual conversion of the voice input to text, rather the name is entered and the operator records some phonemes that are linked to the name entry. When the name is spoken, the Palm-size PC address book looks up the correct name by the sound of the recording, not by the actual text.

Windows CE uses the JOT character recognition software that provides recognition of natural characters instead of a special set of stroked characters like the PalmPilot [10].

Palm-size PC Synchronization

The Palm-size PC provides automatic, unattended synchronization of desktop and Palm-size PC data with the ActiveSync software. ActiveSync can synchronize any type of data with the desktop and the handheld PC. Companies that supply applications where data can be shared between the desktop and handheld supply a small portion of the synchronization code specific to their data format. The synchronization code is layered, so only a small portion is unique to the application. This allows synchronization programs to be written very quickly.

Connections are made from the handheld to the desktop at a default speed of 19.2K baud. This speed was chosen to provide compatibility with older PCs that did not have the ability to go above 19.2K baud. Some older PCs used non-buffered UARTs that could not reliably receive at speeds greater than 19.2K baud. If desired, the speed can be increased to 57.6K baud but Microsoft cautions users to experiment to find the optimal settings for their particular system configuration.

When the synchronization software detects a conflict in the handheld and desktop database, it displays a window on the user's PC to resolve the conflict. ActiveSync allows the user to customize the synchronization settings so that the conflicts do not have to be resolved manually.

When transferring some types of data from the PC to the handheld device some attributes are affected since the program on the handheld is generally less capable than the program on the desktop. For instance, transferring a Word file from the desktop to the handheld device will cause pictures to be converted to 4-color bitmaps and fonts to be converted into one of the standard Palm-size PC fonts. Likewise, when transferring data from the handheld to the PC, some attributes are changed. For example, 4-color bitmaps are converted to 16 colors.

Chapter 3. - Methodology

Approach

Our approach was to first examine the features of the Palm-size PC, the Psion, and PalmPilot, including the operating system, memory architecture, user interface design, communications capability and display technology, and how each of these affected the usefulness of the device. Next, we put together a set of tools for each platform and evaluated them on usefulness, functionality and price. We used these tools to write a sample program for each platform, noting how long it took for an average programmer to complete the task. We looked at the various types of applications available for handheld systems that make them desirable for the mobile user. We explored some actual case histories of companies that have rolled out handheld systems to their employees. We reviewed several scenarios and examined how each scenario would be carried out with each type of device, and attempted to determine which device was best suited to the task. We interviewed several users to find out what they felt were the most useful or compelling features of their handhelds, where they fell short, and what features they would like to see in the next generation of handheld devices. Finally, we examined current market data and sales forecasts to determine which of these devices is likely to still be around in a few years.

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 judgement given the facts presented here. It is quite possible that others may form a different opinion or view based on the evidence.

Originality and Limitations of Data

The data presented here represent a snapshot in time. Technology is changing rapidly, and it is possible that by the time this report is finished that a new competitor will have entered the handheld field. Selecting a particular handheld platform may make sense at the time, but like all major decisions of this type, the decision should be continually reviewed and evaluated using current technical and market data.

Summary

Based on the data we were able to gather as well as our actual experiences with these devices, we are convinced that the Palm-size PC represents the best platform on which to develop and deliver mission-critical mobile applications. The hardware and software architecture, availability of tools, functionality and affinity with the Windows operating system make the Palm-size PC the only logical choice for enterprise-worthy applications.

Chapter 4. - Analysis of the Problem

How We Tested

In the first part of our evaluation, we examined the features of the operating system which make it more or less likely to be a point of failure, cause bad performance, or make it difficult to write programs for. We believe that mobile computing will demand more and more from the mobile devices, adding such features as secure transactions and mobile access to corporate databases. For these types of enterprise applications, we wanted to be sure that the operating system had the built-in features that might allow the system to be used in a sensitive corporate market. In particular, we looked for some of the following features:

- Secure TCP/IP communications
- Memory protection
- Large program size
- Data compression
- Easy-to-use user interface design
- Network connectivity and protocols
- User logon and password protection
- Good display size and quality
- Support for industry standard peripherals
- Ability to share data with popular desktop applications
- Ergonomics, size, weight
- Battery life

Next, we took a close look at the tools available for writing software on these devices.

We then took a look at the types of applications that ship with the handheld device and those applications that are available from third parties.

We searched the news archives for noteworthy announcements from the major suppliers that might influence a company's decision to deploy one of these handheld devices. We then searched for as much data as we could find on companies that had already supplied handhelds to their employees, and what the results were. We acknowledge that this data may be incomplete because the handheld PCs are still in the early stages of adoption and implementation.

Next, we examined the connectivity features of these devices and the options they offered. We believe this to be very important since the handheld PC relies heavily upon being connected in some fashion to get data.

Next, we evaluated each device to see how easy it was to use. We picked a simple operation where the user enters a contact in the their address book and an appointment for the next day. We had each user note the time it took them to perform the task, and how easy they found the task to complete.

Finally, we gathered as much market data and forecasts to try to sort out what the market was doing and where it might be headed in the future.

Our conclusion is not based on any one feature alone, but on a combination of all of the features we reviewed.

Operating System Evaluation and Comparison

PalmOS

PalmOS is a slimmed-down version of the AMX real time operating system from Kadak, Ltd. of Canada. The AMX operating system is designed to be processor independent, and has been ported to the 680xx, PowerPC, ARM, x86, R3000, Z80, and I960 microprocessors. What makes PalmOS special is that the operating system was specially configured for the DragonBall microprocessor and the PalmPilot hardware. Although AMX is a thread-based operating system, the version of AMX in the PalmPilot has only two running threads. One thread is used for the event handler of the current application, while the second thread is used to run the HotSync® program. There is no notion of any real-time operation or any type of background task. Palm Computing elected to configure AMX with only two threads to conserve battery life. With only two threads to support, the amount of power needed to run the PalmPilot was kept to a minimum.

Each task monopolizes the processor and the application's event handler. All events that occur, such as pen taps or button pushes, are routed to the currently running application's event handler and the *Form* within that application. There is only one logical event handler that consists of the application's event handler and the system event handler. Events appear at the application's event handler, and the application can decide whether or not it wants to handle the event. If not, it passes the event to the system event handler. This design requires that each application provide the proper event linkage to the operating system. If an event does not occur in a specified time (default of two minutes) the processor is put in a "sleep" mode where it actually stops running. When an external event occurs, the occurrence of the event causes the processor to begin executing.

Events that occur in PalmOS are routed through the application's event handler. Each application must install its own special event handler, as the system calls the application's event handler for each event. If there are no applications active (except the PalmOS default application or idle loop) the system handles all external events. When an application is loaded, the entry point of the program's event handler is passed to the operating system, which calls it when an event occurs. If the application's event handler fails or does not return to the operating system, the PalmPilot will cease to respond to events and will require a reset.

The PalmPilot application "signs up" for events when it is loaded. When the event occurs, the PalmPilot application's event handler is called. If the application wishes to handle the event, it returns TRUE to the system, indicating that no further action on the part of the operating system is required. If the application's event handler returns FALSE, the system continues to handle the event as if there were no applications present. Using this architecture, the application can intercept certain (or all) system events and act upon them before the system gets a chance. The events can be filtered, ignored and even translated into other types of events.

PalmOS applications are designed around the event-driven architecture of the PalmPilot. Since there is no notion of foreground or background tasks, everything must be done in the current thread. There can be no polling or timing loops in the application, nor can there be any processing of data in parallel with the handling of the events. Everything that needs to be done must be executed in the context of the event handling architecture. If an operation requires processing for an extended period of time, the system becomes unusable during that time. Unlike other threaded operating systems where operations can be performed while still handling events, PalmPilot events do not get processed. The system does appear to buffer one event while processing, and executes that event once the processing has completed.

Programmers design PalmPilot applications by creating display "windows" based on a container called a *Form*. The Form acts as a holder or container of all the objects that exist in the form. The Form is actually an object with links to the objects contained within the form. The objects may be buttons, checkboxes, fields, bitmaps, gadgets, labels, lists, popup triggers, push buttons, scroll bars or tables. The PalmPilot programmer assembles these objects together in a form using a visual placement tool such as the CodeWarrior Constructor. The program must then fill the desired objects with data, display them, and wait for events to determine what to do with those objects. If the program needs to show a different screen, the current form calls the operating system to load the new screen form and all of its objects. When that new form is done, it then calls the operating system to load the next form.

The PalmPilot stores data and applications in compact, memory-resident databases. The only difference between data and applications is the database type. Each database has a special field called the CreatorID that specifies the creator or owner of the database. When a user taps an icon or button to launch a PalmPilot application, the operating system searches the database for a program with the requested CreatorID and loads it. Data is also stored in the PalmPilot database. The data can be application resources such as strings and bitmaps, address book data or an executable program. The PalmPilot makes no distinction as to the data, only the database type. PalmPilot databases can be compacted and compressed, and PalmPilot applications make good use of this feature. In the front of each PalmPilot database record is a 32-bit field that acts as a bitmap to describe which database fields are present. If a database record can contain 32 fields but only has data in three of the them, the bitmap can be set to notify the reader of that fact, and the data can be stored efficiently by eliminating empty fields. The standard PalmPilot database record also contains a

4-bit field in the front of every record that is used for categorizing the data. The 4-bit field yields up to 16 distinct categories to be used for sorting and display. The PalmPilot operating system supports this 4-bit field with several systems APIs to handle category operations. These APIs are highly optimized to perform fast sorting operations, and rely upon the special sorting field format.

The architecture of applications written for the PalmPilot virtually guarantees that none of the code written for the PalmPilot can be used anywhere else. Since the operation of the PalmPilot is so closely coupled to the operation of the event manager, the software can’t be used on any other platform without major changes or a total rewrite. There is some evidence that programs written in Java may get around this limitation, but those programs would be successful only if the PalmPilot had a reasonable Java Virtual Machine, or JVM. Even if the JVM was well written, the speed of the PalmPilot’s processor becomes a limiting factor. With the processor running at only 16MHz and with a narrow data path, the PalmPilot barely provides acceptable performance even for simple games.

The PalmPilot works well when it does what it was design to do – display calendars and small memos. The PalmPilot was not designed to do any strenuous data processing tasks, handle large amounts of data, process voice input or handle high bandwidth communications. The PalmPilot is just an organizer with connections to a desktop system that handles the heavy work and storage. Forcing the PalmPilot to do anything more will lead to a disappointing outcome. In one particular instance, we wrote some PalmPilot code to sort 500 records, each 300 bytes long, based on a four-byte string located in the record. The operation took over two minutes, during which time the PalmPilot would not handle any user input of any type. We then changed the sort routine to use one of the PalmPilot database category bits, part of a four-bit field at the beginning of every record. The sort time was reduced to approximately 30 seconds, still somewhat undesirable for a commercial application. But these are not the kind of tasks that the PalmPilot was designed for. In fact, for a much smaller number of records, 100 or less, the PalmPilot is snappy and provides acceptable performance.

Since the heart of the PalmPilot is the event manager, we spent a lot of time with the manual that described events and their parameters. We also spent a great deal of time with the system runtime library manual trying to figure out why a simple application we wrote was not working correctly. Normal C-language runtime libraries include such standard functions as printf, scanf, and memset. Unfortunately, these functions can't be used on the PalmPilot. Instead, programmers are forced to use special PalmPilot functions such as MemSet and StrCat. The PalmPilot libraries are non-reentrant, so multiple clients cannot call the library routines or share the libraries. There is no code sharing, so runtime libraries are linked at application load time to the single application that has focus.

PalmOS supports only a single address space. There is no memory protection between programs, and applications are free to read from or write to any hardware or memory address without any type of protection. However, because the application does not require memory protection, it can be loaded very quickly. It can be loaded even faster if the program is compiled and linked in the compact mode. In this mode, the code and data for the application resides within one 64-kilobyte chunk. If the program is small enough, the system will also allocate local storage in the same memory chunk to minimize access to variables.

Some PalmPilot system functions reside inside PalmPilot libraries. A PalmPilot library is also a database of a special type, and the library is loaded by calling a system API. This allows libraries to be loaded as needed, just like Windows DLLs, to save memory until the code is actually needed. Unlike their Windows equivalent, however, PalmPilot libraries are not loaded as a result of a memory fault. PalmPilot libraries must be explicitly loaded by the application that will use the library. PalmPilot libraries are non-reentrant. The library is not linked with the application, as the entry points are resolved at library load time.

PalmOS has no device driver model or interface. The system handles pen taps, stylus swipes and button presses by generating events for them. The PalmPilot application's event handler must intercept the events if it wishes to handle them itself. Each event is generated with a structure that gives more information about the event, such as the type of key that was pressed or the state of the stylus at the time of the event.

The only external device on the PalmPilot is the serial port. The serial port is accessed by a handful of APIs that in turn call a library of lower-level functions in the PalmPilot serial library. Since there is no concept of a background task in the PalmPilot, reading and writing asynchronous serial data with the PalmPilot can be a chore. The application must continually handle events generated by the operating system yet still be able to read data from the serial port fast enough so that no data is lost. In what we feel was a really foolish move on the part of 3COM, the important APIs that would allow you to perform simulated background serial communications are left purposely undocumented. A clever programmer can read through the header files and with some experimentation, get most of them working. This seems to be a waste of time, and should be corrected.

The PalmPilot does a good job of conserving battery life. It seemed like whenever we took our eyes away from it the PalmPilot would turn itself off. There is a system parameter that a user can set to lengthen the time before it turns itself off, and this is the first parameter we changed. Although 3COM claims that the batteries last for three months, we found that with average use of an hour or two per day that the batteries need to be changed in less than two weeks.

Currently, 3COM does not officially support a Java Virtual Machine, or JVM, on the PalmPilot. There are, however, a few JVMs that have been written for the PalmPilot. We did not test Java on the PalmPilot since the JVM is not a commercial product or part of the standard shipping software. We also feel that PalmPilot is highly under-powered, and running interpreted Java code on a system that is essentially a lightweight appointment viewer may be a cool research project but lacks any real value.

The PalmPilot is very limited in terms of the display area and visibility. With only 160 by 160 pixels and no officially supported gray scale support, we found the display very difficult to read. Programs designed for the PalmPilot have to be cleverly designed to be able to display the right amount of information on the PalmPilot’s tiny screen. Icons are limited to 16 by 16 pixels, not much to work with to depict a program’s purpose. Icon designers have to work hard to come up with a clear, helpful icon design.

Psion EPOC32

Psion's EPOC32 stands out as the most advanced operating system we found in the three classes of handheld devices we tested. EPOC32 was designed to run on a 32-bit RISC microprocessor like the PowerPC or ARM7. Its message-passing architecture is of the type normally found on much larger machines such as 680xx workstations. In many respects, EPOC32 looks a great deal like the Carnegie Mellon Mach operating system. Like EPOC32, Mach also relies on the client-server model of user-mode applications and kernel-mode applications. Applications that wish to perform a certain task call a user-mode or kernel-mode service. The idea behind this architecture is simple and obvious, in that the services abstract the underlying detail of how the service gets its work done. Thus the server is a block box which can be changed without affecting the clients, providing that the interface does not change. Another benefit is that services in general can be easily replaced or removed without impacting the rest of the system. In an embedded operating system like EPOC32, this is important because it allows the EPOC kernel and support components to be used to build an operating system in a "building block" fashion. OEMs or ISVs who don't wish to use a particular user service are free to architect their own service. For example, a complete operating system can be built and configured without a graphical user interface. ISVs can also write their own graphical user interface service and replace or augment the functions of the original service. The ability to provide “plug-in” services is a powerful feature of EPOC32.

Because these services run in user mode, they are easier to debug and provide faster service since mode transitions are required. However, services that run in user-mode generally run slower than in kernel mode.

One of the problems with the user-mode and kernel-mode services is that there is a performance price for using them. It takes time to send the messages back and forth between clients and servers, especially if the messages need to cross the user-mode and kernel-mode boundaries. Since kernel-mode and user-mode programs operate in different address spaces, transferring data between them becomes somewhat problematic, requiring time-consuming pointer conversions and address mapping.

Another problem with operating in both user and kernel mode is the handling of time-critical data and interrupts. Hardware devices operate asynchronously, so data can appear with no prior warning. The kernel-mode software is responsible for handling critical, time-related functions in an efficient manner. It accomplishes this with special kernel-mode programs called device drivers. Device drivers usually contain some type of high-speed data handling functions in routines called interrupt handlers. The interrupt handler is responsible for handling interrupts from hardware devices, indicating such events as a character received or a disk read completed. Because they need to access system hardware directly, interrupt handlers are generally written to run in kernel mode. When the interrupt occurs, the driver must handle it quickly and get ready for the next interrupt[12]. If the interrupt was caused by incoming data, the data must be retrieved and passed up to an application that can use it. This means that the interrupt handler must send the operating system a message to indicate that an interrupt has occurred. When the system receives the message from the device driver, it must notify the waiting application that there is data to be processed by sending a message to the application[13].

It is possible that interrupts can be generated faster than the interrupt handler can handle them, so EPOC32 (as well as Mach and Windows) provides a mechanism to handle this problem. The system provides a separate thread that can be used to process the interrupt outside the context or scope of the interrupt handler. This function is called a Delayed Function Call, or DFC[14]. The interrupt handler can call a DFC to perform operations that take more time than the interrupt handler can afford. This insures that the interrupt handler is ready for the next interrupt and that the system performance remains relatively unaffected.

---

[12] An interrupt may occur even while the current interrupt is being serviced.
[13] This is a common architecture, so most systems provide a lightweight message for this.
[14] In Windows it is called the Delayed Procedure Call, or DPC.

Because of the message-passing architecture of EPOC32, we expected performance of the Psion to be less than optimal, but we were pleasantly surprised by the results.

EPOC32 is written for the 32-bit RISC architecture. This means that all instructions are completed in one machine cycle, even though instructions are four bytes long[15]. Using a 32-bit RISC processor in a small package like the Psion creates a problem, however, since there is no room for the large number of pins required for 32 discrete address lines, 32 discrete data lines, and all of the other signals required. To get around the physical limitation, the ARM7 in the Psion uses only eight address lines, and multiplexes them together to form a full 32-bit address. It does this by fetching one byte at a time and building the address. However, each access would normally take one machine cycle, and that would slow down the processor by a factor of three. To overcome this limitation, the ARM7 processor uses a three-stage pipeline so that all of the bytes for each instruction are in the processor's pipeline at the time of instruction fetch. This keeps any performance degradation to a minimum.

The Psion user interface architecture provides a rich set of graphics functions and support for various types of graphics devices, contexts, fonts, color and measurement functions. The Graphics Device Interface, or GDI, provides base classes for printers and display devices. GDI is built as a service, and can act as

[15] 32-bit RISC processors such as the ARM7, PowerPC and MIPS, execute instructions in one cycle.

a server to any application that wishes to print or use the display screen. It handles the sharing of the screen surfaces between multiple applications by requiring applications to write to logical windows, while GDI handles the actual physical writes to the display. GDI also controls access to the keyboard, buttons and pointer. A higher-level GUI called EIKON can be layered above GDI, providing standard dialogs and controls and a consistent look-and-feel between applications. OEMs that don't require a high-level graphics layer, perhaps because the system has a modified display device or is used as an embedded 'black box', can call GDI directly.

EPOC32 supports Java 1.1 with the EPOC Java Virtual Machine. The EPOC JVM supplies the Java Abstract Windowing Toolkit, or AWT interfaces to the EIKON GUI, allowing Java applications to be written that maintain the normal Psion/EIKON look-and-feel. We could not test Java on the Psion, as the JVM was not available at the time of this writing.

The system idle thread in EPOC32 performs power conservation. In EPOC, like other real time operating systems, at least one thread, usually referred to as the *idle* thread, is always running. When the idle thread senses that no other threads are running, it quiesces the microprocessor and associated electronics until an interrupt occurs. If an interrupt occurs, the processor is immediately brought to full speed and the support electronics powered up. Psion claims that the Psion 5-battery life is 35 hours, although we found it to be much less than 20 hours.

Windows CE

Windows CE is a new operating system, written from scratch by Microsoft providing a platform on which to leverage the Windows operating system in the handheld and embedded system space. Windows CE is the third attempt by Microsoft to provide such an operating system, and it is by far the most advanced and aggressive effort on Microsoft's part to further the "Windows everywhere" mantra. Microsoft itself is not a hardware vendor, and has opted to leave the hardware part of the Windows CE system to the vendors that have experience in that area. A vendor licenses the Windows CE operating system and the Windows CE Services from Microsoft and in turn, Microsoft supplies the vendors with a version of Windows CE that runs on the vendor's hardware. At the time of this writing, that hardware includes the NEC 4102, NEC V4111, Hitachi SH-3, Hitachi SuperH RISC, Philips PR31500 and MIPS 4000 processors. Most of the Windows CE-based systems run at 80 to 100 megahertz, and all of them have at least four megabytes of RAM and four megabytes of ROM. The higher-end units such as the NEC MobilePro 750 have 16 megabytes of RAM, plenty of room for even the most demanding applications.

A key turning point in the development of Windows CE was the decision to add support for the Win32 APIs. This decision alone will virtually guarantee the success of the Windows CE operating system. Developers already familiar with Windows programming using the Win32 APIs will find it easy to write programs for a Windows CE device. Programmers will likely find that a large portion of their code from desktop applications can be easily modified to run on a Windows CE handheld device.

Windows CE supports most of the popular Windows programming interfaces such as Common Object Model or COM, ActiveX, Microsoft Foundation Classes or MFC, and the Active Template Library or ATL. The COM interface is Microsoft's strategy for creating robust, reusable components. These binary objects can be queried at run time by exposing their methods through interfaces.

The COM model is language-independent, which allows these binary components to be updated, changed, or replaced without regard to what language the calling program is written in.

ActiveX is a specification for controls that is actually a specific type of COM object. The ActiveX control exposes its properties, methods and events through a COM interface.

MFC is a large user interface class library for building graphical applications in C++. It provides classes for Window management in an object-oriented form, and acts as an object-oriented shell over the Win32 API layer.

ATL is a set of C++ templates for ActiveX controls and COM components.

Windows CE offers a fine granularity of control over the operation of Windows CE applications. Using thread priorities, a program can perform critical functions such as data acquisition in high-priority threads, and less important functions in lower-priority threads. Windows CE priorities are based on the thread model, not the process model. At this time, Windows CE supports 8 priority levels for each thread, but Microsoft has announced that they are going to increase the number of choices in the near future. These priority features, when combined with the deterministic latencies of Windows CE, make it a good choice as the operating environment for real-time data acquisition systems.

The Windows CE operating system uses demand paging to bring applications in from ROM or another type of persistent store. In most cases, the persistent store is ROM, so when pages are brought in to memory off of ROM, the old pages are discarded. This means that Windows CE can be used on a system with as little as 500 kilobytes of RAM and 500 kilobytes of ROM, permitting larger applications to be paged in and out of memory as necessary [6]. However, some processors, such as the MIPS, do not use the page table architecture that is an integral part of the x86 architecture. Windows CE implements the Translation Lookaside Buffer, or TLB, in software to provide compatibility for those systems that do not have a hardware TLB. Page tables are built as needed and recycled. When the system takes a page fault, the address is looked up in the TLB, and if it doesn't exist, the mapping is performed, and the TLB is updated. To keep the TLB size under control, the table is not allowed to grow larger than 40 pages or so, and the existing entries are reused. To conserve the number of page faults, many of the APIs are implemented with pointers. Calling a user-mode service from a user-mode program does not require copying of any data, as the service can use the same pointer passed in the call to access memory in the caller's address space.

Windows CE applications communicate using the client-server method. Messages are passed back and forth using an RPC method. The RPC method allows applications to call remote APIs transparently as if they were located in the same physical computer even though they might be located in another computer on the network.

Windows CE defines a device driver architecture based on a user-mode model. Windows CE device drivers can be loaded at boot time or after the system is running. The drivers are loaded as user-mode tasks similar to the EPOC32 operating system. This saves expensive pointer mapping when an application calls the driver, as no mode transitions are required. The driver can copy data directly to and from the user-mode application's buffer without any address translation. Another benefit of loading device drivers in user mode is that the operation of the device driver is protected under the standard memory protection features of the operating system. User-mode device drivers are also much easier to debug.

A great deal of work was done to Windows CE to minimize memory usage. With a one-kilobyte or four-kilobyte page size, allocating small amounts of RAM can lead to an inefficient allocation of memory. The operating system keeps a user-mode and kernel-mode stack for each thread in a process. For an application with several threads, this could result in a lot of wasted memory space. To minimize this, the stacks used by the application and the device driver are shared. Windows CE uses a unique type of service called a Protected Server Library, or PSL. A PSL is a cross between a DLL and an application, and it registers its callable functions with the Windows CE kernel. When an application calls the PSL function, the PSL function traps the Windows CE kernel, which saves the stack image in a linked list and adjusts the parameters to fit the particular API call. It also changes the called function's return address to point to the kernel. When the function exits, the kernel traps once again, the stack is restored and the linked list is updated.

Applications are loaded into Windows CE in *slots.* Slots are simple 32 megabyte slices of virtual memory. Each slot contains 512 64-kilobyte blocks; each block contains pages that are one kilobyte or four kilobytes in size. The currently running process always resides in virtual slot 0, while the operating system resides in slot 1. As services or programs are installed, they occupy slots starting at the lowest number. As services are run, the currently running service gets assigned to slot 0. Thus if an application in slot 0 calls the graphics engine in slot 5, the graphics engine becomes slot 0 until it finishes, at which time it becomes slot 5 again. No program data is moved, and the operating system changes slots simply by swapping pointers.

Windows CE contains a registry much the same as its bigger brothers, Windows 98 and Windows NT. The Windows CE registry is capable of holding hundreds of thousands of registry entries or *keys*.

The file system supports several types of devices, including ROM, SRAM, DVD and ATA hard disks. The file system lower layers are documented to allow ISVs to write custom file systems. A developer need only supply their own read, write, and seek functions to provide support for their own file system. Windows CE also allows installable file systems to be used.

Development Tools Evaluation

PalmPilot Development Tools

PalmPilot developers have several choices for development tools. The most popular development environment for the PalmPilot is the CodeWarrior development toolkit from Metrowerks [www.metrowerks.com]. The CodeWarrior tools are available for Windows or Macintosh, and are based on Metrowerks' highly successful line of cross compilers and cross-development tools for other platforms. In fact, the Windows version of the PalmPilot development tools is actually written on a Mac, then recompiled and linked by a high-end translator program that converts the object and executable code to Windows format.

Also shipped with CodeWarrior for Palm OS v4 is the Palm OS Conduit SDK (for Windows). The Palm OS Conduit SDK is used to develop "conduits". Conduits are software that works with Palm Computing's HotSync® architecture on Microsoft Windows computers to synchronize PC application data with data on the PalmPilot organizer [34].

The PalmOS services are documented in the Metrowerks CodeWarrior for PalmOS toolkit. The toolkit does not come with any printed documentation, but does include several large documentation files on the CDROM in postscript format. The volume of documentation is quite large but barely adequate, and includes many API definitions that were labeled “Not Documented – For System Use Only”. These were generally the APIs that we though were the most necessary to document.

The good news for part-time developers and those on a tight budget is that a complete set of Gnu tools now exists for the PalmPilot. Gnu tools are free and can be downloaded from the Web at www.roadcoders.com/pilot/index.html. Other tools are listed below.

CASL Tools (Compact Application Solution Language) from Feras Information Technologies [www.caslsoft.com] is a Windows-based integrated development environment for creating high-level interpreted applications for PalmPilot. CASL Tools include a development environment, a HotSync conduit, and sample programs with source code examples. CASL Tools are for creating integrated desktop and handheld applications with conduits.

The Massena.com site contains tools that let you create PalmPilot applications using many development environments ranging from a BASIC interpreter to an assembler. Many of these tools are shareware or freeware and work in Windows, Macintosh and Unix development environments.

OnTap from Aegean Associates, Inc. is an elegant and simple solution to format and distribute HTML corporate data on the PalmPilot. OnTap puts instant access to portable corporate data in the palm of your hand. OnTap is used to create handheld applications.

Pendragon Forms from Pendragon Software, Inc [www.pendragon-software.com] is a fast and simple way to create PalmPilot data collection applications. PilotForms is for creating handheld applications and conduits. Pendragon Software is a leading developer of software solutions for the PalmPilot.

RTAccess from the Windward Group [www.wwg.com] is for developers that want to easily synchronize their custom applications with the standard databases on the PalmPilot. RTAccess is for creating conduits for the built-in PalmPilot applications.

Satellite Forms from SoftMagic Inc. [www.softmagic.com] is a visual, rapid-application-development environment for creating custom applications and conduits for Palm Computing handhelds. Applications can be designed with "drag and drop" ease, and developers can quickly add functionality with filters and actions. Programmers incorporate business logic through event-driven scripts in a language similar to Visual Basic®, and extend Satellite Forms with custom SFX controls, plug-ins, and the Satellite Forms API. Satellite Forms can be used with Satellite Forms' ActiveX control to create integrated applications.

The Palm OS Emulator emulates the hardware of the various models of Palm Computing Platform devices, letting you run a virtual Palm device on your desktop computer (Windows, MacOS and other platforms). It is extremely valuable for writing testing and debugging applications, particularly when used in combination with the Debug Palm OS ROMs and the built-in Gremlins facility.

The Palm OS Emulator was originally known as Copilot, and was not written by Palm Computing, but by several different developers, principally Greg Hewgill. Palm Computing has enhanced it with new features, debug support, and support for the PalmOS 3.0 ROM. The source code is still available to the public.

Psion Development Tools

The tools required to develop software for the Psion are not easy to find. Developers have two basic toolkits with which to build Psion applications.

One choice is to use the Psion C++ SDK, which works in conjunction with the Microsoft Visual C++ toolkit. The Psion C++ SDK adds extensions and tools for the Visual C++ compiler and IDE, and provides an emulator for debugging applications under Windows 95, Windows 98 and Windows NT. The emulator platform contains a subsystem called WINS that acts as a Psion emulator, allowing the Psion application to be debugged on the Windows desktop. Once debugged, the application is recompiled for the Psion and downloaded via the PsiWin desktop software.

Psion development can also be done using a high-level language called OPL. With this BASIC-like language, programs can be written very quickly, providing a good development solution for programmers that can't afford the Visual C++ compiler or who need to prototype screens very quickly. The OPL language SDK contains a set of Gnu tools for the Psion. Both SDKs can be obtained from the EPOC World Administrator at Symbian Ltd. in London. Symbian retains control of the tools, and licenses each copy.

OPP is an OPL preprocessor for the Psion family of computers which performs a job similar to that of a standard C preprocessor. OPP is designed for Psion application developers who program using the Psion OPL language. The Psion provided utility called S3ATRAN provides similar facilities when translating OPL code on a PC. OPP also provides a number of extensions to the OPL language, which are not available when using standard OPL or the S3ATRAN utility. Examples of language extensions include support for multi-dimensional arrays (standard OPL is limited to one-dimensional arrays) and C style structures. When using OPP, additional preprocessor commands are added in-line to the source file. These commands are read and executed by the preprocessor that strips the lines out prior to the actual compilation or translation of the source file. OPP looks for lines within the OPL source file which begin with a “#” character. These lines contain commands that are read and acted upon by the preprocessor. Any “#” lines are removed prior to dispatching the lines to the OPL translator. The standard OPL code is also scanned by the preprocessor which may substitute symbolic names for text defined using the “#”. The output from OPP is pure OPL source code. This resultant pre-processed OPL code is passed on to the OPL translator that does the actual job of creating an OPO or OPA file.

Bundled with OPP is a utility called OPPDBG that can be used when developing applications on the Psion in the OPL language. Debugging OPL programs requires that the programmer use PRINT statements to locate errors. This can be a time consuming and error prone process since it often requires the code to be retranslated many times in order to pinpoint where a problem is occurring.

With OPPDBG, debugging OPL applications is much more straightforward. You can see exactly which lines of code are being executed, and the values of all local variables without the need for print statements or retranslating the code multiple times. You simply write you application using standard OPL, translate the code once under debug and then use the debugger to track down what exactly is happening within the running application.

An interesting product that has just been released for the Psion is OPL++, an object-oriented version of the popular OPP language. The OPL++ emulator versions run on a PC under the release and debug versions of the OPL emulator. Two versions of OPL++ are available, one for the C++ SDK and one for OPL SDK.

There are two main components to OPL++, the editor and the preprocessor. The OPL++ editor uses OPP as the back-end preprocessor and translator. Combining OPL++ with the standard OPP SDK provides a cross-platform, object-oriented 16-bit and 32-bit development environment for EPOC16 and EPOC32.

Palm-size PC Development Tools

Programs written for Windows CE are developed using the Microsoft Visual C++ compiler and Integrated Development Environment, or IDE. Microsoft provides a Windows CE SDK that complements the standard Visual C++ compiler and tools. Like it big brothers Windows NT and Windows 98, Windows CE provides the standard Win32 APIs, although the actual number of APIs it provides is much less than the number of APIs available on Windows 98 or Windows NT. The SDK includes the required headers, samples, make files and libraries for Windows CE development.

Programs can be written in C, C++, J++, or Visual Basic. The Windows CE Toolkit for Visual C++ includes cross-compilers for all the supported Windows CE processors, the Microsoft Foundation Classes, Class Wizards and Active

Template Library. ATL allows developers to easily create ActiveX controls for Windows CE. The toolkit also supplies remote versions of the familiar Windows development tools such as Spy, Registry Editor and Memory Viewer. Microsoft supplies several samples of Windows CE applications that can be used as an aid for developers new to Windows CE. The Windows CE development environment provides seamless integration with the Visual C++ IDE.

Currently, Windows CE exports 600 of the Win32 APIs, although Microsoft is in the process of adding more of them. The Windows CE development tools provide a Windows CE emulator that runs on Windows NT only. The emulator allows program development to be done entirely on a Windows NT system, and provides a graphical representation of a Windows CE system on the NT desktop. Programmers can use the emulator to debug programs until they are ready to be downloaded to the actual hardware. This saves time because the download operation does not have to be performed for each build of the software. When the code is downloaded, a remote debugger allows the application to be debugged from the host NT system. The remote debugger does not work on Windows 95 or Windows 98.

Porting applications from Windows 95, Windows98, and Windows NT is a fairly straightforward task. Most of the standard Windows controls are supported, with some limitations. For instance, menu and toolbars are combined into a single command bar. Applications that use semaphores need to be modified to use other types of synchronization methods such as critical sections. Some user interface interaction may be different due to the fact that the handheld uses a stylus instead of a mouse. Also, the user interface may include speech or handwriting recognition not normally present in the desktop version of the application. The display properties also vary widely between devices. Some may have a color display while others have a black and white display. The Palm-size devices in general have a smaller screen size, while some of the Windows CE palmtops have much larger screens. At the high end of the spectrum, the NEC MobilePro 750 sports an eight-inch display screen with 256 colors.

Windows CE supports a limited set of Winsock, TAPI, RAS, and serial APIs, so applications that use these APIs may have to be modified slightly. Windows CE supports a limited number of the WSA APIs, and supports connections to networks running IPX, NetBEUI, or TCP/IP, and also supports the SLIP protocol.

Handheld Applications

PalmPilot Standard Applications

The PalmPilot comes with an adequate set of applications to perform the tasks that it was designed for. These applications include a Date Book, an Address Book, a To Do List, a Memo Pad, and an Expense Minder. On the current PalmPilot III, users can store thousands of address, ToDo, memo pad and date book entries. PalmPilot databases are compressed to allow maximum storage by eliminating blank fields and packing records with null-separated fields. On the PalmPilot III, 3COM claims that users can store some 6000 addresses, 5 years of appointments (approximately 3000), 1500 to do items, 1500 memos, and 200 e-mail messages. With the expense application, users can enter expenses and then automatically transfer them into an Excel (5.0 or higher) expense report form or the user's expense template during synchronization. The PalmPilot's Email application allows users to view, compose, delete or save messages. The calendar application lets users prioritize their daily tasks, sort items by priority or due date, check off what they've completed and carry forward the tasks that have not yet been completed.

The PalmPilot includes a handwriting application called Graffiti, which is integrated into the PalmPilot operating system. Graffiti is a unistroke recognition system, requiring users to write in a "limited manner". The system converts small gestures from the stylus into characters that can be used as input for most of the PalmPilot's applications. The PalmPilot display contains a special area below the display that is reserved for Graffiti input. As the characters are input in the Graffiti area, they are immediately converted to their character equivalent and displayed, providing immediate feedback for the user. To aid users in learning Graffiti, the PalmPilot includes an application called Giraffe. To help users learn Graffiti, 3COM designed Giraffe as a game. Users chase falling letters and attempt to erase them by entering the Graffiti strokes for the falling letters.

Users who are uncomfortable with Graffiti or who desire to use more traditional forms of input can pop up a small soft keyboard on the bottom of the display screen. To enter data, the user taps the desired key on the soft QWERTY keyboard.

PalmPilot Shareware and Commercial Application Highlights

There are hundreds and hundreds of applications written for the PalmPilot, and many of them are available as freeware or shareware on the Web. While there are some commercial grade applications, these applications usually are part of a corresponding application that runs on the desktop. For instance, Symantec's Act! contact management software can be accessed from the PalmPilot, but the full featured program runs on the user's desktop machine, not on the handheld. There are literally hundreds of calculators, calorie counters, language dictionaries, cookbooks, bibles, golf scorekeepers and money converters available for almost nothing. There are, however, very few serious applications for the PalmPilot. This is due in large part to how the PalmPilot was designed. The PalmPilot provides a “window” to the customer’s data, but does not provide the resources necessary for any major programming tasks. The PalmPilot works very well as a viewer of data, which is what it was designed to do. It cannot, however, perform any major data processing tasks.

The number of freeware, shareware, and commercial applications available for the PalmPilot could fill several pages. To pare the list down, we have instead provided a list of the most popular PalmPilot programs. Users can find more information regarding these applications and others by pointing their browser to http://www.palmpilot.3com.com.

AportisDoc Mobile Edition from Aportis [http://www.aportis.com] allows books, reference materials, stories, manuals, and any other large documents to be efficiently stored, read, and exchanged. It contains full search, book-marking capabilities, multiple fonts, and a teleprompting feature. Thousands of free books and other documents are currently available in AportisDoc format.

AvantGo from AvantGo, Inc. [http://www.avantgo.com] allows users to synchronize their PalmPilot with information from the Web or a corporate Intranet. Users can HotSync product inventory or schedule information from their company databases directly onto their PalmPilot. They can also HotSync sports scores, headline news or stock quotes from AvantGo Content Partners such as The New York Times, InfoWorld, Wired, Mercury Center, Excite and CNET.

TripMate from DeLorme [http://www.delorme.com] is an easy-to-use navigational tool. With TripMate, users can download a door-to-door route from Street Atlas USA® 5.0 and use the DeLorme Tripmate™ GPS Receiver and the user's PalmPilot organizer to constantly update their position, speed, elevation, latitude and longitude, and to see the distance and time to their next turn.

Maximizer Link from Maximizer [http://www.maximizer.com] allows users to transfer data between their Maximizer PC database and the PalmPilot. The user can then execute the PalmPilot's built-in functions including ToDo, DateBook, AddressBook, and MemoPad to access the Maximizer data.

IntelliSync from Puma [http://www.puma.com] enables users to synchronize their PalmPilot organizer directly with their favorite PC-based personal information management, contact management and group scheduling applications all in one easy step.

ECCO Pro from Netmanage [http://www.netmanage.com] is an information manager that lets workgroups and individuals keep up to date on their calendars, contacts and projects. ECCO's Group Scheduling allows users to easily schedule meetings with colleagues, conference rooms and AV equipment. The PalmPilot conduit within ECCO lets users synchronize easily between the PalmPilot and the desktop version of ECCO.

ACT! from Symantec [http://www.symantec.com] provides synchronization tools for sharing data and communicating with the desktop version of the popular ACT! contact management software.

OneTouch from JP Systems [http://www.jpsystems.com] offers two-way alphanumeric messaging and email by utilizing the PalmPilot Minstrel modem. OneTouch integrates with SkyTel's SkyWriter or MCI's interactive two-way paging service.

EasySync from Lotus [http://www.lotus.com] provides users with the ability to synchronize the data from data from their popular Notes mail and Organizer 97 GS desktop software. Lotus Organizer 97 GS integrates an on-screen Calendar, To Do list, Planner, Address book, Call manager, Notepad, Anniversary reminder, and integration with the Lotus Notes and Domino.

PocketChess from Scott Ludwig [http://www.eskimo.com/~scottlu/pilot/] is a high quality, well-featured chess playing program for the PalmPilot.

Froggy from PilotFan [http://www.pilotfan.com] is a popular game based on the older PC-based arcade game by the same name.

EPOC32 Standard Applications

EPOC32 comes with a rich set of applications including an Email application that provides integrated Internet SMTP/POP3 Email and fax messaging. Also included is a Web browser that support HTML 3.2, excluding frames but including support for tables, forms, GIFs, JPEGs and animated GIFs. The built-in terminal engine supports login scripting, XMODEM and YMODEM file transfer, and emulation of TTY and VT100 terminals.

EPOC32's standard applications suite includes:

- System – the system shell
- Word, a full featured word processor
- Sheet, a two-dimensional spreadsheet
- Agenda – a powerful scheduling application with ToDos and activity management
- Data – a general purpose database with object imbedding, sorting and searching
- Email – a message application that supports SMTP/POP3-based internal mail
- Web – a full featured Web browser
- Comms – a terminal emulator that supports VT100 and TTY emulation
- Sketch – a bitmap editor and drawing tool
- Time – a world clock with multiple views
- Spell – a 100,000 word spell checker
- Calc – a full-featured scientific mode calculator
- Record – a voice recorder that allows the user to imbed recordings as documents

EPOC32 and the Psion applications are written and supported by Symbian, a software company formed as a partnership between Psion PLC, Nokia, and Ericsson. It is not surprising therefore that Nokia and Ericsson have chosen the EPOC operating system to be used in their cellular telephones. Nokia's 9000

Communicator model, which is based on the ARM7 core, uses EPOC32 to provide PDA services as well as telephone functions.

EPOC32 Shareware and Commercial Application Highlights

EPOC32 Shareware and Freeware can be downloaded from two sites, one in the United Kingdom [http://src.doc.ic.ac.uk/packages/psion/icdoc/] and one in the United States [ftp://ftp.nwt.com/pub/].

The EPOC shareware and freeware sites can only be described as a hodge-podge of one-liners, with most programs written for the older Psion 3 Series systems. With no information or program names available, users have to play a guessing game to determine what software is available and what features it has. In the Psion 5 section, we found only 20 or so entries listed only by the name of a zip file.

Q-plus Bridge from Brea Software [http://www.q-plus.com] is a bridge game.

VoyageMaster from Dolphin Software [http://ourworld.compuserve.com/homepages/Dolphin_software/voyage5.htm] contains topographical and statistical data for over 600 European Ports, from the Azores to Norway, covering England, Scotland, Ireland, Wales, Channel Islands, Norway, Denmark, Germany, Netherlands, Belgium, France, Spain, Portugal, Azores, Madeira and Canary Islands and over 200 Ports in the U.S.A.

MemoVoc from Estuary Technologies [http://www.cix.co.uk/~djoyce/] is a voice recording application for the Psion 3C. It features a one-button record and playback, and says things like “Begin recording, Captain” and “Thank You, Captain”.

Geographical Database from Palmsoft [http://www.palmsoft.co.uk/] is a road atlas of the United Kingdom for the Psion 3C.

Report Writing from HS Software [http://www.argonet.co.uk/h.s.soft/] allows users to write reports on the Psion 3C.

Sierra Flight Planner from NavTech Software [http://www.avnet.co.uk/navtech/] provides users the ability to schedule and track flights along with dozens of functions for pilots.

Odessey from NavTech [http://www.avnet.co.uk/navtech/] is a pilot log book application for the Psion 5.

Filmcalculator from Zebra Film [http://www.xs4all.nl/~zebra/html/the_filmcalculator.html] is a calculator for people in the film industry.

Windows CE Standard Applications

Windows CE systems include a rich set of applications, including pocket versions of Microsoft Word, Excel, and PowerPoint. The natural affinity to Windows makes these applications undeniably the most important reason to use a Windows CE system. Users can transfer documents, data files, and presentations from the Windows CE machine to the desktop or from the desktop to the Windows CE system. In some instances, certain types of formatting or font information can be lost or reformatted for the particular device. For instance, fonts that exist on the desktop may not exist on the handheld device. For this reason, documents, data, and presentations tend to remain in the form acceptable by the handheld device.

Most Windows CE systems come with the Windows CE version of Internet Explorer called the Pocket Explorer.

Windows CE provides the JOT system for handwriting recognition. Unlike the PalmPilot’s unistroke Graffiti recognition system, JOT allows users to enter the desired characters in a natural form, as if they were actually writing the characters. JOT does not require users to memorize special characters. In order to keep the memory requirements low, JOT uses Windows CE’s eXecute In Place, or XIP technology to run directly out of ROM.

The Windows CE 2.0 and later versions support voice recording and voice command and control with limited vocabularies. The user selects a particular operation and then records the command by speaking into the built-in microphone. Windows CE uses the recorded phonemes to match the spoken words with their corresponding commands. Windows CE also allows users to store contacts in the address book and look them up later by speaking their name. The user enters the name into the address book, then speaks the name twice. Windows CE then links the voice with the name and uses the data to look up the name at a later time. The user can open up the address book and say “lookup Johnson”, and the address book program will locate the matching address book entry for that voice command. This method of voice recognition lacks the sophistication of higher-end products such as IBM’s ViaVoice, or L&H’s Voice Professional, which determine words based on not only their sound but on the surrounding context and sentence structure.

Windows CE systems provide voice recording from a built in microphone. A voice recorder application allows for the recording of approximately 60 minutes of audio at a low sampling rate.

Windows CE systems include a copy of Microsoft’s Windows CE Services, a set of services that provide connectivity and synchronization with a desktop system.

Most Windows CE systems ship with extra free software such as Web browsers, spell checkers, POP3/IMAP4 Email applications, FAX applications, financial calculators, database utilities, PDF file viewers and screen capture utilities.

Windows CE Shareware and Commercial Application Highlights

Calligrapher from Paragraph PI [http://www.paragraph.com] provides a handwriting recognition program for Windows CE that recognizes cursive writing or mixed handwriting. It integrates with all Windows CE applications and includes a spell checker and deferred recognition[16].

eWallet from Ilium Software [http://www.iliumsoft.com] safely stores information on credit cards, account numbers, passwords, PINs, insurance information and more. The data is encrypted for security.

bFAX from bSQUARE Development [http://www.bsquare.com] is a complete fax application, allowing users to compose and send faxes providing the Windows CE system has a fax modem installed.

XTCE from Pyram-ID [http://www.pyram-id.demon.co.uk/XTCE.html] DOS Emulator, an x86 emulator for Windows CE systems.

Visual CE from Syware [http://www.syware.com] allows users to create custom forms and databases for Windows CE.

Pocket On-Schedule from Odessey Computing [http://www.odesseyinc.com] is a powerful and complete contact management system with multiple calendars, to do lists and multiple views.

Microsoft PowerToys [http://www.microsoft.com/windowsce/hpc/software/power.htm] is a set of utilities for Windows CE that includes squirting data from one Windows CE machine to another via the infrared port.

Data Anywhere from Smart Ideas Software [http://www/smartidz.com] provides client and server software to utilize existing desktop or network databases on the handheld device.

ICQ from ICQ, Inc. [http://www.icq.com] is an Internet chat tool for Windows CE.

TN3270 from Murkworks [http://www.murkworks.com/products/tn3270ce] provides 3270 connections and emulation on a Windows CE device.

[16] Deferred recognition means that the data can be stored and converted later at synchronization time.

Pocket Maximizer from Multiactive Software [http://www.maximizer.com] is a powerful contact management system that shows connections between companies, People, tasks, appointments and notes associated with them. It links to Pocket Word and has one-button email. It also shares databases with the Windows 98/NT version of its big brother Maximizer.

Newsworthy

PalmPilot News

On September 28, 1998, 3COM [http://www.3com.com] announced a partnership with Sybase, Inc. [http://www.sybase.com] to provide access to Sybase's SQL database software and network-based synchronization of those databases with the PalmPilot. See the entire story at http://www.palmpilot.3com.com/pr/sybase.html.

Using this technology, users of the PalmPilot can access Sybase data located on a desktop machine or directly from a server. Mobile PalmPilot users can synchronize the Sybase database with their PalmPilot using a dial-up connection directly to the server. PalmPilot users can have access to corporate data and can perform updates on the PalmPilot, which are later used to update the master database records at synchronization time.

"The relationship between Sybase and 3Com and the introduction of Sybase's UltraLite database for the PalmPilot organizer will enable us to break new ground in delivering point-of-care mobile healthcare solutions," said Mathieu Wiepert, senior technology consultant at BORN Information Services, a Sybase System Integrator Partner. "Doctors and nurses will now be able to access and update critical patient information at a patient's bedside, significantly reducing administrative costs and increasing the accuracy of patient profile information."

"Both Sybase and 3Com have an early technology lead in the handheld market," said Dale Okuno (CLU), president of E-Z Data Inc., a leading sales force automation vendor. "Many of our 70,000 users in the insurance industry have already expressed interest in Palm based devices, and we look forward to working with Sybase and 3Com to spur our growth in the handheld marketplace."

On September 14, 1998, Palm Computing announced a partnership with SAP to provide mobile access to SAP’s R/3 data. The remote access to R/3 is made possible by Ábaco [http://www.abaco.com], which spearheaded the solution development. Ábaco's software, The Bridge for R/3, connects Palm Computing platform devices to information resident in SAP R/3 applications. The mobile access solution is ideal for applications such as sales force automation, help desk automation and warehousing. See the full story at http://www.palmpilot.3com.com/pr/sapphire.html.

"The Palm Computing platform enables SAP customers to realize significant productivity gains, cost savings, efficiency and time savings by integrating handheld devices with R/3, the backbone of the SAP ERP systems," said Mark Bercow, vice president of strategic alliances and platform development for Palm Computing. "These devices offer SAP customers a revolutionary and cost-effective option for accessing R/3 that requires very little training to get end users up and running."

Fernando Alvarez, president of Ábaco International Group said, "This mobile initiative has created an extremely attractive option for technology decision makers to increase company-wide efficiency. Since we announced our relationship with 3Com and SAP in June, we've seen tremendous customer interest in the concept of accessing ERP data from a handheld device."

On July 30, 1998, the Experimental Aircraft Association announced that it would supply the PalmPilot to its judging staff that allows the judges to rate aircraft while in the field. Read the entire story at http://www.palmpilot.3com.com/pr/eaa.html.

Andy Stadler, an EAA member and volunteer software developer, designed an application for the PalmPilot organizer which allows judges to score “homebuilt” aircraft in a variety of categories, from the strictly technical (including fuselage, power plant and landing gear) to the creative (including fit and finish, innovation and overall execution). The software application will automatically tally each score, providing the judge’s overall score for that particular aircraft. Once the judge has made the rounds at the air show, he returns to the central scoring station, pops the PalmPilot organizer into its HotSync® cradle and, with the touch of a button, downloads the scoring data into a PC. We've created a complete end-to-end solution that saves time and improves accuracy," said Stadler. "

EPOC32/Psion News

On July 24, 1998, Symbian announced a partnership with Nokia and Ericsson to develop the EPOC32 operating system to develop seamless, wireless connectivity using the new Bluetooth 2.4GHz wireless technology. The entire story can be view at http://www.symbian.com/news/press/1998/pr980724.html.

"Bluetooth is no less than a revolution in device connectivity and inter-operability. Capitalizing on our extensive experience in providing robust operating systems for Wireless Information Devices, Symbian has long been committed to short range wireless technology." Says Juha Christensen, Executive Vice President, Marketing and Sales, Symbian. "Seamless connectivity between mobile devices and all other computing devices has always been at the core of our strategy, and the wireless connectivity that Bluetooth provides will dramatically change the way Wireless Information Devices and other computing devices communicate with each other. EPOC will be the first mobile operating system to support Bluetooth".

"We are delighted to welcome Symbian to the Bluetooth initiative, which has garnered enormous interest since its launch," said Simon Ellis, Communications Marketing Manager, Intel. "Together our technology will replace the need for business travelers to purchase or carry multiple, often proprietary, cables by allowing devices to communicate with each other through a single port."

On March 25, 1998, Symbian announced a partnership with Sweden's Sendit AB to provide wireless messaging services for EPOC32-based mobile computers, telephones and communicators. Read the entire story at http://www.symbian.com/news/press/1998/pr980325.html.

Juha Christensen, Vice President, Licensing, Psion Software comments "Wireless email and content provision are critical enablers for the mobile user. Our partnerships with content providers and technology companies mean more than just technology provision - users of EPOC Smartphones and Communicators will have the most flexible and reliable access to a wealth of critical information. The interoperability involving Sendit's ICSA-server and EPOC-based devices further increases EPOC's offerings to mobile communications OEMs."

"The fit for purpose nature of EPOC for mobile telecommunication devices, together with Psion Software's success in signing with major telecommunications companies, makes it the ideal choice for the platform." added Hjalmar Winbladh, Managing Director of Sendit. "Together we're shaping the future of wireless Internet connectivity, offering end users a higher quality of service and more functionality in a cellular environment than most Internet Service Providers can offer in a fixed one."

On March 5, 1998, Philips Consumer Communications [www.philips.com] of the United Kingdom announce the Smartphone, a clip-on telephone based on the Philips Ilium GSM mobile phone. The Smartphone provides email, fax, data and Internet browsing capabilities. Read the full story at http://www.symbian.com/news/press/1998/pr980305.html.

"Synergy is a radically new concept, destined to open up new opportunities in the Smartphone market." commented Juha Christensen, Vice President, Licensing, Psion Software. "Using EPOC as their operating system allows OEMs to produce innovative voice and data communication devices which offer great ease-of-use and thus appeal to mass market users. The introduction of these new smart devices into the market increases choice and mobile functionality for the corporate and consumer user. With conservative estimates of the size of the Smartphone and Communicator market at 29 million units in 2001, Psion Software is uniquely placed with its licensees to benefit from the rapid growth in this industry."

Jorgen Bredesen, General Manager Europe, Middle East and Africa, Philips Consumer Communications comments "Ever since wireless technology was first introduced, people have been seeking ways to provide full communications functionality - including comprehensive voice, data and messaging - over the airwaves. With Ilium Synergy we believe we've achieved it."

On February 18, 1998, Psion announced a partnership with Lotus [http://www.lotus.com] to provide mobile synchronization for Notes calendar, To Do, memo pad and address databases as well as Notes email. See the full story at http://www.symbian.com/news/press/1998/pr980219.html.

Lotus Development Corporation's UK Internet Product Manager, Victor Aberdeen, comments, "This connectivity tool helps address the growing demand from users who want to access Lotus Notes whilst on the move. Access to Lotus' calendar and scheduling allows us to move further towards our goal where Domino will be the most well connected and easily accessed messaging and groupware platform."

"We've responded to the demands of our OEMs' corporate customers by providing a total mobile synchronization solution for Notes." adds Simon East, Vice President of Products, Psion Software. "Our mission is to continue defining and leading the mobile ROM-based computing industry. The Lotus Notes mobile solution will expand this market by providing the first truly useful mobile application for corporate users."

Windows CE News

On June 30, 1998, Hitachi [http://www.hitachi.com] announced the establishment of a dedicated development organization focused on building Windows CE-based solutions, reportedly with broad support and cooperation from Microsoft. “Microsoft will provide this new organization with broader access to Windows CE technology, enabling Hitachi to rapidly build a wide range of customer solutions powered by Windows CE” said a Microsoft spokesperson. The entire story can be seen at http://www.microsoft.com/presspass/press/1998/Jun98/HitchPR.htm.

"We are very excited about this new level of cooperation between Microsoft and Hitachi," said Bill Gates, chairman and CEO of Microsoft. "The real beneficiaries of this combined effort will be the businesspeople and consumers who, in coming years, will be able to incorporate these new products into their daily lives and benefit from the increased productivity and entertainment provided by these connected, intelligent digital devices."

"Hitachi selected Windows CE because of its synergy with Windows NT® and its potential for the future," said Dr. Tsutomu Kanai, president and representative director of Hitachi Ltd. "Microsoft is a worldwide leader in software for office and home computers with its Windows family of operating systems. Hitachi is one of the world's largest companies, covering a wide range of markets and product applications, including enterprise systems, consumer products, real-time industrial control systems and electronic components. Through the combined strengths of our two companies, we envision an environment of seamless data networks across user environments and product areas."

On May 12, 1998, Microsoft announced a partnership with BellSouth [http://www.bellsouth.com] to provide wireless networks for Windows CE. BellSouth was formerly called RamMobile Data, and was one of the first wireless provides to achieve any critical mass in the mobile wireless market. The entire story can be viewed at http://www.microsoft.com/presspass/press/1998/May98/BellSopr.htm.

"BellSouth Wireless Data's expertise and experience in providing proven wireless data communications solutions makes it the right choice for Microsoft in this exciting endeavor," said Harel Kodesh, general manager of the consumer appliance group at Microsoft. "The strength, intelligence and maturity of its network provide us with a powerful tool as we create wireless applications for Windows CE."

"We're proud that Microsoft has selected BellSouth Wireless Data to bring two-way wireless capabilities to its millions of customers at price levels that will make it attractive for more companies to deploy enterprise-wide solutions," said William F. Lenahan, president and CEO of BellSouth Wireless Data.

On April 7, 1998, Microsoft announced an alliance with Intel to provide voice-activated navigation, telephony, driver information and entertainment for the AutoPC, a Windows CE-based device announced earlier that that in January of 1998. See the full text at http://www.microsoft.com/corpinfo/press/1998/Apr98/intlmspr.htm.

On January 8, 1998, Microsoft announced the AutoPC, a Windows CE-based system for the automobile combining speech, information, connectivity and entertainment. The AutoPC replaces the automobile radio and media player, and substitutes a Windows-CE based system that provides the same functions as the original player and adds additional functions such as speech recording, voice command and control, internet connectivity via satellite and cellular radio and interactive navigation. See the entire story at http://www.microsoft.com/corpinfo/press/1998/Jan98/AutoPCpr.htm.

On March 11, 1998, Microsoft released the Japanese version of Windows CE 2.0, enabling Japanese manufacturers to use Windows CE 2.0 for the Far East market. See the entire story at http://www.microsoft.com/corpinfo/press/1998/Mar98/Jpnv2pr.htm.
Windows CE also supports English, French, German, Italian, Portuguese and Spanish.

On January 8, 1998, Microsoft announced the PalmPC, a handheld version of the handheld PC powered by the Windows CE operating system. See the entire story at http://www.microsoft.com/presspass/press/1998/Jan98/PalmPCpr.htm.

On October 2, 1998, Microsoft announced that they would enable Windows CE to use both sort range and long range wireless communications to connect to Windows NT 5.0. This project, called Chimera, would enable a remote user to stay connected to email and back-end data using a Windows CE device and a wireless service instead of a dial-up connection. For the complete story, see http://www.zdnet.com/pcweek/news/0928/02ece.html.

"We made a decision a year ago to focus on wireless infrastructure," said Phil Holden, group product manager for Windows CE. "Without the rest of the plumbing, wireless communications is pretty much irrelevant."

On July 24, 1998, Anheuser-Busch announced that it would be deploying Wright Strategies' [www.wrightstrat.com] ShelfAudit software that cost-effective automates merchandising and sales activities. For the full story, see http://www.wrightstrat.com/news/main.html.

Anheuser-Busch brewery sales force will be using ShelfAudit nationwide to determine sales opportunities at wholesaler locations. Richard Sleight, Senior Developer, Business Process Improvement with Anheuser-Busch, stated that Anheuser-Busch expects to reduce data turnaround with the field from a month and a half to around a week – a 600% improvement. Additionally, the number of statisticians, who are primarily responsible for entering and analyzing the field-based data, are expected to be reduced by over 50%.

While faster and more accurate data is of great value, Anheuser-Busch views the primary ShelfAudit benefit to be the management of the brewery sales force activities and reporting flexibility. Sleight said, "ShelfAudit enables Anheuser-Busch to customize our distributor surveys across the 11 regions the brewery sales force operates in for specific regional requirements, and still allows for headquarter-level standardization. No other product we evaluated provided us this level of flexibility for field management or reporting".

Connectivity

PalmOS Connectivity

3COM claims to have shipped just over two million PalmPilots since its introduction in 1995 and 1996, yet the PalmPilot still sports only a single serial port for connectivity. The PalmPilot still needs the cradle and serial cable assembly to transfer data to and from the desktop system. Because of some hardware limitations, the speed of the serial port is limited to 57K baud. A few companies have added telephone-like devices to the PalmPilot. A company called Symbol Technologies [http://www.symbol.com] actually uses the PalmPilot as a cellular telephone[17]. If any of these devices are attached, however, the PalmPilot cannot be placed in the cradle and can't connect to the desktop machine.

In June of 1998, PalmPilot introduced the PalmPilot III, a slimmed-down unit with two megabytes of ROM and two megabytes of RAM. The PalmPilot III comes with an infrared port for communications with other infrared-enabled devices. Infrared communications protocol is specified by an independent group of industry participants called the Infrared Data Association, or IrDA. The IrDA specifications define several modes and speeds of infrared connections. The **low speed** mode defines communications from 9.6K baud to 115K baud. The **high-speed** mode covers communications at one megabit per second or four megabits per second. The PalmPilot's infrared port operates at low speed, but since the infrared device utilizes the serial device to operate it, the infrared communications are limited to 57K baud. Another problem is that the infrared communications is not IrDA compliant and thus won't connect or communicate with other IrDA devices. The PalmPilot is capable only of beaming data to another similar PalmPilot, and can't even print to an infrared-enabled printer.

The PalmPilot uses a proprietary 32-byte serial protocol called the Serial Link Protocol to provide reliable synchronization over the serial link.

Psion Connectivity

EPOC32 provides a rich set of networking service and protocols allowing users and developers to connect to many types of networked devices. EPOC32 supports serial communications, serial infrared communications, and dialup TCP/IP connections.

Two servers, the serial communications server and the socket server, provide EPOC32 communications. The serial communications server provides a serial port API to clients, as well as an IrComm interface to the serial infrared transceiver. Using the serial communications server, devices may be shared by more than one client, as the server arbitrates for use of the device. The serial communications server also allows applications that use a serial device to be

[17] Symbol also adds a bar code scanner for use in grocery stores.

written without regard to the specific configuration of the device. An application written for the serial port, for example, could just as well use the serial infrared port with no source code changes.

The socket server provides a BSD-like socket API to clients. The Psion socket API supports TCP/IP, PPP, IrDA, and the Psion Link Protocol, or PLP. EPOC32 uses PLP to provide reliable serial connections for synchronization with the desktop PC. EPOC32's socket API is object-oriented, but also exports a set of C APIs that are provided as a layer on top of the object-oriented socket code. The socket server contains a network interface manager or NIF that is responsible for routing of socket protocols.

EPOC32 provides several integrated applications that are closely coupled to its communications services. EPOC32 contains a mail application that provides integrated Internet SMTP/POP3 email and fax messaging. A dialer is also part of the standard communications suite, and allows dialup settings to be stored for multiple locations and connections. Psion plans to add support for SMS to support messaging over data-enabled phones. A built-in Web application provides support for HTML 3.2, and support for DHTML and XML is planned for the near future. The browser does not support frames, but does support JPEG and GIF graphics as well as animated GIFs. The Web browser, together with the required DLLs, consumes well over two megabytes of memory, making it viable only on the newer Psion 5 devices.

Symbian claims that their strategy for wireless devices includes support for the Wireless Application Protocol, or WAP as well as the new 2.4 gigahertz Bluetooth devices. However, we could find no evidence that these components exist beyond the EPOC32 white papers.

Infrared support uses the serial infrared device on the Psion hardware. The infrared transceiver is driven by the UART located on the ARM7 processor. There is only one serial port available on the processor, so the device cannot drive the serial port and the infrared transceiver simultaneously. The IrDA socket server supports the standard IrDA stack, providing IrLAP, IrLMP, TinyTP and IrMux services. EPOC32 also implements IrComm. EIKON, the Psion user interface service, provides integral support for beaming data from one EPOC32 machine to another. EPOC32 also contains an integrated class 1 and class 2 fax application.

Windows CE Connectivity

Windows CE networking allows a Windows CE system to connect to any network that supports NetWare's IPX protocol, Microsoft's NetBEUI protocol, or the industry standard TCP/IP protocol. As part of its networking support, Windows CE supports:

- ICMP requests (use for PING)
- HTTP and FTP Protocol
- Network file and printer access
- Windows sockets, including IrSock and secure socket layer (SSL)
- IrDA protocols
- NDIS 4.0 for local area and IrDA networking
- PPP and SLIP for networking with a serial cable or modem (PAP and CHAP)
- Remote Access Services (RAS) client support

Network communications are also supported on:

- Serial cables
- IR Transceivers
- Radio transceivers
- Local area networks
- Modems

Windows CE supports the industry-standard IrDA infrared protocol using standard serial APIs. It supports both serial infrared communications at 9600 bits per second to 115,000 bits per second and fast infrared communications at one or four megabits per second. Communications is supported via the IrSock and IrComm interfaces.

Windows CE supports RAS, a software-based, multi-protocol router used to connect a remote device to a host PC. The RAS connection can be a direct serial connection or a modem connection. The Windows CE debug and program development environment uses RAS to send and receive data from the Windows CE device.

Serial support in Windows CE is supplied by the standard Win32 serial APIs and by the standard file system functions. Programs can call the file system to open, close, read from and write to the serial port, as well as to set serial port parameters such as baud rate, parity, stop bits, and handshake options. Windows 98 and Windows NT support a special form of serial communications called Overlapped I/O. With Overlapped I/O, a thread can be reading from the serial port at the same time another thread is waiting for data from the same serial port. This provides the ability to perform full duplex communications using the serial port. Overlapped serial communications utilizes semaphores to notify the waiting thread that data has been received or is done being sent. Since Windows CE does not provide standard Windows semaphores, Overlapped communications does not work in Windows CE. However, Windows CE does permit multiple reads and writes to be pending on the serial device.

Serial communications can take place over a serial cable, modem, or infrared transceivers. Windows CE provides support for managing modem connections through the telephony API, or TAPI. To more effectively support wireless networking, the Windows CE TCP/IP stack is designed so that additional communications parameters such as the acknowledgement timeout can be controlled by software.

Windows CE provides the WinINet API to support the HTTP 1.0 and the FTP Internet browsing protocols. Gopher, cookies, proxies, bypass lists and URL caching are not supported. Three security protocols are supported, secure socket layer (SSL) 2.0, SSL 3.0, and Private Communication Technology, or PCT.

The Wnet API provides access to the Common Internet File System for remote printers and files. The redirector supports the Universal Naming Convention (UNC) names such as \\yktprs04\w360scad but does not support drive letters.

Like the PalmPilot and the Psion, Windows CE uses a proprietary protocol over the serial port to support reliable synchronization under the ActiveSync™ architecture. Windows CE can synchronize data via infrared, modem, LAN, and RAS dial-up connections. Windows CE also supports printing to HP LaserJet [www.hp.com] and Epson-compatible [www.epson.com] printers

Chapter 5. - Summary and Conclusions

Summary

We found all of the handheld systems we evaluated to have a mix of both good and bad features. We divided our evaluation into several sections that appear below. Our conclusions are based not on any one particular feature or device, but on the total value of the device including its operating system, applications, connectivity, development tools, support, and market viability.

Each system was well designed for the purpose it was intended for, and in the case of EPOC32, the system architecture was so well done that we felt it did not deserve being constrained or confined by the Psion handheld device. Looking at the unattractive Psion 3 or Psion 5, one cannot appreciate the beauty of the underlying software architecture and how well it was implemented. However, there is much more to a good handheld device than the operating system, so this feature alone unfortunately will not insure the Psion's success.

Operating System Wrap-up

Each of the operating systems that we evaluated, PalmOS, EPOC32, and Windows CE, were well suited to the task that they were designed for. Palm Computing's PalmOS worked well on the PalmPilot, but did not have the features necessary to support handheld devices in a true mobile computing or enterprise environment. EPOC32, while nicely designed, lacked the polish and features to support enhanced networking, secure sockets, point to point tunneling protocol, PPTP, and NetBEUI. While Windows CE supports all of these features, it is clearly much to big to fit onto the tiny PalmPilot, and has much more functionality than required by a handheld PIM device.

The PalmPilot's operating system, PalmOS, was by far the leanest and meanest of the three we tested. It was also the leanest in terms of features. PalmOS supports only one thread of execution, with one thread dedicated to the HotSync program. This is not a limitation of PalmOS, or more specifically the AMX kernel. In fact, the AMX kernel has no such limitations, and runs on a wide range of processor architectures [http://www.kadak.com]. Palm Computing's decision to cripple the features of PalmOS was reasonable given that the PalmPilot was designed to do just a few simple tasks which do not require any advanced operating system features. The PalmPilot was designed as a viewing device, and as such, has little computing power to perform complex calculations or process large amounts of data. For simple games or viewing calendar data, the PalmPilot was by far the fastest and snappiest of the handhelds we tested. When we tried to perform a few tasks that required any reasonable amount of computing power, the PalmPilot ran out of steam.

EPOC32 is at the other end of the handheld operating system spectrum. This operating system is object-oriented, written in C++, and provides the kinds of features and architecture normally found in high-end desktop and minicomputer systems. Its message-passing architecture, client-server model and well-architected interfaces make it the technological leader in our research. In fact, while reading about the design of EPOC32, it sounded like we were reading about the Carnegie-Mellon Mach operating system. The system is so well designed that we wondered why the designers went to such lengths for an operating system that was primarily designed for embedded devices. Clearly, though, they were designing for the future by providing EPOC32 in this form. By designing most of the key features as services, they have enabled these pieces to be replaced or eliminated entirely. This allows EPOC32 to be constructed in a "building block" style, which can be custom fit to any processor or environment. For example, a system developer could replace the entire graphics subsystem with a custom replacement, or remove it entirely.

The EPOC32 device driver architecture is nicely layered to provide user and kernel mode device services with the benefits of memory protection, easier debugging and quicker time to market.

Windows CE is also a very nicely constructed operating system. Microsoft made the correct decision when it decided to rewrite Windows CE from scratch, eliminating any old 16-bit code and any dependencies on the older Windows 3.1 code. Microsoft’s decision to use the Win32 APIs for Windows CE was a critical decision that we believe will help insure the success of the Windows CE operating system. By including Win32, Microsoft enabled all of the other technologies that make the Windows operating system so attractive, such as ActiveX controls, the Microsoft Foundation Class libraries, Active Template Library, Telephony API, and standard file system and communications APIs. This decision also enabled the standard Windows development tools such as Developer Studio and the Windows C/C++ compiler to be used to develop software for the handheld PC. Windows CE SDKs are loaded as extensions to the standard Windows SDK, and allow applications to be written and debugged from within the same integrated development environment. This creates a seamless work area for software developers already skilled on the Windows platform, and allows applications originally written for 16 and 32-bit Windows to be ported quickly to the Windows CE platform. But two other important features of Windows CE help to make it the only logical choice in handheld operating systems. First, Windows CE’s affinity to Windows makes applications easy to use and the user interface “second nature” to most users. Whether you like the Windows user interface or not, it’s the most widely used and understood interface to the modern PC operating system. Second, the fact that Microsoft supplies the Windows CE operating system virtually guarantees that it will be successful. Windows CE is a part of Microsoft’s “Windows everywhere” mantra, and thus carries with it all of the good (and bad) connotations of Windows.

Developer Tools Wrap-up

PalmPilot developers have two basic choices in development tools, the Gnu compiler and associated tools, and the Metrowerks CodeWarrior development toolkit.

In June of 1998, 3COM assumed responsibility of the CodeWarrior development tools, finally providing developers with a single source of information for PalmPilot development. This was always a sore spot with developers, in that 3COM supplied the hardware while Metrowerks controlled the development tools. The two companies’ goals and directions did not always seem to mesh well. Developers who complained to 3COM were told that Metrowerks supplied the tools and there was not much that they (3COM) could do. Complaints to Metrowerks were often shifted to 3COM, with Metrowerks claiming that 3COM did not supply the proper documentation to them for the PalmPilot toolkit.

3COM was clearly interested in selling more PalmPilots, but Metrowerks held the keys to the PalmPilot's success by controlling the tools. This was an uncomfortable position for Palm Computing to be in, so the recent effort to combine the tools and hardware under one company was a positive step for PalmPilot development. Another positive step was the licensing by Metrowerks and 3COM of the former shareware product known as CoPilot. CoPilot is a PC-based PalmPilot emulator that allows programs to be developed and debugged on a PC running Windows or some flavor of Unix. CoPilot displays a bitmap picture of the PalmPilot and PalmPilot display in an emulation mode on the PC, allowing developers to see what their program will look like when it is run on the PalmPilot. CoPilot is now included as a standard part of the Metrowerks toolkit.

EPOC32 developers have only one choice for development tools, and this is clearly not a good situation. During our research, we saw references to Gnu tools that were being developed for EPOC32, but we could not find any of these tools available for download. EPOC32 developers must register for the EPOC32 SDK by sending a letter and MasterCard number to EPOC World in London, England. The address is EPOC World Administrator, Symbian Ltd., 16 Harcourt Street London W1H 1DS UK. The price is 175 pounds, or about $250 U.S. Updates to the SDK are available for downloading from the EPOC work site at http://developer.epocworld.com/.

The major problem is that all Psion toolkit and program development is centered in the United Kingdom. All major listed Psion ISVs as well as development resources such as consultants and software are located in the United Kingdom. The Psion platform is poorly supported, and the main channel for developer support seems to be email or public newsgroups. We could find no organized development or support effort for the Psion on any Web site.

Windows CE developers clearly are in the best position to obtain development tools and support. Microsoft supplies both the Windows CE operating system and the development platform, the Microsoft Developer Studio. Developer Studio contains Visual C++® and the Microsoft Foundation Class libraries, as well as a collection of excellent debugging tools and utilities. Developers already versed in the Visual Studio environment will find it easy to develop software Windows CE. The only new thing that developers will have the learn is the specifics of the subset of the Win32 APIs that are available in Windows CE.

Development tools and fixes are easy to obtain, and Microsoft provides a good source for them as part of the Microsoft Developer Network, MSDN. Updates are shipped at least every quarter to MSDN members and are also available for download from the MSDN Web site. The MSDN program also supplies updates and service packs for all Microsoft software as well as international versions of all operating systems, allowing developers to test their software on the latest Microsoft platforms. Microsoft also supplies the Windows CE Services, a set of connectivity and synchronization services for Windows CE systems to connect to a desktop system.

The Microsoft tools allow developers to create a consistent look-and-feel on the Windows CE devices using the Microsoft Foundation Class libraries and the standard Win32 APIs and libraries. The combination of proven, supported and readily available tools make the Windows CE platform and the Windows CE tools an unbeatable combination for software developers.

Applications Wrap-up

All of the systems we looked at contained an adequate suite of applications.

The PalmPilot comes with the smallest number of applications, consistent with its size and speed. For users that require quick access to their address book, calendar, to do list or memo pad, the PalmPilot is hard to beat. It turns on instantly from what appears to the user to be a power-off state, and in most cases retains the exact position where the application was at the time the PalmPilot powered off. The PalmPilot lacks any other type of useful applications, however, making it the low-end competitor in terms of built-in features and application support. This does not necessary make the PalmPilot a bad choice, however, if the user’s primary need is to view calendar entries or notes. There are hundreds of freeware and shareware applications available for the PalmPilot.

In fact, there are more applications written for the PalmPilot than any of the handheld devices we looked at. There are two reasons for this. First, the PalmPilot was the first handheld device to achieve any critical mass in the handheld market, and second, the availability of free Gnu development tools make the PalmPilot an attractive development platform for students and part time hackers. The PalmPilot's handwriting recognition program called *Graffiti* was the hardest to learn and the most annoying to use, frequently failing to recognize the characters entered. Some users we spoke to, however, loved Graffiti. It seemed to be one of those features that was either loved or hated, with no middle ground.

EPOC32 comes with a much larger suite of applications as part of the standard operating system. One of the reasons for this seems to be that the operating system supports extensive features not found in a lower-end device such as the PalmPilot, and to take advantage of these features, applications have to be written. Besides the standard group of five PIM applications (email, calendar, to do, memo pad, and address book) EPOC32 includes an advanced scheduling application called Agenda. This application not only provides basic “to do” functions and scheduling, but timed and un-timed entries, multi-day events, attachment of memos to events, and note-taking, all in one integrated application. EPOC32 contains an Internet email application with POP3 support, as well as a full-featured Web browser. These advanced features are not included in the PalmPilot. However, the PalmPilot was not designed with this type of connectivity in mind, and in that respect does not have to compare features one-to-one with EPOC32. The PalmPilot also does not include any database application, which we believe to be a necessary component of any handheld operating system. Although we have read that EPOC32 supports color displays, we have never encountered an EPOC32-based handheld device that contained a color display. Since the Psion device has a fairly large display, however, we found the visibility to be quite adequate. The Psion display was much more visible in sunlight than the display on the PalmPilot.

Although the Psion pen can be used to draw small pictures, the primary input comes from the small keyboard supplied with the Psion. We found the keys much too small to be used effectively. Users that have more to do than to take a few short abbreviated notes will find the Psion keyboard tedious to use and a limiting factor in acceptance.

Windows CE comes with a nice set of applications including Web browsers, spell checkers, POP3/IMAP4 Email applications, FAX applications, financial calculators, database utilities, PDF file viewers and screen capture utilities. Like the PalmPilot, users can find lots of freeware and shareware on the Web.

Vendors that make Palm-size PC and handheld hardware are always looking to differentiate their system from the other Palm-size and handheld PCs. Each vendor is free to package software that is above and beyond the normal suite of software. One of the most attractive software offerings for Windows CE is the inclusion of the reduced size versions of Word, Excel, PowerPoint and Access. These limited-function relatives of the most popular desktop office suite share data seamlessly with their desktop counterparts, and provide the user with familiar applications to use on the handheld device.

Another feature that makes the Windows CE platform so attractive is its affinity to the Windows operating system. Users are familiar with the Windows desktop paradigm, and users that already have Windows installed on their desktop systems will find the Windows CE devices easy to use. The Windows CE user interface is essentially the same user interface found on the user’s desktop machine, so the user can work without confusion on both systems. Operations such as highlighting, selecting and manipulating desktop icons and data are the same on both platforms. Because of the familiarity with Windows, and the ability to share data seamlessly with the world’s most popular office applications, we consider Windows CE to be the clear winner in the applications category. While the software offered by the PalmPilot and Psion is adequate, the software supplied with Windows CE-based systems provides users with a set of professional quality applications.

Connectivity Wrap-up

The PalmPilot was the “least connected” of the devices we examined. By least connected, we mean that is had the fewest options in terms of hardware and software to actually get connected. The PalmPilot does quite well with the cradle and serial cable, and in fact, provides reasonable performance in spite of the overhead introduced by the PalmPilot's serial link protocol. The PalmPilot optimizes its standard databases, removing fields that contain no data and packing the records into the database. This allows to PalmPilot to perform delta shipping, minimizing the synchronization time.

Serial communications is limited to 65K baud on the internal serial device, although the aggregate throughput is much lower. The serial device also connects to an attached modem, allowing HotSync to be performed with the desktop machine over a dial-up line. The protocol used on the dial-up line is the PalmPilot serial link protocol. In fact, the HotSync components don't know the difference between synchronization on a dial-up line and the serial cradle. The PalmPilot does not use a more reliable protocol such as PPP. For the casual home user that checks an occasional address or appointment in their handheld device, the PalmPilot offers the best all-around connectivity package. Fast, simple and easy to use, the PalmPilot connectivity makes the most sense for the casual or home user.

The PalmPilot's infrared port does not communicate with IrDA compatible devices, although we understand that Palm Computing will issue an upgrade later this year to provide that feature. The only infrared capability supported by the PalmPilot is the beaming of business card data from one PalmPilot to another.

Communications support for the PalmPilot is provided with a set of poorly documented APIs with limited operating system support and no adequate programming examples or tools.

EPOC32 has a great set of connectivity functions, including IrDA-compatible infrared support, serial communications, sockets, IMAP4 and POP3 mail, and fax services. In addition, Symbian's current strategy is to include support for the Wireless Application Protocols, or WAP, a suite of networking protocols which makes efficient use of mobile communications bandwidth. Psion also plans to support Bluetooth, a short-range radio device. EPOC32 supports a full IrDA protocol stack including the IrLMP, IrLAP, TinyTP, IrMUX and IrCOMM services. This allows EPOC32 systems to connect to any IrDA-compatible device including printers, networks, and other computer systems.

Since EPOC32 implements its networking systems as server components, developers are free to pick and choose the desired communications support to fit their needs. The EPOC32 services are well defined in the documentation, although sample code is hard to find.

EPOC32's serial engine contains integral support for XMODEM and YMODEM transfers, and emulates TTY and VT100 terminals.

EPOC32's socket API, like all its APIs, is object-oriented. EPOC also provides a standard C library that is layered over the top of the object-oriented socket library. These C APIs provide a true Berkley Standard Definition, or BSD C socket API, which provides support for applications that use the standard C sockets.

While EPOC32 certainly provides a rich set of connectivity, perhaps equal to or better than that of Windows CE, we feel that the designers of EPOC32 fell short in not providing the connectivity to the majority of Windows-based networks and clients. They also failed to correctly judge the importance of secure communications, and finally, the object-oriented design of the communications services makes them overly complex to implement, difficult to debug and unnecessarily slow.

Windows CE contains a rich set of connectivity features including support for Windows sockets, telephony API, or TAPI, Remote Access Services, or RAS, and serial communications.

The Windows CE networking stack support supports TCP/IP and IrDA communications via the Winsock interface. Client applications can connect to networks running IPX, NetBEUI or TCP/IP via a subset of the PPP protocol suite. Windows CE also supports the Serial Line Interface Protocol, or SLIP.

Windows CE supports ICMP, HTTP, and FTP protocols, as well as NDIS 4.0. Clients can access network files and network printers, and Windows CE supports secure sockets for secure communications.

Windows CE provides two high-level APIs, the WinInet API and the WnetAPI that together provide support for the Common Internet File System redirector and Private Communication Technology, or PCT. The redirector supports the Universal Name Convention, or UNC.

In our judgement, Windows CE's communication services provide the best all-around set of connectivity for the enterprise and e-business environment. Microsoft clearly spent a great deal of time and effort on the networking services of Windows CE, and they obviously saw the value in reliable and secure connectivity.

Conclusions

When we compared the major features that each of these devices provide, we concluded that Windows CE-based handheld devices were technologically superior in the areas of computing, networking, display technology, connectivity, usability, programming environment and ease of use. While each of the devices had their strengths and weaknesses, the Windows CE-based systems consistently provided better functionality in every category.

We conducted online interviews with 123 handheld users that had used or programmed at least one device in each category. We presented them with a questionnaire asking them to rate each of the categories for each type of device using a 0-100 scale. The results are present in Appendix A – Numerical Rating. In this category, the Windows CE device category scored approximately an 89, while the PalmPilot scored a 78. The Psion came in at a respectable 80.

Using the PalmPilot’s Graffiti handwriting recognition was more painful than the Palm-size PC’s JOT recognition system, but both were tedious and required several hours of training. We got cramped fingers from the repetitive motion required by both programs. Although the Psion comes equipped with a keyboard, the keyboard is much too small, requiring many special key combinations to enter data.

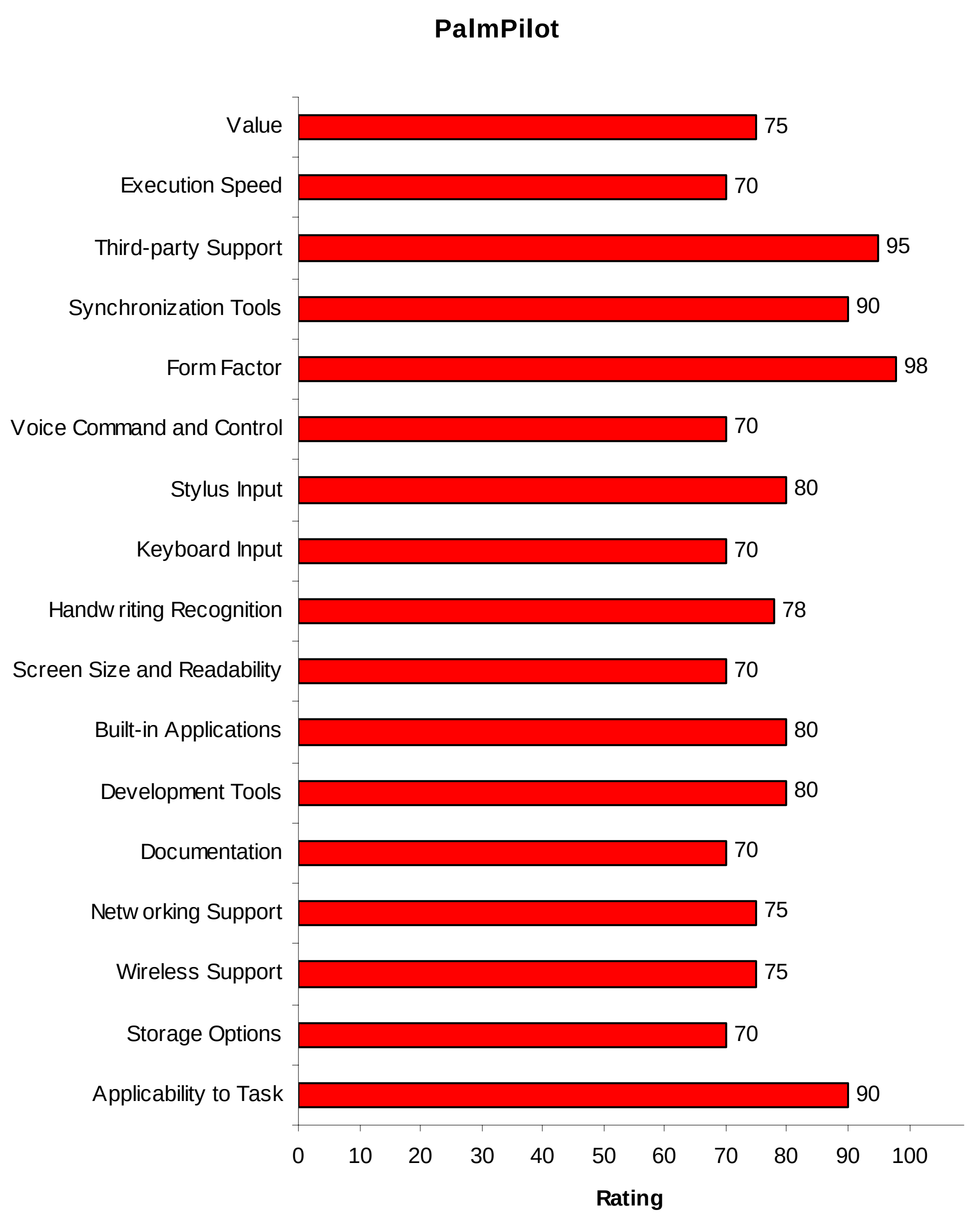
The PalmPilot was the least attractive, offering only basic services for the casual user. Its lack of a good operating system, poor connectivity, lack of good development tools, poor documentation, awful handwriting recognition, and tiny display make it tedious to use for any length of time. With a price of $399, the PalmPilot is almost as expensive as the more powerful Windows CE devices, yet lacks much of the functionality. Get this one for mom.

Although the EPOC32 operating system is nicely designed, it brings along too much baggage with it, causing it to require lots of memory for its bloated C++ applications and services. This powerful system needs to stay where it belongs, in the Pentium-class desktops. A huge impediment to the Psion's success is that the Psion is supported only in the United Kingdom, and support in the United States is virtually non-existent. Many of the applications written for the Psion are designed to run only in the United Kingdom, and are not worth much in the United States. An application that provides maps of London, for example, would not be useful in the United States. Proprietary SDKs and a complicated process to sign up for development tools serve to further restrict its acceptance.

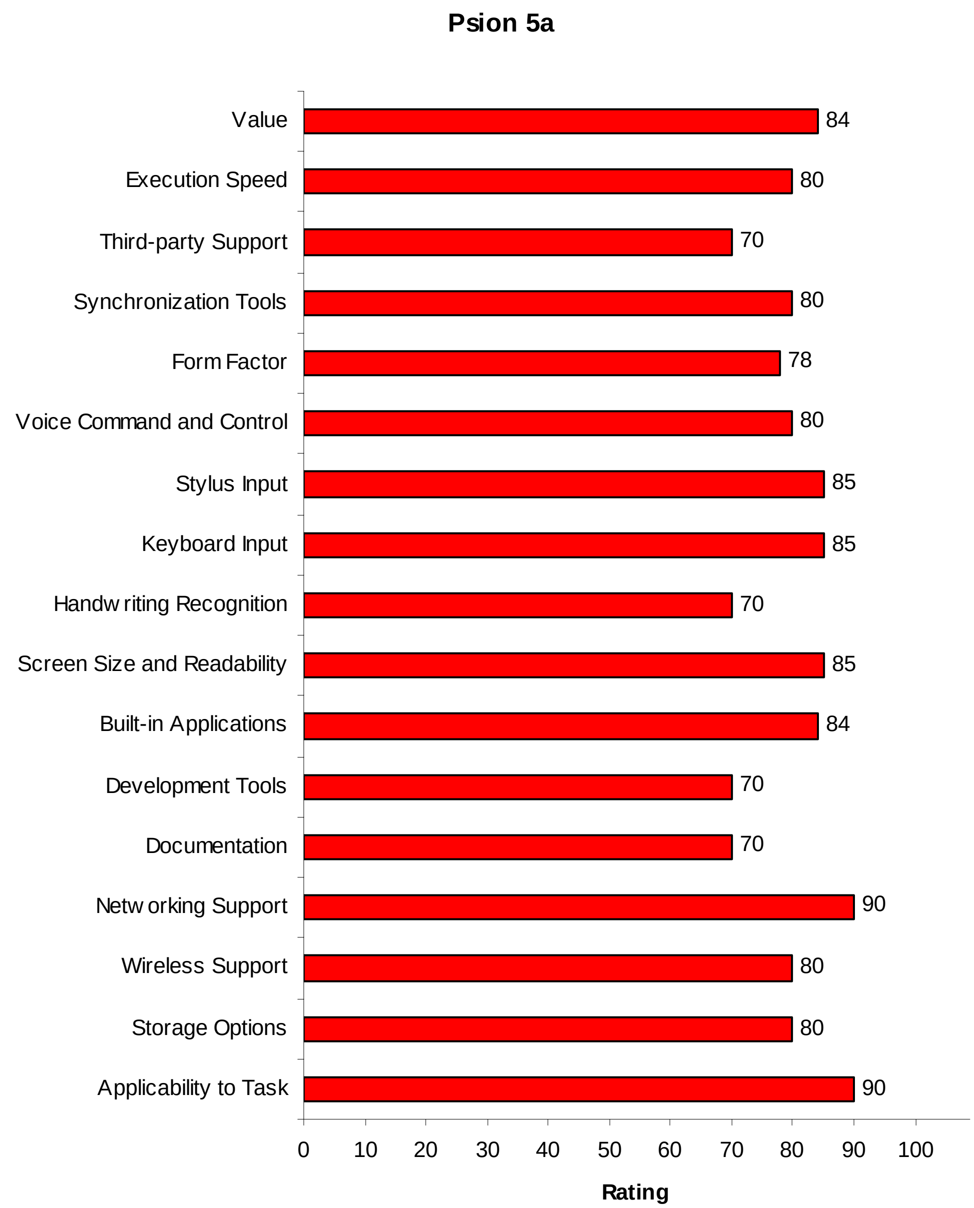
Windows CE brings together a fairly lean operating system, great tools, lots of developer support, terrific networking support, industry-standard network and connectivity protocols, and the obvious affinity to the most popular desktop operating system in the world. When you add to that the support of Microsoft, the result is an unbeatable solution that will be running the handhelds of tomorrow [2, 9, 24, 37]. Choosing Windows CE is not only the right choice; it's the only choice.

It should be noted that even though we loved the Windows CE operating system, we did not like the methods of input provided with any of the handhelds, including the Windows CE devices, with one exception – the NEC MobilePro 750. The NEC MobilePro 750 stands alone in the pack of Windows CE-based machines as the largest of the group. The MobilePro 750 has an eight-inch, 256-color screen with a keyboard that feels like a full size unit. We had no trouble entering data into a word processor, and data entry was no more tedious than using a standard desktop. The advantage is that the MobilePro weighs only one and one-half pounds and runs on batteries, and provides a built-in modem, compact flash slot, PCMCIA card slot, and external monitor connector.

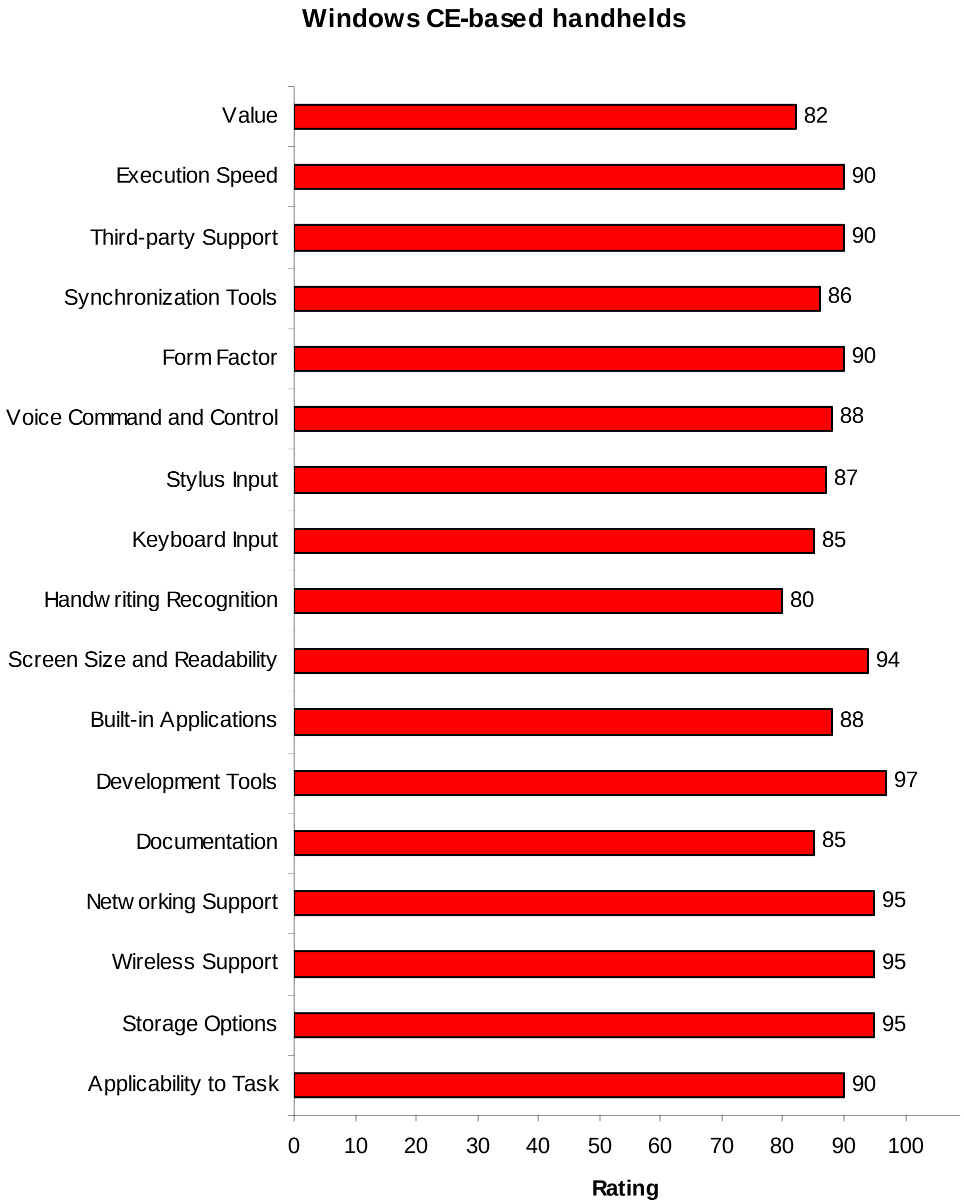
Appendix A – Interview Ratings



Score 78


Psion 5a
Value
Execution Speed
Third-party Support
Synchronization Tools
Form Factor
Voice Command and Control
Stylus Input
Keyboard Input
Handw riting Recognition
Screen Size and Readability
Built-in Applications
Development Tools
Documentation
Netw orking Support
Wireless Support
Storage Options
Applicability to Task
84
80
70
80
78
80
85
85
70
85
84
70
70
90
80
80
90
0
10
20
30
40
50
60
70
80
90
100
Rating

Score 80


Windows CE-based handhelds
Value
82
Execution Speed
90
Third-party Support
90
Synchronization Tools
86
Form Factor
90
Voice Command and Control
88
Stylus Input
87
Keyboard Input
85
Handwriting Recognition
80
Screen Size and Readability
94
Built-in Applications
88
Development Tools
97
Documentation
85
Networking Support
95
Wireless Support
95
Storage Options
95
Applicability to Task
90
0
10
20
30
40
50
60
70
80
90
100
Rating

Score 89

Appendix B - Handheld PC Comparison

Model	Casio E-10	Philips Nino 320	NEW MobilePro 750	Psion 3C	Psion 5	HP 620LX	PalmPilot III
Price (list)	$499	$499	$899	$299	$399	$799	$399
Display	240x320BW	240x320BW	640x480C	480x160BW	640x240BW	640x240C	160x160BW
ROM	8MB	8MB	16MB	2MB	6MB	10MB	1MB
RAM	4MB	8MB	16MB	2MB	4MB	16MB	2MB
Processor	NEC4111	NEC4111	NEC4111	NECV30H	ARM7100	SH-3	68328
Speed	49MHz	80MHz	80MHz	7.7MHz	18.4MHz	60MHz	16MHz
Op Sys	Win CE	Win CE	Win CE	EPOC16	EPOC32	Win CE	PalmOS
Size	5x3x.8	5x3x.75	1.4x9.6x5.4	1x6.5x3.5	1x6.7x3.5	1.5x7.8x4.1	4.7x3.2x.7
Weight (oz)	6.6	6.7	29	10	12.5	21	6.6
Keyboard	Soft	Soft	64-key	58-key	64-key	64-key	Soft
Voice	Yes	Yes	Yes	Yes	Yes	Yes	No
Pen	Yes	Yes	Yes	No	Yes	Yes	Yes
Battery	2 AAA	2 AA	Li-on	2 AA	2 AA	Li-on	2 AAA

Appendix C – Handheld Resources

PalmPilot Development Tools

PalmPilot SDKs http://www.palmpilot.3com.com/devzone/tools/sdks.html

PalmPilot 3.0 SDK http://www.palmpilot.3com.com/devzone/tools/sdk30.html

Conduit Development http://www.palmpilot.3com.com/devzone/conduits.html

PalmPilot Emulator http://www.palmpilot.3com.com/devzone/pose/pose.html

PalmPilot Debug ROM
http://www.palmpilot.3com.com/devzone/rom3/debugrom.html

Development FAQs http://www.palmpilot.3com.com/devzone/faqs/faqs.html

SDK Documentation http://www.palmpilot.3com.com/devzone/docs.html

PalmPilot Whitepapers http://www.palmpilot.3com.com/devzone/docs.html#white

PalmPilot Developer Downloads
http://www.palmpilot.3com.com/devzone/dzdownld.html

PalmPilot Software and Downloads

CNET Downloads PalmPilot Selections for PC
http://www.download.com/PC/FrontDoor/0,1,0,0-335,00.html?st.dl.fd.cats.cat335?tt.palm.links.txt.cat335

CNET Downloads PalmPilot for Mac
http://www.download.com/Mac/FrontDoor/0,1,0,0-335,00.html?st.dl.fd.cats.cat335?tt.palm.links.txt.cat335

CNET Downloads PalmPilot Guide for PC
http://www.download.com/PC/EdFeature/CurrentFeature/0,15,0-509-1,00.html?tt.palm.links.txt.psg

CNET Downloads PalmPilot Guide for Mac
http://www.download.com/Mac/EdFeature/CurrentFeature/0,15,0-509-1,00.html?tt.palm.links.txt.psg

Bill's PalmPilot Resource Page http://www.electronhut.com/pilot/

CNET's Download.com
http://www.download.com/PC/EdFeature/CurrentFeature/0,15,0-509-1,00.html?palm.links.txt.psg

DaggerWare Home Page http://daggerware.com/

EuroCool http://www.eurocool.com/

Lending Library http://www.macduff.net/

MemoWare http://www.memoware.com/

Palm Zone http://www.palmzone.com/

The PalmPiloteer http://www.pmn.co.uk/

PalmPilotGear H.Q. http://www.pilotgear.com/

PalmPilot Light http://the-tech.mit.edu/pilot/

PalmPilot OS http://www.palmos.com/

PalmPilot Zone http://www.palmpilotzone.com/

PalmTastik http://www.palmtastik.com/

QuickFiles.com http://www.quickfiles.com/key_search.pl?platform,palm

Ray's PalmCentral Software Archive http://www.palmcentral.com/

RoadCoders Resources for PalmPilot http://www.roadcoders.com/pilot/index.html

Tap! http://www.tapmagazine.com/

Tucows http://www.pdacentral.com/

ZDNet's PalmPilot Software http://www.palmpilotsoftware.com/

PalmPilot Commercial Applications

CNET's BuyDirect
http://www.buydirect.com/Windows/Product/List/0,11,18,01.html

Computer Discount Warehouse http://www.cdw.com/3com/pda.asp?source=PAP

MicroWarehouse
http://www.warehouse.com/oasis/bin/searchinterface.dll?num_rows=100&fieldName=section_path&old_value=Robotics&value=computers+not+hitachi&longbar=http://images.warehouse.com/www/kiosks/3com/USRPalmPilot.gif&result_template=usrsearch.tem

MyPilot http://www.mypilot.com/

PalmPilotGear http://www.palmpilot.3com.com/software/index.html

PalmPilot Software http://www.palmpilot.3com.com/catalog/index.html

PalmPilot On-line Catalog http://www.palmpilot.3com.com/catalog/index.html

PDA Page http://www.pdapage.com/

PalmPilot Accessories and Add-ons

Calvin's PalmPilot FAQ http://www.pilotfaq.com

The Gadgeteer http://www.the-gadgeteer.com/palmos.html

HandJive http://www.handjivemag.com/index.shtml

Link Archive http://www.cyberramp.net/~dmbrowne/pilot.htm

Macduff http://www.macduff.net/

MemoWare http://www.memoware.com/

PalmPilot Files http://www.palmpilotfiles.com/

PalmPilot Library http://www.pilotlibrary.org/

PalmPilot Programs http://www.seimitsu.demon.co.uk/pilotprogs.htm

PalmPlanet http://palmtopplanet.interspeed.net/palm/palmplanet.shtml

PalmTastik http://www.palmtastik.com/

The Pickled PalmPilot http://www.pickled.com/

PilotGear http://www.pilotgear.com/

SmallWare Games http://www.smallware.com/

Tim Warner's PalmPilot Page http://www.timwarner.com/palmos/index.html

PalmPilot Books

Stacey's Book Store http://www.staceys.com/

IDG Books http://www.idgbooks.com/cgi-bin/db/fill_out_template.pl?idgbook:1-55828-586-5:book-idg::uidg709

Handheld Computer Almanac http://www.cdpubs.com/hhca/hhca.html

PalmPilot Companion
http://merchant.superlibrary.com:8000/catalog/mcp/PRODUCT/PAGE/07897/bud/0789712482.html

Computer Literacy Bookstore
http://www1.clbooks.com/asp/sweepstakes/Sweepstakes.asp

PalmPilot Newsgroups

Pen Computing news:comp.sys.pen

Palmtop Computers news:comp.sys.palmtops

Handheld Computers news:comp.sys.handhelds

PalmPilot news:alt.comp.sys.palmtops.pilot

PalmPilot news:comp.sys.palmtops.pilot

PalmPilot Online Publications

Handheld Systems http://www.cdpubs.com/hhsj/hhsj.html/

The Piloteer Magazine http://www.pmn.co.uk/

TapOnline http://www.tapmagazine.com/

The Palm Zone http://www.palmzone.com/

Mobilis Magazine http://www.volksware.com/mobilis

PalmPilot Press Clips

Supercharging your handheld by Steve Kichen
http://www.forbes.com/tool/html/98/oct/1005/side1.htm

It's a Phone! It's a PalmPilot! Qualcomm merges two devices into one that runs PalmOS programs
http://www.zdnet.com/pcweek/stories/news/0,4153,355307,00.html

PDAs: The battle is joined. In this corner, the champion, PalmPilot. In that corner, a bunch of bruisers with styles all their own
http://sales.supersites.net/pcdsn2/pc9809/fd1.htm

Netscape Communicator 4.5 adds Palm Computing Platform email and address book synchronization
http://www.zdnet.com/products/stories/reviews/0,4161,2137679,00.html

Just How Helpful Are Handhelds?
http://cnn.com/TECH/computing/9809/09/palmtops.idg/

Pilots Help Doctors Navigate Their Reesidency
http://www.nytimes.com/library/tech/98/07/cyber/articles/19pilot.html

Love That Pilot (PC Magazine)
http://www.zdnet.com/pcmag/issues/1713/325707.htm

Synapse Pager Card: Silicon Bundle of Joy (Tapped)
http://www.tapped.com/thepager.htm

Mac's Best Friend (Macworld Online)
http://macworld.zdnet.com/pages/august.98/Feature.4408.html

We Were All Beaming About Our Palm Pilots
http://www.techweb.com/se/techsearch.cgi?action=View&VdkVgwKey=%2E%2E%2F%2E%2E%2Fdata%2Ftechsearch%2F1998%5F%5F%5B16821%5D&DocOffset=6&DocsFound=23&Collection=techwebnews&Collection=techinvestor&Collection=coll1998&ViewTemplate=cmpview%2Ehts&&publication=

The PalmPilot Sequel is a Hit
http://www.pathfinder.com/fortune/digitalwatch/0413fil3.html

Piloted to the Promised Land
http://online.guardian.co.uk/technology/archive/889628508-second.html

Editor's Choice: 3COM PalmPilot Professional
http://www.zdnet.com/pcmag/features/handheld_pc/rev10.html

Accessing the Net Right in Your Pocket
http://www.techweb.com/wire/news/1997/10/1007att.html

Best of 1997 (PC Magazine)
http://www.zdnet.com/pcmag/special/bestof97/rev5.htm

The Ever-evolving PC
http://www.zdnet.com/pcmag/insites/miller_print/mm980106.htm

Geek's Guide to Gadgets
http://www.cnet.com/Content/Reviews/Special/Gizmos/ss08.html

Hot Trends to Bet On http://pathfinder.com/fortune/1997/971110/ten3.html

Mobile Trends (Computer Shopper)
http://www.zdnet.com/cshopper/content/9712/cshp0173.html

MVP Award Winners (PC Computing)
http://www.zdnet.com/pccomp/features/excl0198/mvp97/systems/3com_palmpilot.html

Secrets of the PalmPiloteers
http://www.cnet.com/Content/Features/Howto/Pilot/index.html

PalmPilot User Groups

Canada's Premier PUG http://www.kpoole.com/pug

ChicagoLand Palm Users Group mailto:davidh@aramis-inc.com

Colorado PalmPilot User Group http://www.creativeconsulting.com/coppug.htm

New England Palm Users Group http://www.bnug.org/ne-palm

New York Palm Users Group http://www.nypalm.org/

North Alabama Palm Users Group mailto:faspina@mindspring.com

Philadelphia PalmPilot User Group http://www.pond.com/~azion/pilot.html

WebBlazers http://www.MAPUG@webblazers.com/

Psion Development Tools

Psion EPOC World http://www.software.psion.com/EPOCWorld/

OPL programming manual
http://www.software.psion.com/EPOCWorld/binpub/oplhtml/opl/op-toc.html

Psionics files http://www.gold.net/users/cdwf/psion/psionics/

PPC (OPL software development) http://members.xoom.com/PerssonPC

Psion Software and Downloads

Ad Bosch http://www.adbosch.demon.nl/

Adrian Harper http://www.ady.net/

Daniel Pfund http://www.geocities.com/SiliconValley/8130/

FatCatz http://www.fatcatz.tm/

Jason Kneen http://www.kneen.demon.co.uk/

Jon Read http://www.ecs.soton.ac.uk/~jnr95r/

Marc de Oliveira http://www.deoliveira.dk/pythia/

Mark Wheadon http://gos.ukc.ac.uk/cgi-bin/hpda/mcw/pats.html

Mat Ripley http://www.salted.clara.net/

Matt Thomas http://home.clara.net/mattthomas/

Mayur Shah http://www.btinternet.com/~devrag.software

Mick Jagger http://www.palmscape.com/Products/Yartsie/

Peter Csutora http://members.tripod.com/~csutora/

Russ Spooner http://www.labrat.demon.co.uk/

Simon Berridge http://ourworld.compuserve.com/homepages/SimonBerridge/

Steve Litchfield http://3lib.ukonline.co.uk/

Alan Richey http://ourworld.compuserve.com/homepages/alanrichey

Phile Spencer http://homepages.nildram.co.uk/~nicko/

Adrian Burgess http://www.artamedia.com/

Psion Commercial Applications

Geofox http://www.geofox.com/

Yellow Computing Company http://www.yellow.de/

New World Technologies http://www.nwt.com/

Wodget Software http://www.widget.co.uk/

Palmtop B V http://www.palmtop.nl/

RegNet http://www.swregnet.com/

Palmscape http://www.palmscape.com/

Psion Accessories and Add-ons

Psion http://www.symbian.com

Psion Books

Psioneer Magazine http://www.pmn.co.uk/frameson/psioneereertrial.html

Psion Newsgroups

Psion Programming news:comp.binaries.psion

Security news:comp.security.psion.announce

System News news:comp.sys.psion

Applications news:comp.sys.psion.apps

Communications news:comp.sys.psion.comm

Marketing News news:comp.sys.psion.marketplace

Psion Online Publications

Psion FAQs http://www.geocities.com/SiliconValley/8130/faq.htm

Palmtop U.K. Magazine http://www.palmtop.co.uk/

Psion Monthly Noticeboard http://www.pmn.co.uk/

Psion software database http://www.psion.priv.at/psion/psiidb.htm

Psion Press Clips

Symbian gains approval from the European Commission
http://www.symbian.com/news/press/1998/pr980812.html

Symbian to cut wires http://www.symbian.com/news/press/1998/pr980724.html

Ericsson, Nokia, and Psion partner to create major engine of growth for Wireless Information Devices http://www.symbian.com/news/press/1998/pr980624d.html

Psion Software joins Wireless Application Protocol Forum
http://www.symbian.com/news/press/1998/pr980615.html

Connecting with you. Everywhere.
http://www.symbian.com/news/press/1998/pr980603.html

Psion Software and Sendit announce Smart Wireless Internet Messaging and Content Delivery http://www.symbian.com/news/press/1998/pr980325.html

Philips Synergy makes Mobile Phones Smarter with Psion
http://www.symbian.com/news/press/1998/pr980305.html

Lotus Notes Takes EPOC Platform into Corporate Markets
http://www.symbian.com/news/press/1998/pr980219.html

EPOC32 First mobile platform to support MultiMedia card
http://www.symbian.com/news/press/1998/pr980127.html

Psion User Groups

Psion User Group http://www.user-group.co.uk/
Psion User Group, Germany http://www.wave.cube.net/puc.html

Windows CE Development Tools

Windows CE Logo Program
http://www.microsoft.com/windowsce/logo/default.asp

Microsoft Developer Network http://www.microsoft.com/msdn/

Mobile Channels Development Kit
http://www.microsoft.com/workshop/delivery/mobile/mcdk_in.asp

Embedded Systems http://www.microsoft.com/windowsce/embedded/default.asp

Windows CE Technology Partners
http://www.microsoft.com/windowsce/embedded/partner/default.asp

Windows CE Developer News
http://www.microsoft.com/windowsce/community/newsbyemail.asp

Windows CE Developer Downloads
http://www.microsoft.com/windows/downloads/default.asp?site=cehome&openmenu=cehome

Visual C++ Web Site http://www.microsoft.com/visualc/

Visual Studio Web Site http://www.microsoft.com/vstudio/

Windows CE Software and Downloads

CompUSA www.compusa.com

J & R Music www.jrmusic.com

Micro Center www.microcenter.com

Compaq PC Companion Backup and Restore
http://www.compaq.com/products/handhelds/pccompanion/restore.html

H/PC Explorer for Microsoft Windows CE 1.0
http://www.microsoft.com/windows/downloads/contents/Products/CEHPCExplorer/default.asp?custarea=bus&site=cepalmsize&openmenu=&highlighteditem=

Handheld PC File Converter for Microsoft Windows CE 1.0
http://www.microsoft.com/windows/downloads/contents/Products/CEHPCFileConv/default.asp?custarea=bus&site=cepalmsize&openmenu=&highlighteditem=

Hewlett Packard Rex Update
http://www.hp.com/handheld/getting_help/promotions/rex/rex_intro.html

Microsoft Expedia Pocket Streets for Windows CE 2.0
http://www.microsoft.com/windows/downloads/contents/Products/CEStreets/default.asp?custarea=bus&site=cepalmsize&openmenu=&highlighteditem=

Microsoft SDK for Java for Windows CE
http://www.microsoft.com/windows/downloads/contents/Betas/CEjavaSDKbeta/default.asp?custarea=bus&site=cepalmsize&openmenu=&highlighteditem=

Microsoft Pocket Internet Explorer 1.1
http://www.microsoft.com/windows/downloads/contents/Web/CEPocketIE/default.asp?custarea=bus&site=cepalmsize&openmenu=&highlighteditem=

Power Toys for Windows CE 2.0
http://www.microsoft.com/windows/downloads/contents/PowerToys/CEHPCPwrToys2/default.asp?custarea=bus&site=cepalmsize&openmenu=&highlighteditem=

Microsoft International Glossaries for Windows CE 2.0
http://www.microsoft.com/windows/downloads/contents/Other/CEglossaries/default.asp?custarea=bus&site=cepalmsize&openmenu=&highlighteditem=

Microsoft Free Visual Basic Downloads For Windows CE
http://www.microsoft.com/vbasic/download/

Microsoft Free Visual C++ Downloads for Windows CE
http://www.microsoft.com/visualc/download/

Microsoft Windows CE 2.0 SDK
http://www.microsoft.com/windows/downloads/contents/Other/CEHPCplatformSDK/default.asp?custarea=bus&site=cepalmsize&openmenu=&highlighteditem=

Windows CE Commercial Applications

Computer Discount Warehouse www.cdw.com

PC Connection www.pcconnection.com

Creative Computers www.creativecomputers.com

Multiple Zones www.multiplezones.com

Programmer's Paradise www.programmersparadise.com

Mobile Planet www.mobileplanet.com

Pro Advantage www.provantage.com

CE Central www.cecentral.com

Windows CE Accessories and Add-ons

Best Buy www.bestbuy.com

CompUSA www.compusa.com

Computer City www.computercity.com

DataVision www.datavision.com

Electronics Boutique www.electronicsboutique.com

J&R Music www.jrmusic.com

Office Depot www.officedepot.com

OfficeMax www.officemax.com

Sharper Image www.sharperimage.com

Staples www.staples.com

Mobile Planet www.mplanet.com

Mobilesoft www.mobilesoft.com

Windows CE Books

Microsoft Windows CE for the Handheld PC www.microsoftpress.com

Microsoft Windows CE Programming Guide www.microsoftpress.com

Inside Microsoft Windows CE www.microsoftpress.com

Windows CE Applications Programming www.macmillan.com

Windows CE Newsgroups

Windows CE News news:comp.os.mswindows.ce

Microsoft Windows CE News news:Microsoft.public.windowsce

Windows CE Online Publications

Windows CE Developer News
http://www.microsoft.com/windowsce/community/newsbyemail.asp

Comm Link http://www.microsoft.com/windowsce/hpc/columns/commlink.asp

Embedded Space
http://www.microsoft.com/windowsce/embedded/columns/embeddedspace.asp

From the Road http://www.microsoft.com/windowsce/hpc/columns/road.asp

Help Desk http://www.microsoft.com/windowsce/hpc/columns/helpdesk.asp

Market Place http://www.microsoft.com/windowsce/columns/marketplace.asp

Mobile Musings http://www.microsoft.com/windowsce/ppc/columns/musings.asp

Product Spotlight
http://www.microsoft.com/windowsce/hpc/columns/spotlight.asp

Windows CE Solutions
http://www.microsoft.com/windowsce/hpc/columns/solutions.asp

Windows CE Tips and Techniques
http://www.microsoft.com/windowsce/hpc/columns/tips.asp

Windows CE Press Clips

Microsoft Announces Handheld PC Professional Edition
http://www.microsoft.com/windowsce/hpcpro/basics/overview/default.asp

Microsoft, Hitachi Announce Strategic Alliance on Windows CE Platform
http://www.microsoft.com/presspass/press/1998/Jun98/HitchPR.htm

Microsoft Product Announcements Reflect Changing Face of Enterprise Computing http://www.microsoft.com/presspass/press/1998/Jun98/pcexpopr.htm

Microsoft, BellSouth Wireless Data Team to Enable Windows CE for Two-Way Wireless Data
http://www.microsoft.com/presspass/press/1998/May98/BellSopr.htm

Paul Maritz Addresses Windows CE Developers Conference, Underscores Commitment of Microsoft, Vendors to Platform
http://www.microsoft.com/presspass/press/1998/Apr98/MaritzPr.htm

Software Is Shaping Consumer Electronics as Devices Go Digital, Says Microsoft's Gates
http://www.microsoft.com/presspass/press/1998/Jan98/CESkeypr.htm

Intel, Microsoft Join Forces to Driver in-Car Computing Solutions
http://www.microsoft.com/corpinfo/press/1998/Apr98/intlmspr.htm

Microsoft Announces Auto PC, PC Companion Powered by Windows CE 2.0
http://www.microsoft.com/corpinfo/press/1998/Jan98/AutoPCpr.htm

Microsoft Ships Windows CE 2.0 for the Handheld PC Supporting the Japanese Language http://www.microsoft.com/corpinfo/press/1998/Mar98/Jpnv2pr.htm

Handheld PCs Powered by Windows CE Achieve Major Milestone
http://www.microsoft.com/corpinfo/press/1998/Jan98/HandPCpr.htm

Microsoft and 3Com Resolve Product Naming Dispute
http://www.microsoft.com/presspass/press/1998/Apr98/3COMPr.htm

Third Windows CE Conference Draws 2,000 Developers for Newest Tools, Insight From Microsoft and Vendors
http://www.microsoft.com/presspass/press/1998/Apr98/2000DVPR.htm

Microsoft Introduces PalmPC, PC Companion Powered by Windows CE 2.0
http://www.microsoft.com/presspass/press/1998/Jan98/PalmPCpr.htm

Microsoft, Sega Collaborate on Dreamcast: The Ultimate Home Video Game System http://www.microsoft.com/corpinfo/press/1998/May98/Segagmpr.htm

TCI, Microsoft Finalize Licensing Agreement for Windows CD On Advanced Set-Top Terminals http://www.microsoft.com/corpinfo/press/1998/May98/TCIpr.htm

Windows CE Palm-size PC Suppliers

Casio E10 and E11 http://www.casio-usa.com/e10/

Everex Freestyle http://www.freestyle.everex.com/

Palmax PD-300 http://www.palmax.com.tw/

Philips Nino 300 and 320 http://nino.philips.com/

Uniden UniPro PC-100
http://www.uniden.com/docs/product/prdetail.asp?prodcode=PC100&prodcat=18

Windows CE User Groups

www.microsoft.com/windowsce/embedded/news/thirdparty/default.asp

Index

3COM, ii, iv, 21, 22, 23, 24, 25, 26, 67, 68, 116, 154
Active Template Library, 75, 87
ActiveSync, 49, 50, 123
ActiveX, 75, 83, 87
Advanced Risc Machines, 34
AMX, 27, 60
ARM7, 34, 69, 72, 119
AutoPC, 41
Bluetooth, 119
client-server architecture, 36
CodeWarrior, 63, 80
COM, 39
Common Object Model, 75
Conduit, 80, 149
conduits, 30
Copilot, 83
CreatorID, 63
Delayed Function Call, 37, 71
DLL, 38, 78
DragonBall, 24, 60
dynamic link libraries, 45
EIKON, 38, 73, 119
EPOC, 33, 34, 35, 36, 37, 38, 39, 69, 73, 84, 118, 119, 158, 168
EPOC32, 35, 36, 37, 69, 71, 72, 73, 77, 86, 117, 118, 119, 158
Franklin RexPro, 9
FTP, 12, 46, 120, 122
GIF, 118
Gnu, 12, 35, 81, 84
Gnu tools, 81
Graffiti, 29
Graphics Device Interface, 72
Hawkins, Jeff, 21
Hewlett Packard, 41
Hitachi, 43
HotSync, 26, 30, 31
IBM, 23
Infrared Data Association, 12, 116
IPC, 44
IPX, 88, 120
IrComm, 117, 119, 121
IrDA, 12, 46, 116, 118, 119, 120, 121
IrLAP, 119

IrLMP, 119
IrMux, 119
IrSock, 120, 121
Java virtual machine, 64
JVM, 12, 64, 68, 73
Kadak Products Ltd, 27
Mach, 36, 69, 71
Microsoft Foundation Classes, 75
MIPS, 43, 72, 74, 77, 170
MIPS 4101, 74
Motorola, 24
mutex semaphores, 45
NetBEUI, 88
New Kernel, 44
OPL, 33, 35, 84, 85, 155
OPL for, 33
OPP, 84, 85, 86
Organiser, 34
Organiser II, 33
Palm Computing, 21
PalmOS, 27, 28, 29, 60, 61, 62, 66, 80, 83, 116, 153, 169
PalmPC, 53
PalmPilot, ii, iii, iv, 10, 21, 22, 23, 30, 31, 32, 53, 170
PalmPilot-III, 22
PDA, 44
Pegasus, 40
persistent store, 76
PowerPC, 43
PPP, 46, 118, 120
priorities, 76
Protected Server Library, 78
Psion, 8, 33, 53
Psion Link Protocol, 39, 118
Psion Software, 35
Pulsar, 40, 44
RAS, 14, 46, 88, 120, 121
registry, 79
RPC, 77
Satellite Forms, 83
SLIP, 88, 120
SMS, 118
StrongARM, 43
SuperH RISC, 74
Symbian, 35, 84, 119, 158
Symbol Technologies, 25, 116
synchronization, 30

TAPI, 15, 46, 88, 122
threads, 36
Translation Lookaside Buffer, 77
TrueType, 47
UART, 15, 119
UNICODE, 47
Universal Naming Convention, 123
Win32 APIs, 75
WinINet, 122
WinPad, 40, 44
Winsock, 16, 46, 88
Wireless Application Protocol, 119, 158
WorkPad, 23

References

[1.] "About Palm Computing." 1998. Online posting. www.3com.com/newspromo/corporate/3com.html. Internet. 23 September 1998.

[2.] "Anheuser-Busch Signs With Wright Strategies." 24 July 1998. Online posting. www.wrightstrat.com/news/main.html. Internet. 23 September 1998.

[3.] Badrinath, R. and and Welling, G. "Event Delivery Abstractions for Mobile Computing". Rutgers University. June 1998.

[4.] Badrinath, R., Acharya, A., and Imielinski, T. "Impact of mobility on distributed computations". *Operating Systems Review*, 27(2):15–20.

[5.] Baker, Jeff. *Windows CE Programming – Developing Applications for the Handheld PC.* Indiana: Macmillan, 1998.

[6.] Boling, Douglas. "Minimizing the Memory Footprint of Your Windows CE-based Program." *Microsoft Systems Journal* May 1998: 37-50.

[7.] Boling, Douglas. (1998). *Programming Windows CE*. Washington. Microsoft Press, 1998.

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

[9.] Cummiskey, Major James. "Semper Mobilis." *Handheld PC Magazine* July/August 1998: 20-23.

[10.] Dao, Jeff. "Handwriting Recognition Comes of Age." *Windows CE Tech Journal* March 1998: 10-13.

[11.] "EPOC Overview" 29 September 1998. Online posting. www.symbian.com/tech/papers/overview/overview.html. Internet. 5 October 1998.

[12.] Gareau, Jean L. "Porting MFC Applications to Windows CE." *Windows CE Tech Journal* September 1998: 31-37.

[13.] Gunasekara, Oliver. "Developing a digital cellular phone using a 32-bit Microcontroller". 29 September 1998. Online posting.

www.arm.com/Documentation/WhitePapers/CellPhone. Internet. 5 October 1998.

[14.] Hall, Rich. "NEC's MobilePro 750C." *Handheld PC Magazine.* July/August 1998: 6-10.

[15.] Hall, Rich. "Off The Shelf and In Your Palm – The Cassiopeia E-10 Palm-size PC." *Handheld PC Magazine*. July/August 1998: 12-15.

[16.] Hazzah, Karen. *Writing Windows VxDs and Device Drivers*. Kansas: R&D Publications, 1995.

[17.] "Intel, Microsoft Join Forces to Drive in-Car Computing Solutions." 7 April 1998. Online posting. www.microsoft.com/presspass/press/1998/Apr1998/intlmspr.htm. Internet. 23 September 1998.

[18.] Jones, Anthony. "Windows CE 2.0 Networking Offers Exciting Mobile Computing Possibilities." *Microsoft Systems Journal.* May 1998: 53-68.

[19.] Liming, Sean. "Device Driver Development for Windows CE." *Windows CE Tech Journal.* March 1998: 18-25.

[20.] Lorch, Jacob R. and Smith, Alan Jay. "Reducing Processor Power Consumption by Improving Processor Time Management in a Single-User Operating System" in *Proceedings of The Second Annual International Conference on Mobile Computing and Networking*. November 1996.

[21.] Mann, Steve. "Palm Ups the Ante with Pilot." *PDA Developer*. March/April 1996.

[22.] Milne, George et al. "Microcontroller Design Advantages for Portable Computing". 29 September 1998. Online posting. www.computer.org/micro/mil1997/m4049abs.htm. Internet. 5 October 1998.

[23.] *Developing PalmOS Applications.* Metrowerks, 1996.

[24.] "Microsoft, BellSouth Wireless Data Team to Enable Windows CE for Two-Way Wireless Data." 12 May 1998. Online posting. www.microsoft.com/presspass/press/1998/May98/BellSopr.htm. Internet. 23 September 1998.

[25.] *Microsoft Windows CE Programmer's Guide.* Washington: Microsoft Press, 1998.

[26.] Murray, John. *Inside Microsoft Windows CE.* Washington: Microsoft Press, 1998.

[27.] 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.*

[28.] O'Hara, Robert. *Introducing Microsoft Windows CE for the Handheld PC.* Washington: Microsoft Press, 1998.

[29.] Ottey, Sean. "Windows CE and the Object Store API." *Windows CE Tech Journal*. September 1998: 18-20.

[30.] "The Palm Operating System – Technology Overview." *PDA Developer*. March/April 1996.

[31.] Puque, David. *PalmPilot: The Ultimate Guide*. O' Reilly and Associates, 1998.

[32.] Radtke, Bruce. "Windows CE WIN32 API Programming". *Dr. Dobbs Journal*. April 1998: 62-70.

[33.] 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.

[34.] Sbihli, Scott. "PalmPilot Development Tools." *PDA Developer*. July/August 1997.

[35.] Seybold, Andrew. "The Marriage of Communications and Computing." *Handheld PC Magazine*. July/August 1998: 55-58.

[36.] Slack, Stu. "Extending Your Desktop With the Pilot." *PDA Developer*. September/October 1996.

[37.] Spooner, John G. "Windows CE to go wireless". 2 October 1998. *PCWeek Online*. www.zdnet.com/pcweek/news/0928/02ece.html. Internet. 5 October 1998.

[38.] Stoner, Greg. "MIPS - The Premier Architecture for Windows CE." *Windows CE Tech Journal.* September 1998: 52-53.

[39.] Stroustrup, Bjarne. *The C++ Programming Language.* New York: Addison Wesley, 1996.

[40.] Tanenbaum, Andrew S, Albert S Woodhull. *Operating Systems, 2nd Edition.* New Jersey: Simon and Schuster, 1996.

[41.] "The Ultimate Personal Peripheral" 10 March 1998. Online posting. http://www.zdnet.com/products/stories/reviews/0,4161,281508,00.html. Internet. 23 September 1998.

[42.] Thielen, David, and Bryan Woodruf. *Writing Windows Virtual Device Drivers.* New York: Addison-Wesley, 1993.

[43.] Wall, Thomas B. "Windows CE in Test and Measurement: A Case Study." *Windows CE Tech Journal.* September 1998: 8-15.

[44.] Yao, Paul. "Microsoft Windows CE 2.0: Its Not Just for Handheld PCs Any More." *Microsoft Systems Journal*. May 1998: 19-34.

[45.] Zhao, X., Castelluccia, C., Baker, M. *Flexible Network Support for Mobility* in *Proceedings of The Second Annual International Conference on Mobile Computing and Networking*. November1998.