FedExDB Shipping Rate Calculator

Authors: Jonathan Tang, Joseph Wong, & Albert Chan

Introduction

As more and more people start using the Internet, many businesses soon discovered the advantages of an economy through e-commerce. Prior to the advent of the e-commerce and the dot com economy, many businesses did not have the opportunity to expose their products worldwide. With this new emergence of the e-commerce revolution, people are starting to purchase products over the Internet. With more and more people making purchases, businesses would require a more structured way of shipping products. To fulfill their shipping needs, they have depended upon the services of couriers, i.e. FedEx.

To simplify the lives of businesses and individuals, we have developed an application, called FedExDB, that calculates the shipping rate of packages in the state of California, given the weight and approximate distance of destination. Our application provides shipping rates for two FedEx services, the 2Day and the Express Saver services. The application also has a feature where the user can place a package on an infrared enabled weight scale and have the weight scale beam the weight to the palm pilot. Along with the shipping rate calculation, palm pilot users also have the opportunity to print shipping labels through the palm pilot’s infrared port. To keep the user updated with the current FedEx shipping rates, the application’s database files can be updated with a simple program developed with Perl scripts.

The idea of updating through a wireless HotSync maybe a possible feature in future releases of FedExDB. So far, palm pilots have the ability to HotSync wirelessly with infrared. Infrared is a short range interface where both infrared devices must transmit within a straight path of approximately five to six feet. Any type of interference will disrupt the data transmission. However, with the ability to HotSync wirelessly with Bluetooth or IEEE 802.11b, interferences would not be a problem. Bluetooth like 802.11b is another type of wireless protocol that performs data transmission through the use of radio waves. Since Palm is currently involved with the Bluetooth SIG, they truly believe in the potential of Bluetooth. Palm has been heavily involved in the Bluetooth SIG for the past two years and was recently granted Associate Member status by the Bluetooth SIG. Palm’s involvement in various working groups, including I/O and synchronization shows their commitment to the new standard.

The ability to track FedEx packages wirelessly will also be implemented in future releases. Currently AvantGo offers a FedEx channel for package tracking status and drop-off location information. Currently the end user is required to enter the FedEx tracking number into the AvantGo FedEx channel. For the drop-off locator, the user has to enter his/her address and (or zip only) into the retrieval field. Once the user HotSyncs their palm on the cradle, AvantGo will connect to the FedEx database on the Internet and retrieve the necessary information about your tracking number. Instead of having the user HotSync their palm pilots on the cradle to retrieve
the tracking status and drop-off location, our application will wirelessly HotSync and connect to AvantGo to retrieve them.

With our knowledge of C programming along with the help of Palm documents and websites, we plan to create an application that would be vital to many e-commerce businesses and customers.

Background and Related Work

- **IrDA Web Page** - Infrared Data Association. IrDA is an International Organization that creates and promotes interoperable, low cost infrared data interconnection standards that support a walk-up, point-to-point user model. The standards support a broad range of appliances, computing and communications devices.

- **PalmOS Flat-File Database** - DB is a general purpose database program for the Palm Pilot. Application and source codes.

- **DDH Software** - HanDBase is a database application for the PalmOS handheld computers. With HanDBase on a PalmOS computer, users can extract data stored on a PC and sync it with their handheld computers to read, edit, or collect new information while away from the office. The standard version of HanDBase comes with two applications, one that resides on the Palm computer, and a Windows "desktop" application that resides on the users’ PCs. By offering both a palmtop and desktop application, HanDBase offers users a great deal of flexibility to create, edit, and view databases in the field on the Palm computer or desktop PC. Databases are transferred to and from the users’ PCs via Palm’s "HotSync" function and can be manipulated with the HanDBase desktop application.

- **List Web Page** - A simple database program with executables and source code.

- **JFile** - database program for the PalmOS Platform. Features include an advanced filtering screen, 3 to 5 times quicker in nearly all functions(!), categorization of databases, printing via PalmPrint, font choices, increased to 50 fields per record, and up to 4000 characters per field, scrollbars for fast navigation, and much more! Included with JFile Version 4.x is a PC based application that can convert Comma Separated Value (.csv) files to and from JFile .pdb format databases. This is a really nice program that implements the PalmOS database; unfortunately, the source is not available. We have to buy this program if we want the full version.

- **PalmOS Development Tools** - Helped us start our Palm research.

- **Bluetooth Wireless Technology** - Bluetooth wireless technology is a standard and specification for small-form factor, low-cost, short range radio links between mobile PCs, mobile phones and other portable devices.

- **Bluetooth Software Interest Group** - The Official Bluetooth Website.

- **Palm Programming: The Developer’s Guide** - Online guide for Palm Programming by O'Reilly.

- **XTNDAccess IrDA Printer Adapter** - Wireless infrared printer adapter for any printer. XTNDAccess IrDA Printer provides 'Walk-up-and-print’ convenience with no cable connections to make. Allows a printer to communicate with an infrared-equipped
portable computer. Supports IrDA infrared communication standard - transmitting and receiving data up to 115,200 baud, Connects to a standard parallel printer (Centronics) port - supporting pass-through parallel connection (does not interfere with normal network printing). Supports Win95, Win98, WinCE2.0 or greater, and Win NT4.0. Operates up to one meter (3 feet) away.

- **PACT Digital Precision Powder Scale** - A weight scale with an optional Infrared Data Port. This allows the scale to transmit weight data via a beam of invisible infrared light to the revolutionary PACT Digital Precision Powder Dispenser. The Scale can also print weights through the data port to a HP 82240 portable printer.
- **FedEx Web Site** - To locate the shipping rates for the 2Day and Express Saver service.
- **Stevens Creek Software** - Stevens Creek Software, LLC is a privately held company in Cupertino, California, which has been developing and publishing software for Macintosh, DOS, Windows, and the Palm Computing platform since 1988. Stevens Creek Software has been a leading developer of athletic software for Macintosh and Windows and more recently Palm since 1990, and was among the earliest and is now one of the leading publishers of business software for the Palm platform, with 17 shipping titles and numerous successfully completed custom development projects.
- **Tech Center Labs** - An IrDA prototype page that describes how to build an IrDA module.

The Metrowerks CodeWarrior tutorial and documentation was of great help. Without that tutorial, we would be buried in confusing source code from the other database applications. We first bought a Palm Programming book from Barnes & Noble so that we could learn how to code. It was really confusing because we never programming in the PalmOS before. Even though it is similar to C, C++, and Java, there are still some differences.

The "CodeWarrior PalmOS Tutorial for Windows" document was the most effective in designing the graphical user interface. The document was very effective in showing us the process of creating and testing a sample PalmOS application. It first starts out by illustrating how to create the main form for your application. Then it continues on with creating menus and also multiple forms. All we had to do was read up to chapter five of the documentation and we were able to develop our user interface. Using the Metrowerks CodeWarrior Constructor, developing the user interface became much easier. However, it took us awhile to figure out how to create more than two forms. When adding forms, buttons, fields, and bitmaps for the user interface, the Constructor will add corresponding parameters into the header file and the developers have to basically make use of the new parameters. This was not realized until we found out that none of our buttons would go to its corresponding form. We had to add the new parameters into our C file so that when the button is pressed, the form will appear.

In our 2Day and ExpSav forms, the user has to enter a weight and approximate distance as input. These two inputs will be handled by our PalmOS application module. This module with evaluate both input parameter and send them to our Database Operations module. The Database Operations module will evaluate both input parameters, perform a database search and provide the results in the shipping rate field. The Database Operations API modules will be called when the user presses the calculate button.
For the Infrared Module, we read the Beaming (Infrared Communication) section in the "PalmOS Companion." The PalmOS Exchange Manager provides a simple interface for PalmOS applications to send and received typed data from any number of remote devices and protocols. For additional help, we visited the IrDA web page. We also found a web site that has an application called "PalmPrint" available for download. The PalmPrint application prints via infrared, serial, or even parallel (with optional cable) to a wide variety of desktop, portable, and receipt printers. They also have some of the source code available to software developers. We also found another web site that was doing a research project on IrDA devices and a guy named David Bowerman from the site made his IrDA communication C source code available. After going through the documentation and the sample source codes from both websites, we are still unclear on how to implement our application to communicate through the infrared port.

**Project Specification**

The goal we hope to achieve from this senior research class is to be able to learn the different interfaces the PalmOS offers. Using the PalmOS, we are able to learn the functionality of the infrared port, PalmOS programming, & the interaction of other devices. Designing a FedEx shipping rate calculator will help us learn the different interfaces the palm pilot offers. In order to accomplish our goal, we have created FedExDB. FedExDB is a software package that will allow palm pilot users to calculate the shipping rate of a package by its weight and distance between the origin and destination. This program is useful for people who are in e-commerce. With our program, people won’t need to spend time standing in front of a scale to weigh their packages and figuring out how much shipping would be. They can also purchase a weight scale from PACT. PACT offers a weight scale with an option of an infrared port which can beam the weight of an object to a receiving infrared port, like a palm pilot. The program will also require an IrDA printer adapter so that it can utilize the shipping label printer. The user has the option to print shipping labels through the infrared port. Since not all printers provide with a built in infrared port, user can purchase the XTNDAccess IrDA printer adapter and be able to print shipping labels. The palm-side application will look like the user interfaces (see GUI & Description link) and has the option of getting the shipping rates for either a 2Day service or the Express Saver service.

The platforms we are assuming our program will function correctly would be for all models of the palm pilots. The FedEx palm database files (*.pdb) are about 38k. The FedExDB.prc file is approximately 5k and will continue to get bigger as more code gets added. So utilizing the either the 2MB or 8MB version would be suit the program well.

The program generated from our FedExDB.prc (see Source Code link for updates) file will create the FedExDB application. The FedEx icon is in the form of a FedEx truck. Upon tapping the FedExDB icon, the PalmOS application program will allow the user to choose to get the shipping rate for a 2Day service or an Express Saver service. Other choices will include the Label Printer and the Weight Scale as mentioned above. When either the 2Day or Exp Saver button is tapped, the PalmOS application will require the user to enter the weight of the package
and approximate distance between the origin and destination. The output will be the shipping rate for the specific weight that was entered. To make this function work, the program will look through a palm database file (*.pdb) and will generate the shipping rate.

Our Print Label module expects the user to input the destination address where their packages will be sent. The whole form should be filled out to ensure proper output on the printer. This module is going to use infrared as the means of data transmission. Some constraints we might encounter in our data transmission may include beaming wrong signals to the other infrared device. Depending on the size of our data transmission of the shipping label, some packets may drop and may have to retransmit the packet again. To retrieve the FedEx tracking status and drop-off locations, our application will utilize a wireless protocol to HotSync with AvantGo to retrieve the information. To make this happen, Bluetooth or IEEE 802.11b will be implemented to communicate to AvantGo.

Project Design

There will be five main modules that need to be created:

- **PalmOS application** - This application will obtain commands through the user interface. Depending upon the event the user chooses: 2Day & Express Saver buttons - This module of the application will take the user to the forms that handles the database search.
- **Print Label** - This module of the application will take the user to the Print Label form that allows to user to fill-in an empty shipping label and allows the user to beam through infrared to print the label.
- **Infrared module** - This module of the Print Label application will allow the user to communicate with a printer with infrared capabilities. For those that don't have an infrared enabled printer, there is a universal infrared printer adapter available. **XTNDAccess IrDA Printer Adapter.** The commands necessary to implement the enumeration for both the palm pilot and the infrared enabled printer are described in Table 2. Enumeration is the procedure in which a host and a peripheral recognize each other to enable communication between them. A peripheral must be enumerated with a host before it can exchange data with the palm side application layer. To initiate the enumeration, the function calls in Figure 2 would need to be called. The module would first need to call ComsOpen(), ComsEnableInfrared(), the TestPrint(), then ComsDisableInfrared(), and finally ComsClose() to end the session. These routines will allow the use of the standard serial library on the palm. Once the "Print Label" button is pressed, the function calls in Figure 1 will start the data transmission through the infrared port to the infrared enabled printer.

```c
static void TestPrint ( void )
{
    CharPtr dataP;
    VoidHand dataH;
```
DWord result;
Word err;
LocalID dbID;
int theInfo;
char theChars[ 32 ];

dataH = MemHandleNew( 256 );
if ( dataH != 0 )
{
    dataP = MemHandleLock ( dataH );
    StrCopy ( dataP, "This is the string to print" );
    dbID = DMFindDatabase ( 0,"PalmPring" )
    if ( dbID )
    {
        err = SysAppLaunch ( 0,dbID,0,cmdGetChars,(char*)&theInfo,&result );
        StrIToA ( theChars, ( long )theInfo );
        StrCat ( theChars," chars per line" );
        WinDrawChars ( theChars,StrLen(theChars),15,30 );
        err = SysAppLaunch ( 0,dbID,0,cmdGetLines,(char*)&theInfo,&result );
        StrIToA ( theChars,(long)theInfo );
        StrCat ( theChars," lines per page" );
        WinDrawChars ( theChars,StrLen(theChars),15,42 );
        StrCopy ( dataP,"This is the string to print" );
        err = SysAppLaunch(0,dbID,0,cmdPrintChars,dataP,&result);
    }
    MemPtrFree ( dataP );
}

Figure 1: Sample Code from PalmPrint for data transmission through infrared.

<table>
<thead>
<tr>
<th>Infrared Launch Code</th>
<th>Message</th>
<th>Pointer</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdPrintChars</td>
<td>32768</td>
<td>CharPtr</td>
<td>Print characters at CharPtr</td>
</tr>
<tr>
<td>cmdXmitChars</td>
<td>32769</td>
<td>CharPtr</td>
<td>Transmit characters at CharPtr</td>
</tr>
<tr>
<td>cmdStartPrint</td>
<td>32770</td>
<td>n/a</td>
<td>Start printing in line by line mode</td>
</tr>
<tr>
<td>cmdStartXmit</td>
<td>32771</td>
<td>n/a</td>
<td>Start transmission in line by line mode</td>
</tr>
<tr>
<td>cmdPrintLine</td>
<td>32772</td>
<td>CharPtr</td>
<td>Print a line (or set of lines) in line by line mode</td>
</tr>
<tr>
<td>cmdXmitLine</td>
<td>32773</td>
<td>CharPtr</td>
<td>Transmit a line (or set of lines) in line by line mode</td>
</tr>
<tr>
<td>cmdEndPrint</td>
<td>32774</td>
<td>n/a</td>
<td>Finish printing in line by line mode</td>
</tr>
<tr>
<td>Command</td>
<td>Code</td>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>--------</td>
<td>-----------</td>
<td>------------------------------------------------------------</td>
</tr>
<tr>
<td>cmdEndXmit</td>
<td>32775</td>
<td>n/a</td>
<td>Finish transmitting in line by line mode</td>
</tr>
<tr>
<td>cmdPrintLinePassThru</td>
<td>32776</td>
<td>CharPtr</td>
<td>Print a line of characters in passthru mode</td>
</tr>
<tr>
<td>cmdXmitLinesPassThru</td>
<td>32777</td>
<td>CharPtr</td>
<td>Transmit a line of characters in passthru mode</td>
</tr>
<tr>
<td>cmdGetChars</td>
<td>33000</td>
<td>int*</td>
<td>Return number of characters per line in <em>(int</em>)</td>
</tr>
<tr>
<td>cmdGetLines</td>
<td>33001</td>
<td>int*</td>
<td>Return number of lines per page in <em>(int</em>)</td>
</tr>
</tbody>
</table>

Table 1: Command calls to send data over to the infrared enabled printer.

Err ComsOpen(void)
{
    DWord err = 0;
    err = SysLibFind("Serial Library", &refNum);
    err = SerOpen(refNum, 0, 38400);
    if (err) return err;
    err = SerGetSettings(refNum, &serSettings);
    if (err) return err;
    serSettings.flags = serSettingsFlagBitsPerChar8 | serSettingsFlagStopBits1;
    err = SerSetSettings(refNum, &serSettings);
    if (err) return err;
    return err;
}

Err ComsClose(void)
{
    SerClose(refNum);
    return 0;
}

Err ComsEnableInfrared(void)
{
    *UART_MISC_REG = *UART_MISC_REG | UART_IRDA_ENABLE;
    *UART_MISC_REG = *UART_MISC_REG | UART_POLARITY_INVERTED;
    return 0;
}

Err ComsDisableInfrared(void)
{
    *UART_MISC_REG = *UART_MISC_REG & UART_IRDA_DISABLE;

*UART_MISC_REG = *UART_MISC_REG & UART_POLARITY_NORMAL;
    return 0;
}

**Figure 2:** Infrared API calls for opening up the infrared port for the infrared module.

<table>
<thead>
<tr>
<th>Command</th>
<th>Message Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>M328BASE</td>
<td>0xFFFFF000</td>
</tr>
<tr>
<td>M328_UARTMISC</td>
<td>(M328BASE + 0x908)</td>
</tr>
<tr>
<td>UART_IRDA_ENABLE</td>
<td>0x0020</td>
</tr>
<tr>
<td>UART_IRDA_DISABLE</td>
<td>0xFFDF</td>
</tr>
<tr>
<td>UART_POLARITY_INVERTED</td>
<td>0x000C</td>
</tr>
<tr>
<td>UART_POLARITY_NORMAL</td>
<td>0xFFF3</td>
</tr>
</tbody>
</table>

**Table 2:** Enumeration commands for the palm pilot and the infrared enabled printer.

- Database operations - This module is internal to the program and will not be influenced by the user directly. The database module will only interface with the 2Day and Express Saver buttons. The database module will support two basic API functions; the first will be called by the application GUI, while the second will be used during the HotSync function. The API that provides service to the application GUI is: dbSearchandRetrieve(). As suggested by the name, this operation will receive the package weight and travel distance values, as inputs, and it will search the relevant database, 2Day or Express Saver, for the corresponding output. The second API function is called dbUpdate(). This function will be used during a HotSync operation to update the database tables. The palm database files we were trying to access in the Table 3.

<table>
<thead>
<tr>
<th>Comma Separated Files (*.csv)</th>
<th>Palm Database Files (*.pdb)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2Day_Zone2</td>
<td>2Day_Zone2</td>
</tr>
<tr>
<td>2Day_Zone3</td>
<td>2Day_Zone3</td>
</tr>
<tr>
<td>2Day_Zone4</td>
<td>2Day_Zone4</td>
</tr>
<tr>
<td>2Day_Zone5</td>
<td>2Day_Zone5</td>
</tr>
<tr>
<td>ExpSav_Zone2</td>
<td>ExpSav_Zone2</td>
</tr>
<tr>
<td>ExpSav_Zone3</td>
<td>ExpSav_Zone3</td>
</tr>
<tr>
<td>ExpSav_Zone4</td>
<td>ExpSav_Zone4</td>
</tr>
<tr>
<td>ExpSav_Zone5</td>
<td>ExpSav_Zone5</td>
</tr>
</tbody>
</table>

**Table 3:** Comma separated files and palm database files for FedExDB.
• Wireless Module - This module is of our application allows the user to retrieve tracking status as well as drop-off locations through a HotSync session. This module will either use the IEEE 802.11b or Bluetooth wireless protocols to retrieve the tracking status and the drop-off locations.

Development Plan

Development of the project will primarily be done using the Metrowerks CodeWarrior development environment, using its default configurations. We will be using the C programming language to code this project. One of the software tools included in the Metrowerks CodeWarrior environment is the Constructor, which will be used to develop the application User Interface.

The Metrowerks CodeWarrior application can be obtained: http://www.palmos.com/dev/tech/tools/cw/

We plan on utilizing the PalmOS Emulator to perform most of our debugging and testing. The PalmOS Emulator can be obtained: http://oasis.palm.com/dev/kb/papers/1311.cfm.

To figure out how to create the program databases, which will hold the shipping rates, we will be using the MobileDB database program as a working model. This application provides a conversion mechanism, which accepts Excel-created comma-separated-values files and creates Palm database (.pdb) files. Another palm database application called JFile is another working example for us to observe. Unfortunately, the developers of both these applications did not make their source code available to the public. However, the developers of an application called PilotDB did make their source code available to the public.

To ensure accurate, and updated source files, we plan on using the CVS source control software. During the development cycle of this project, the majority of our work will be done cooperatively to ensure a thorough understanding of the functionality of each module.

Cross-development methodology will include: program emulation using the PalmOS Emulator, as well as native execution on a Palm IIIxe handheld device.

Testing methodology will include handcrafted generated test cases. The majority of our tests will emphasize strict validity on the types of inputs allowed by our program, specifically numerical values. Our test cases will cover the weight boundary in regards to the FedEx weight tables, and values that exceed this weight table, will not be covered by our application. The distance parameter will be tested only for those values which are valid in the state of California, i.e. values which exceed the length of California will not be handled by our application. Below is a sample of test cases which demonstrate the different areas of our program. Extensive testing will also be done regarding the Infrared module to ensure that it is functioning correctly. We also found a PalmOS application, called IRMonitor, that monitors the transmission of infrared
data through the palm. Below are test cases regarding our modules. We don’t expect all of these test cases to pass entirely because our application is in its alpha stages. However, once each module is complete, we will conduct a full test to ensure each test case passes.

**PalmOS Application Test Cases:**

- Entering an alphanumeric value into the weight or distance field.
- Entering a strictly numeric value into the weight field.
- Entering a value of "0" into the weight or distance field.
- Entering values, which exceed the boundary condition for either the weight or distance field.
- Entering blank values into the weight or distance field.
- Entering a numeric value and then an alphanumeric value in the fields.
- Able to accept the correct weight from weight scale.

**Infrared Module Test Cases:**

- Transmission of correct weight from weight scale.
- Transmission of correct data for printing shipping label.
- Output of shipping label format is correct.
- Ability to accept data transmissions through infrared from weight scale.
- Ability to handle transmission if there is interference.
- Data transmission to XTNDAccess IrDA Printer Adapter is correct.
- Data transmission to build-in infrared enabled printers is correct.
- Infrared port is enabled when initiated.
- Infrared port is closed after transmission.

**Print Label Module Test Cases:**

- Ability to input alphanumeric or numerical values into fields.
- Print Label button activates infrared port to transmit data to receiving infrared port on printer.
- Blank fields will generate an error message.

**Database Operations Module:**

- Entering an alphanumeric value will output error.
- Entering correct numerical value with output correct value for shipping rate.
- Ability to search and retrieve correct values according to weight tables.
Project Evaluation

What is the range of the infrared on the Palm Pilot?

- A PalmVx and a PalmIIIxe were both placed on the a flat surface. The PalmVx initiated a file transfer to the PalmIIIxe. The tested range of the infrared port tested to be approximately five to six feet.

Can one Palm Pilot broadcast a file transfer to more than one destination Palm Pilot?

- A PalmVx was used to broadcast a program to two other PalmIIIxe’s. They were first placed next to each other and then one on top of the other. Both were unable to receive the broadcast from the PalmVx. Only one of the PalmIIIxe’s were able to acknowledge the broadcast from the PalmVx.

Qualitative:

- **Specification:** We were able to complete the PalmOS application and the Database Operations modules. Both of these modules served as a basic building foundation of our application. We felt that the ability to calculate the shipping rates was most important. In our specification, were specified that we were going to search through a Palm database file to conduct the retrieval of the shipping rate. Instead, we had to implement our database as a multi-dimensional array. We had too many problems and spending too much time trying to interpret the PilotDB source code. As for the other modules, the Print Label and Infrared modules are partially complete. We have obtained sample source code and still have to figure out how to apply all the API calls to make our modules operate. We have also read articles on infrared transmission, it is still not enough information for us to implement. As for our Wireless module, we are unable to touch this module because Bluetooth is a new technology that has not been successfully proven to work. Palm has been heavily involved with the Blueooth SIG for the past two years and is still in development.

- **Design:** The design of FedExDB is expandable because our current design has many modules. It was developed using the Palm OS SDK. It’s expandable because we plan on using Palm database files. With these Palm database files, it can be easily updated as well as cover more area. Our modules communicated exceptionally well together. The PalmOS application and the Database Operations modules were seamlessly integrated. Unfortunately, our Infrared module were not able to fully integrate and test because we did not have an infrared enabled printer. We did not fully pursue this part of our application because we were lacking equipment. On the other hand, we did have source code for infrared communication.

- **Development:** Our development methodology worked well with cross compilation, emulation, and API emulation.

- **Protocol Handling:** Our project protocols were sufficiently robust for our specifications. Due to the obvious time constraints, we only implemented basic error checking features. The majority of our protocols, were implemented using the API’s
found in the PalmOS Reference documentation. This type of protocol implementation allowed our own protocols to be sufficiently robust in design.

**Quantitative:**

- **Performance:** The performance is good considering our databases for the 2Day and Express Saver are really big. There doesn't seem to be any lag when calculating the shipping rate. Our database is not very efficient because our database is in our actual program. It is in the form of a multi-dimensional matrix. If we were able to install palm database files (*.pdb) and have our program look in these files, it would be very efficient. However, our application because our code is very short and does the job. There isn't much extra memory copying because our application’s database files are fairly small due to the fact that we only covered the state of California. If we were to cover the domestic rates as well as international rates, our database would be much larger and may consume more memory.
- **Memory:** FedExDB is approximately 19Kbytes when installed onto the Palm Pilot. Since we don't have any hardware devices attached to our palm, our application consumes normal power usage.

**Concluding Remarks**

We have learned how to make a PalmOS application. We also learned how the program is event driven. We learned how to create forms and how to map buttons. Even though we were unable to utilize a true PalmOS database, we were able to mimic a database. Also, since we were unable to implement the infrared module, all the research and documentation we found taught us a great deal about infrared communication.

The implementation of the infrared module and the wireless module still remains. The next time we do this, we need to get all the necessary equipment before starting anything. We should also be experts at Palm Programming before deciding on creating applications that implement specific protocols. We would also like to get some help from people from palm on how to actually open up a palm database file and performing searches within the file. We would also get more information and help on infrared data transmission. Palm should also provide some Bluetooth SDK for developers who want to implement this wireless technology.

If our project was very successful, we could design a weight scale with infrared capabilities that can support more than just a couple of ounces. The infrared enabled weight scale that we found only measures weight that is only a couple of ounces. It doesn’t support pounds. Design an infrared enabled weight scale that can weigh packages that over one hundred pounds would be really useful. We could also develop a bar code scanner for the palm pilot and then interface it with the software so that it can scan the FedEx labels for keeping track of the tracking numbers. After scanning the FedEx labels, the application can store the tracking numbers into memory for further use later.
Some interesting research topics include home remote systems through X10, video conferencing, security through a thumb print scanner, make phone calls by interfacing the palm with some wireless protocol, and play movie clips.

Credits

- **The Programs** - Jonathan Tang & Joseph Wong (PalmOS Application, Print Label Module, Infrared Module, Database Operations); Eddy Tsu (PalmOS Application); Yu Chung Hong (PalmOS Application); Albert Chan (Database Operations)
- **Testing** - Jonathan Tang & Joseph Wong (Test Cases)
- **Source Codes Used** - David Bowerman’s infrared communications C Code; Stevens Creek PalmPrint SDK; PilotDB Source Code
- **Techniques Learned** - We would like to thank the group that was doing their research on their shopping program. Their group members consists of Leo Pan, Eric Pang, and Alex Yiu. Justin Chin also helped us by telling us which infrared web sites were useful. When they were presenting their project in front of the class, they were demonstrating how one of their search modules worked. The function prompts the user to enter an item number of a product and when you hit the search button, it scans the barcode and shows you some information.

Bibliography