

**The Purchase Order, Invoice and Payment Enquiry System:
PIPES**

By

Chris Huff

Submitted to
the Faculty of the Information Engineering Technology Program
in Partial Fulfillment of the Requirements for
the Degree of Bachelor of Science
in Information Engineering Technology

University of Cincinnati
College of Applied Science

June 2004

The Purchase Order, Invoice and Payment Enquiry System: PIPES

By

Chris Huff

Submitted to
the Faculty of the Information Engineering Technology Program
in Partial Fulfillment of the Requirements
for
the Degree of Bachelor of Science
in Information Engineering Technology

© Copyright 2004 Chris Huff

The author grants to the Information Engineering Technology Program permission to reproduce and distribute copies of this document in whole or in part.

Chris Huff

Date

Professor Russell McMahon, Faculty Advisor

Date

James F. Sullivan, Department Head

Date

Acknowledgements

I would like to give special thanks to Bill Kelley, former Director of Financial Systems at the University of Cincinnati, for allowing me the opportunity to work on this project. I would also like to give special thanks to Larry Caras, Director of Accounts Payable at the University of Cincinnati, for all the help that he provided to make this project possible. Without Larry's input, thorough knowledge of purchase order/vendor invoice processing, and confidence in my ability, PIPES would not have made it to implementation. Finally, I would like to thank Dennis Yockey, Associate Controller at the University of Cincinnati, for also supporting PIPES and authorizing the purchase of additional OnBase licenses.

Table of Contents

Section	Page
Acknowledgements	i
Table of Contents	ii
List of Figures	iv
Abstract	vi
1. Statement of the Problem	1
2. Description of the Proposed Solution	3
3. Description of the Actual Product and Intended Use	4
3.1 Technical Details and Resources	5
3.2 User Profile	6
3.2.1 Functional Users	6
3.2.2 Administrators	7
4. Design Protocols	7
4.1 Physical Relationship and Process/Workflow Diagrams	7
4.2 Interface Design	14
4.2.1 Front-End Interface	14
4.2.1.1 Home Page	15
4.2.1.2 Purchase Order, Invoice and Payment Information Page	16
4.2.1.3 Purchase Order Lookup Page	18
4.2.1.4 Standards and Guidelines for Web Interface	19
4.2.2 Automated Process for E-mail Notification	20
4.2.3 Back-End Interface for Invoice E-mail Administration	21
4.2.3.1 Contacts Screen	23
4.2.3.2 E-mail Content Screen	25
4.2.3.3 E-mail Verify Screen	27
4.2.3.4 Pending E-mail Screen	28
5. Timeline	32
6. Budget	33
7. Deliverables	34
8. Testing Plan and Procedures	35
8.1 E-mail Notification Feature	36
8.2 VB.NET Windows Application Interface	36
8.3 Web Interface	37
9. Proof of Design	38

10. Conclusions and Recommendations	43
Appendix A	46
Appendix B	47
Appendix C	50
Appendix D	51
References	53

List of Figures

Figure Number - Caption	Page
Figure 1. Diagram of Physical Layout	9
Figure 2. Diagrams of the Process/Workflow	10
Figure 3. Diagrams of the Process/Workflow	11
Figure 4. Diagrams of the Process/Workflow	12
Figure 5. Diagrams of the Process/Workflow	13
Figure 6. A Controller Web query	14
Figure 7. PIPES Home Page	15
Figure 8. Purchase Order, Invoice and Payment Information page	17
Figure 9. Purchase Order, Invoice and Payment Information page	18
Figure 10. Purchase Order Lookup page	19
Figure 11. Example of e-mail sent by PIPES E-mail Notification program	21
Figure 12. Start Menu screen	23
Figure 13. Contacts screen	25
Figure 14. E-mail Content screen	26
Figure 15. E-mail Verify screen	28
Figure 16. Pending E-mail screen	31
Figure 17. Timeline for PIPES project	32
Figure 18. Budget for PIPES project	33
Figure 19. Example of a TCP3270 screen and a CUFS table	46
Figure 20. Page 1 of Web interface's testing procedures	47
Figure 21. Page 2 of Web interface's testing procedures	48

Figure 22. Page 3 of Web interface's testing procedures	49
Figure 23. PIPES announcement	50
Figure 24. PIPES SQL Server Database Tables	51
Figure 25. PIPES SQL Server Data Sources & Queries	52

Abstract

The Purchase order, Invoice, and Payment Enquiry System (PIPES) makes a great deal of information associated with purchase orders readily available to the University of Cincinnati business community through a Website and an e-mail notification feature. PIPES Web interface pulls together, into one Web page, data that would take most users a lot of time and effort to find using online CUFS (College University Financial System) and printed reports. Pulling data from a SQL Server warehouse of CUFS information, the Web interface, built with ASP.NET and VB.NET technology, provides more historical and concise information than is available through online CUFS. It also enhances CUFS data by providing a link to the images of the invoice documents associated with a purchase order. Additionally, departmental users can conveniently and quickly look up purchase orders by account number.

Microsoft's Visual Studio.NET technology further extends PIPES' capabilities with a feature for sending vendor invoice information to departments via e-mail. Previously, the Accounts Payable Office manually copied and mailed out to departments approximately 700 invoices each week. The PIPES E-mail Notification program, a VB.NET Windows application, has eliminated this manual effort by e-mailing departmental contacts hyperlinks to the invoice images along with associated invoice and purchase order information. Another VB.NET Windows application, the PIPES Invoice E-mail Administration program, allows the Accounts Payable staff to easily manage this e-mail process. Both aspects of PIPES, the Website and e-mail notification feature, greatly reduce the time and effort that the University spends on managing purchase orders, vendor invoices and associated payment information.

The Purchase Order, Invoice and Payment Enquiry System: PIPES

1. Statement of the Problem

Purchasing at the University of Cincinnati is big business. In its 2002 fiscal year, the University of Cincinnati purchased approximately \$262.4 million from vendors using the College University Financial System (CUFS) purchasing module (19). This dollar amount represents over 34,000 purchase orders processed by business offices and personnel throughout the University (19). Currently, close to 500 users at the University have the ability to process purchase requisitions (18). An estimated 360 CUFS users track these purchase orders (along with the associated invoices and payments) for the accounts that they manage (20).

CUFS is a mainframe application that was implemented by the University in 1989 and serves as its core accounting system (4). To gain access to CUFS users have to go through two levels of security in addition to having special terminal emulation software called TCP3270 on their computer (4). (See Appendix A for an example of a TCP3270 screen.) For the past two years, I have handled calls on the CUFS Hotline, which is a help line that assists users with questions about or problems using CUFS. Ten to fifteen percent of the calls that I receive are questions concerning purchase orders, such as what invoices have been received and what payments have been made. In many cases, I have to refer the caller to the Accounts Payable Office to answer their questions because CUFS does not have a simple, consolidated interface that brings all of this information together in one place. Additionally, because some of the information only remains on CUFS for a certain period of time, it is not always accessible. The Accounts Payable Office has the

greatest expertise in answering these calls because they deal with this type of information everyday; it's a part of their function.

To gain a better understanding of the problem and an idea of what kind of application would be needed, I met with the Larry Caras, Director of Accounts Payable, several times. According to Mr. Caras, callers to their office are typically looking for the following information pertaining to purchase orders:

- An invoice from the vendor
- The date that the invoice was paid or will be paid
- The check number of the payment
- The address to which the check was mailed (1).

Users are often frustrated with finding information in CUFS because they have to be familiar with cryptic navigation commands as well as table names. A search for the information above could involve navigating to as many as six screens or tables (1). Even for an expert user, this can take a considerable amount of searching. (See Appendix A for an example of the CUFS table.) Mr. Caras estimates the following:

- 20% of users do not have any idea about how to look up information about purchase orders, invoices and payments in CUFS.
- 60% of users are fairly proficient in looking up some of this information but need help with the more complex searches.
- 20% of users are fairly expert and do not need any help in looking up this kind of information in CUFS (1).

CUFS is also not very versatile in that it does not allow users to easily pull information into an Excel spreadsheet or Access database. Printed reports from CUFS are sent out no less than monthly (22). Users have to wait for an accounting month to close before they

receive reports for the month that was closed (22). It then takes about a week for the reports to reach the departmental users with the current paper distribution process (22).

Finally, CUFS does not have the capability to allow users to view images of any documents. Retrieval of the invoice images through the Internet would be a great benefit to everyone. The Accounts Payable office receives directly from vendors approximately 700 invoices per week that then have to be copied and mailed out to departments (1).

After talking with Mr. Caras, I concluded that the University could greatly benefit from an enquiry system of purchase order, invoice and payment information that has the following qualities:

- **Accessible:** People at the University should be able to access this information through the local Intranet. The information should be presented in a manner that is consolidated and contains more transaction or record history than what is available on CUFS.
- **Intuitive and easy to use:** Users should be able to easily understand and navigate the interface.
- **Versatile:** The system should be capable of allowing users to download the information so that they can pull it into an Excel spreadsheet or Access database. Because of the University's End User License Agreement (EULA) with Microsoft, users have Microsoft Office products on their desktop computers.
- **Enhancement:** The system should be capable of allowing users to retrieve and download/print an image of an invoice.

2. Description of the Proposed Solution

In November 2002, I proposed to develop an enquiry application for the University that simplifies the retrieval of information pertaining to purchase orders. In addition to providing invoice and payment data associated with purchase orders, this application, called *The Purchase order, Invoice, and Payment Enquiry System (PIPES)*,

would also allow users to download information to their desktops and retrieve invoice images.

To make purchase order information more accessible and easy-to-use, PIPES would have the following features or qualities:

- It would be Web-based and incorporated into the Controller's Web site. Users would be able to access to this information more readily and with fewer layers of security and software.
- The purchase order, invoice and payment information would consolidate and make available more historical data
- PIPES would be easy-to-use and intuitive.

To provide versatility and enhancement, PIPES would also have the following features:

- Capability to retrieve an image of an invoice
- Capability to send users an e-mail notification that contains a link to an invoice image.

3. Description of the Actual Product and Intended Use

PIPES is a system of Web and Windows applications that simplify the retrieval of information pertaining to purchases orders and associated vendor invoices and payments. Through a front-end interface, it provides users with purchase order, invoice and payment data in a consolidated manner. Additionally, it allows users to access invoice images for viewing or printing. Simplifying access to this information expedites the resolution of payment inquiries and tracking of invoice and payment activity against a purchase order. Through a back-end interface, PIPES sends e-mails to designated users, notifying them that the Accounts Payable Office has received and processed a referenced invoice for payment. The e-mail also contains a link to the referenced invoice. By eliminating the

need to send copies of invoices in the mail to departments, this feature saves the Accounts Payable Office a great deal of time and the cost of paper.

3.1 Technical Details and Resources

PIPES has both a database and application layer. The database layer has three components. One component makes use of data in a data warehouse to provide textual data and to store and identify invoices for the e-mail notification process. The other component makes use of a catalogue database to pull the invoice images from the imaging system. The third component makes use of data entered by a central office (such as Accounts Payable) that identifies the recipients for the e-mail notifications.

The application layer has three components or interfaces. The first component is a Web interface that links the user to the purchase order, invoice, and payment data (both textual and image). The second component is a VB.NET Windows Application interface that gives the central office (Accounts Payable) access to the e-mail notification data, which includes updating information about the designated recipients. The third component is also a VB.NET Windows Application interface that sends the e-mail notifications.

PIPES was developed using Microsoft Visual Studio.NET. The Web interface was designed using ADO.NET, VB.NET and ASP.NET. The Windows applications were designed using ADO.NET and VB.NET. The database component that serves this system comes from SQL Server 7.0 databases, which extract data from an existing data warehouse (7). This data warehouse resides on a SQL Server in the Financial Systems department (7). VB.NET and ASP.NET were chosen because of the University's EULA with Microsoft and because of their easy integration with SQL Server.

The Web pages are incorporated into the Controller's Web site, which now resides on the UC Web Server. The invoice images are provided by the OnBase imaging system (7). This system, which is administered by Financial Systems, was installed by Assured Micro-Services in October 2002 (7). The components of this system that are most important to this project are the following:

- The SQL database that catalogues the images of the accounting documents
- Internet Information Server that hosts a Website and Web processing for OnBase
- The SNAP server where the images are actually stored
- The program, called OnBase from Hyland Software, which controls the imaging system features (3).

During the installation of the imaging system, I spoke to Rich Diers, Digital Products Manager with Assured Micro-Services, about the capability to view documents over the Internet. Mr. Diers guided me in using the "DocPop URL Creator", a feature in OnBase that allows the invoice images to be accessed over the Internet.

3.2 User Profile

PIPES has two groups of users: functional users and administrators.

3.2.1 Functional Users

Functional users are Accounts Payable and departmental employees throughout the University. These employees are the primary users of PIPES' Web interface. The Accounts Payable employees respond to inquiries from departments and vendors about invoices and payments associated with purchase orders. The departmental employees typically work in the business offices of departments or handle business transactions for the department.

Functional users can range from clerical/secretarial staff to managers. They interact with computers on a daily basis and have, at minimum, an intermediate level of skill using the Internet. These users also have experience with the following:

- PC with Windows Operating System
- Internet Explorer 5.5+, or Netscape 7.0
- Microsoft Office (Excel) 97, 2000, XP, or 2003
- CUFS.

3.2.2 Administrators

The Administrators use the Windows Application interface to manage the e-mail notification data. They are responsible for managing the e-mail notifications on a daily basis and maintaining the information about who receives the e-mail notifications.

This group consists of central office staff members (most likely from Accounts Payable). These employees are intermediate to advanced level computer users, who use Windows applications such as Access to obtain CUFS data from the data warehouse in Financial Systems. These users have experience with the following:

- PC with Windows Operating System
- Internet Explorer 5.5+
- Microsoft Office (Access) 2003
- CUFS.

4. Design Protocols

4.1 Physical Relationships and Process/Workflow Diagrams

The diagram in Figure 1 shows the physical relationships of the hardware for the processing that takes place in and around PIPES. The diagrams in Figures 2 through 5 provide more detail to these relationships and depict the processing order and workflow that precede and include PIPES.

More specifically, the diagrams in Figures 2 and 3 depict the processing and workflow that occur with CUFS, the imaging system and the e-mail notification feature. Although the processing in these two diagrams occurs outside the PIPES interface, it shows the workflow that supports and enhances it. Please note that the time sequence (“Day 1”, “Day 2”, etc.) is in relative not actual terms.

The diagram in Figure 4 depicts the processing that directly involves PIPES. The time reference in this diagram means the processing can only occur on or after “Day 3” relative to the time frame depicted in the Figures 2 and 3.

The diagram in Figure 5 depicts both daily processing and processing that occurs when an update is necessary. The daily processing involves the data that is brought into the SQL Server data warehouse from the CUFS mainframe system. This data supports the PIPES interface. The processing that occurs on an “As Needed” basis involves the administration of PIPES’ e-mail notification feature.

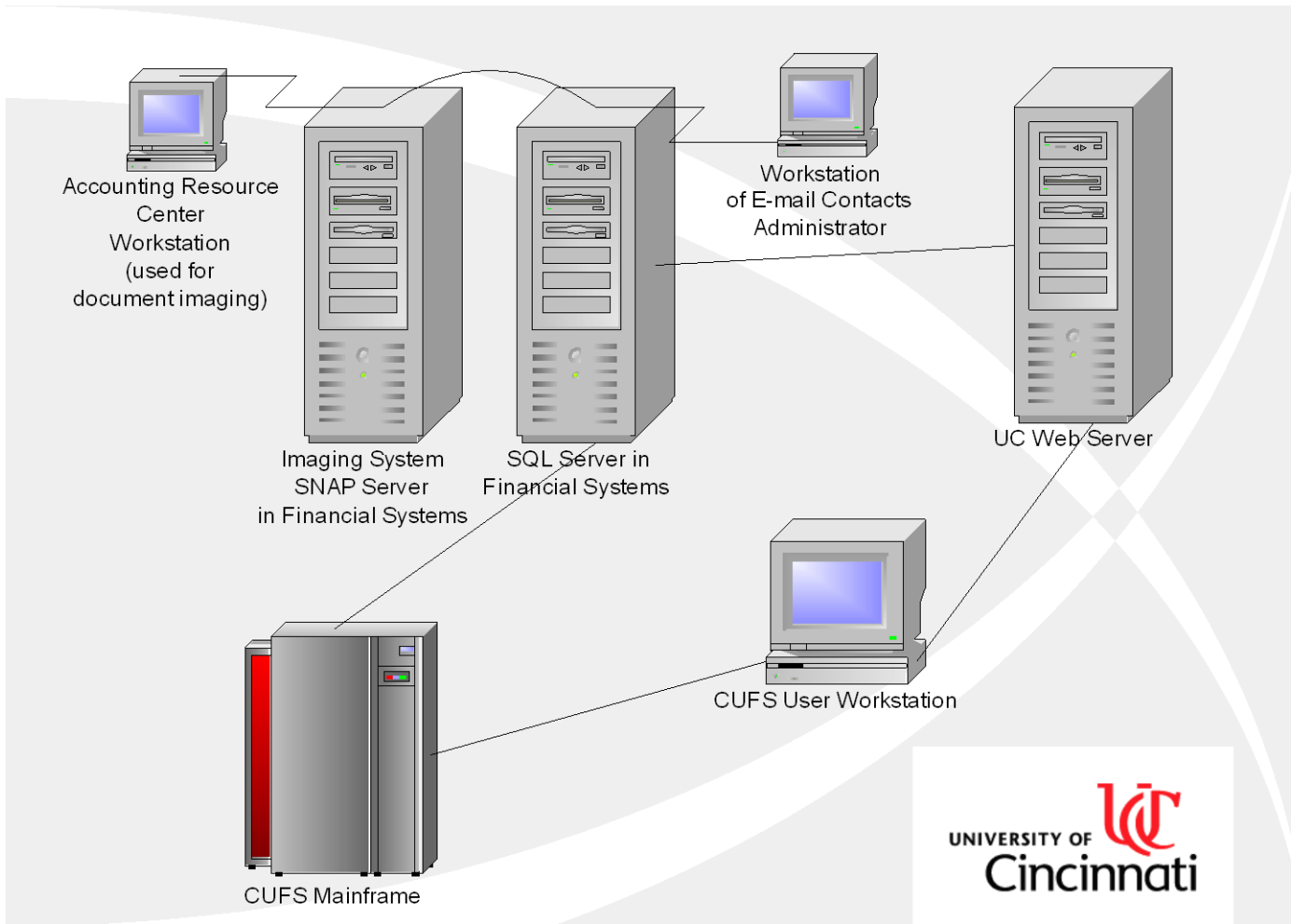


Figure1. Diagram of the Physical Layout

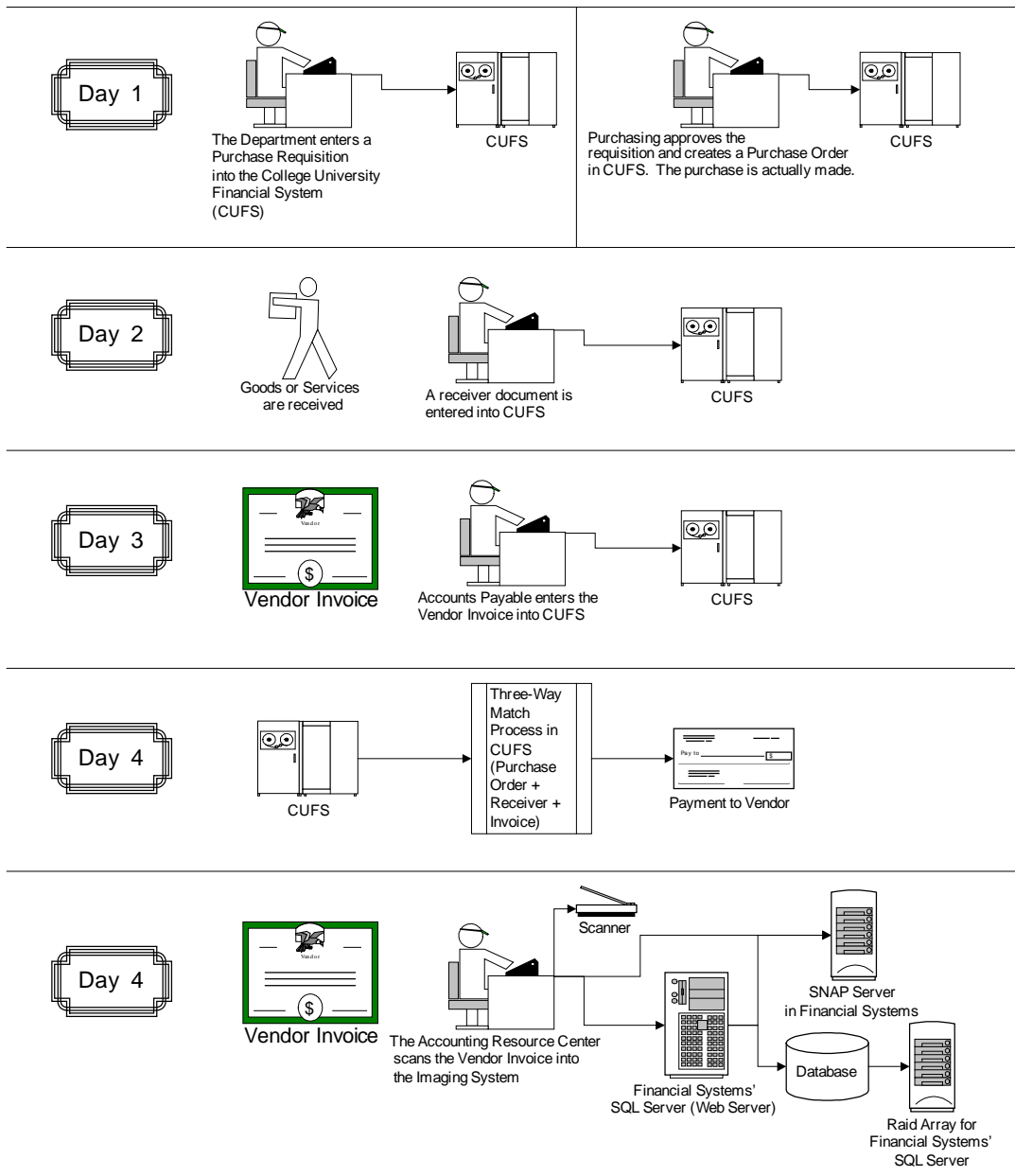


Figure 2. Diagrams of the Process/Workflow

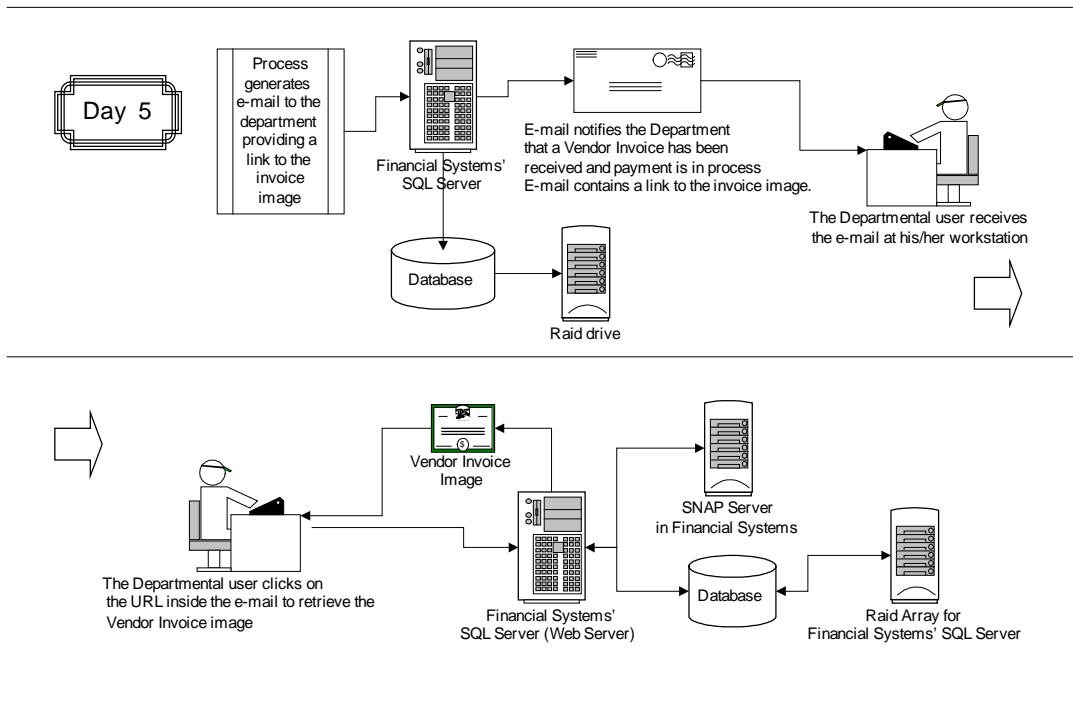


Figure 3. Diagrams of Process/Workflow

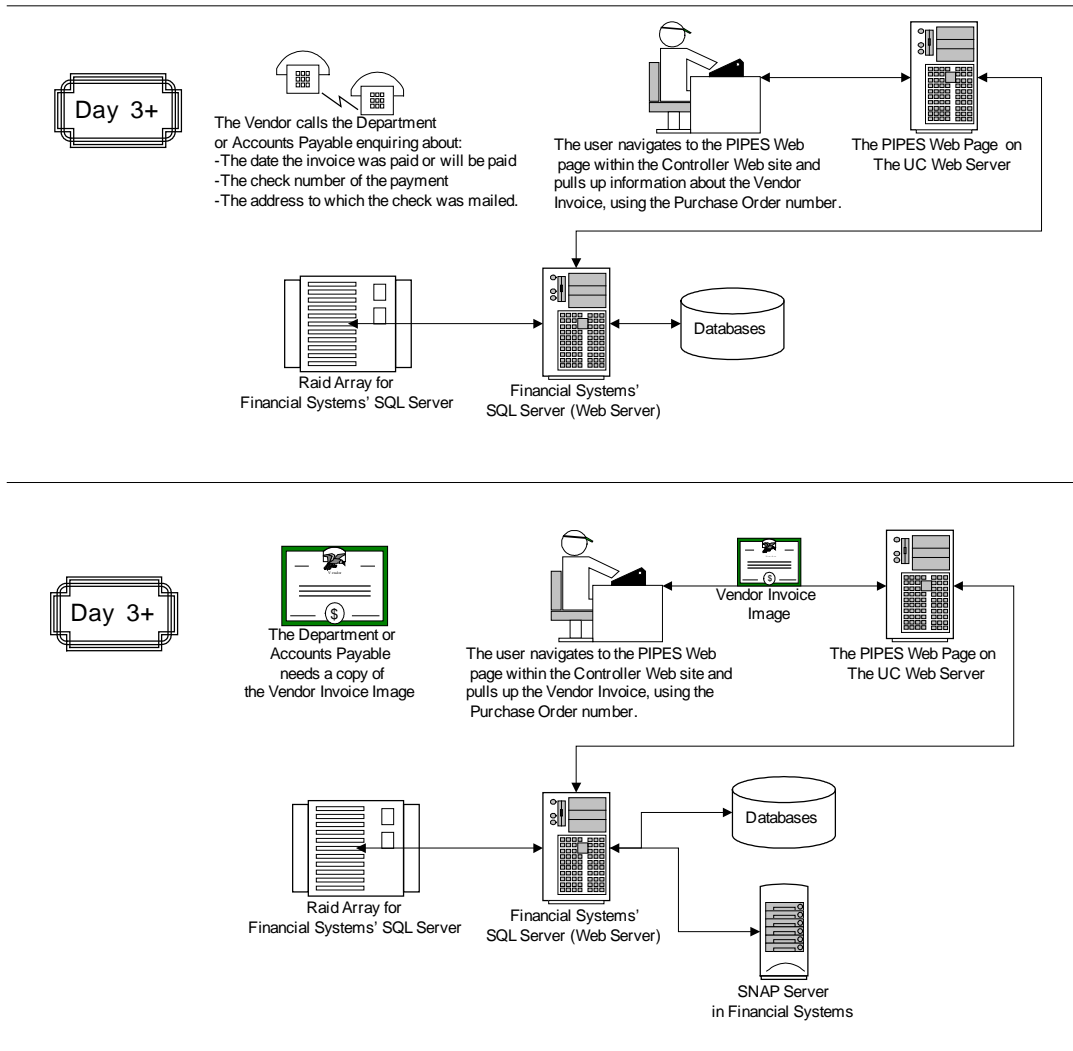


Figure 4. Diagrams of the Process/Workflow

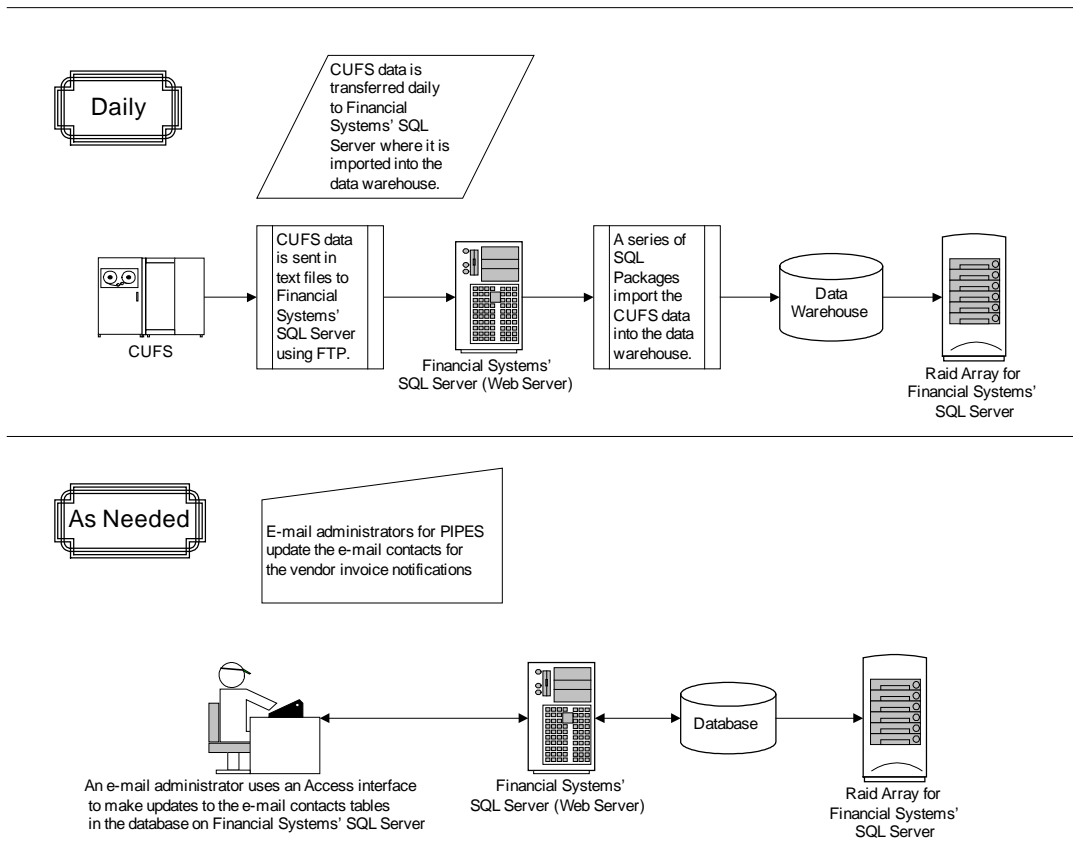


Figure 5. Diagrams of Process/Workflow

4.2 Interface Design

4.2.1 Front-End Interface

The front-end of PIPES has three pages: a *Home* page, a *Purchase Order (PO) Number Lookup* page, and a *Purchase Order, Invoice and Payment Information* page. Users access the Home page of PIPES' front-end interface through a link on the Controller's "Query Page". This page currently lists seven other queries in addition to PIPES. Each query has a title and a description. To access one of the Controller Web queries, the user clicks the query's underlined title on the "Query Page". The link for PIPES is consistent with this format and navigational design. PIPES' Web pages are also consistent with the format, color scheme and design of the existing Controller Web queries. Figure 6 shows the top portion of one of the Controller Web queries.

The screenshot shows a Microsoft Internet Explorer window titled "Departmental Accounting Transactions for a Month - Microsoft Internet Explorer". The browser's menu bar includes File, Edit, View, Favorites, Tools, and Help. The page content features a navigation menu with the following items: Controller, Directory, Documentation, News, Financial Data Queries, and UC Home Page. Below this menu is a grid of links: Account Expenses, Departmental Expenses, Departmental Transactions, Document Details, Endowment Query Page, Fund Balance, Microfilm Index, and PIPES. The main heading reads "Departmental Accounting Transactions by Organization for a Month" next to the University of Cincinnati logo. The form includes three numbered steps: 1. "Select the Fiscal Year:" with a dropdown menu showing "2004"; 2. "Select the Fiscal Month:" with a dropdown menu showing "01"; 3. "Enter the Organization Code:" with an empty text input field. A "Get data" button is positioned below the input field. A section titled "About this Query" contains the text: "This query selects all financial transactions (expense, revenue, encumbrance and balance sheet) for the selected organization number. Budget transactions are not displayed." At the bottom, a note states: "1 Fiscal Year is the first part of the search value that must be entered. The list of allowable Fiscal Year codes is shown."

Figure 6. A Controller Web query

The query shown in Figure 6 allows a department to retrieve the revenue and expenditure transactions for a selected Fiscal Year and Organization Code.

4.2.1.1 Home Page

The first page of PIPES front-end interface contains the following:

- A title area that includes the University of Cincinnati branding logo
- A descriptive section that provides an explanation of PIPES
- A functional section that has both a textbox, where the user inputs the purchase order number (the key link to the information), and a command button that the user clicks with the mouse cursor to get to the associated information
- Another functional section that has a descriptive label and navigation button, linking to a second page that allows the user to look up purchase order numbers by account.

Figure 7 shows the top portion of PIPES Home Page.

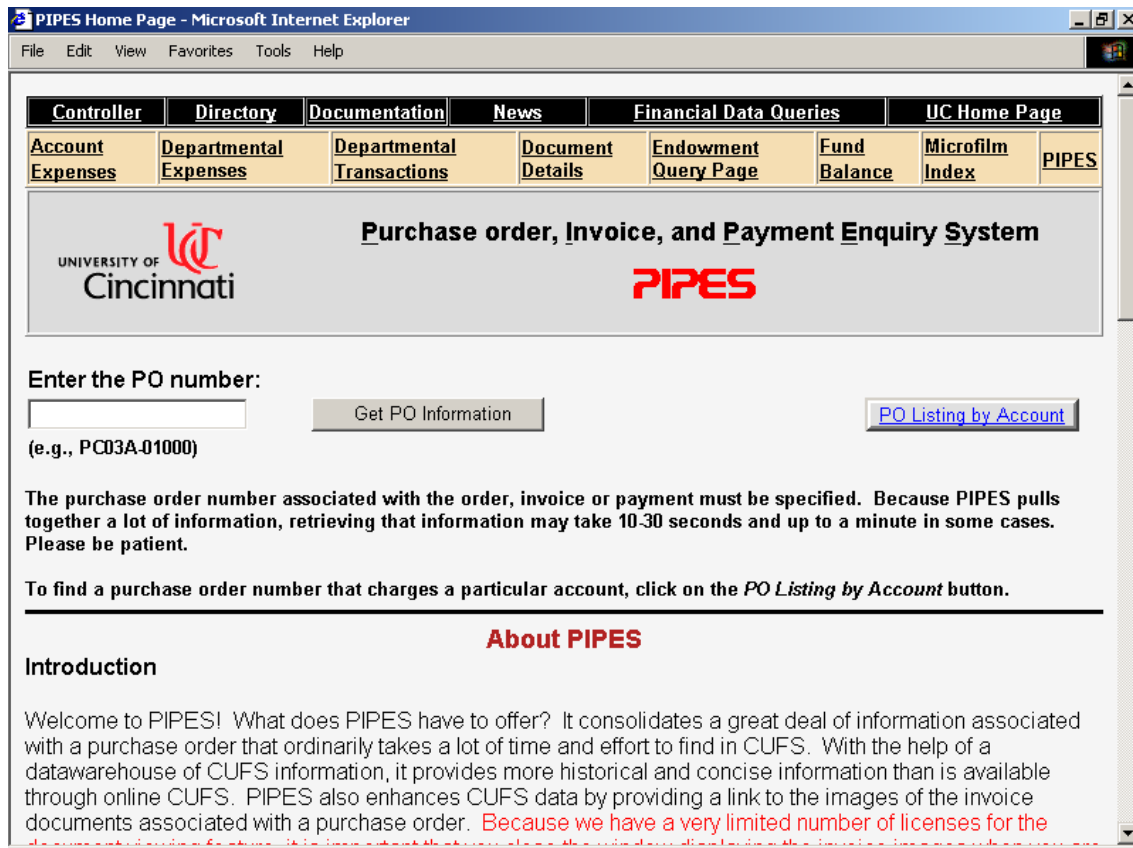


Figure 7. PIPES Home Page

4.2.1.2 Purchase Order, Invoice and Payment Information Page


The *Purchase Order, Invoice and Payment Information* page has the following sections and features:

- A title area that includes the University of Cincinnati branding logo
- A detail section with the following six elements:
 - 1) PO summary information: PO Number, Vendor Name, Vendor Code, hyperlink to view PTEX information, Date of PO, Ordered Amount, Received Amount, Invoiced Amount, Closed Date, Closed Total, Open Balance
 - 2) Account line information: Line Number, Account Number, Function, Object, Sub-object, Organization Name, Line Amount, Expended Amount, Closed Amount, Balance
 - 3) Commodity line information: Line Number, Description, Unit Measure, Unit Cost, Order Quantity, Total Cost, Received Quantity, Last Unit Cost, Invoiced Quantity, Total Invoiced
 - 4) Change order modifications: Modification Number, Date, Purchase Order Amount, Change Order Amount
 - 5) Invoices: Invoice Date, Invoice Number with hyperlink to the invoice image, Archive Reference, Total Invoice Amount, Discount Type, Discount, Number of Days (payment terms), Estimated Paydate, Actual Paydate
 - 6) Charges/payments: Transaction I.D. (check numbers contain hyperlink to view check information), Record Date, Amount, Account Number, Function, Object, Sub-Object, Reference Number, Invoice Number with hyperlink to the invoice image
- A hyperlink on the Invoice Number that, when clicked, opens up another browser session and retrieves the invoice image in that second browser window
- A command button that returns the user to the *Home* page.

Figures 8 and 9 show screen captures of the *Purchase Order, Invoice and Payment Information* page.

PIPES Purchase Order, Invoice and Payment Information - Microsoft Internet Explorer


File Edit View Favorites Tools Help



UNIVERSITY OF
Cincinnati

[PIPES Home Page](#)

[PO Listing by Account](#)



PIPES
Purchase Order, Invoice
and Payment Information

Summary (OPPH):

PO Number	Vendor Name	Vendor Code	
PC03A-01000	INTECOM	0428924721	View PTEX

PO Date	Ordered Amt	Received Amt	Invoiced Amt	Closed Date	Closed Total	Open Balance
10/9/2002	\$107,792.00	\$92,701.12	\$107,792.00	2/25/2003	\$107,792.00	\$0.00

Account Lines (OPPL):

Line	Acct No	Func	Objt	Sobj	Orgn Name	Line Amt	Expended Amt	Closed Amt	Balance
01	A3990503770	5732	T300	03	TELCO FIELD OPRTN & INFRSTR	\$107,792.00	\$107,792.00	\$107,792.00	\$0.00

Commodity Lines (OPPC):

Line	Description	Unit	Unit Cost	Order Qty	Total Cost	Rec Qty	Last Unit Cost	Inv Qty	Total Invoiced
001	TELEPHONE EQUIPMENT PER THE ATTACHED QUOTE DATED 10/2/02	LOT	\$107,792.00	1.00	\$107,792.00	0.86	\$110,370.36	1.02	\$107,792.00

Change Order Modifications (MPPH):

Mod No	Date	PO Amt	CO Amt
00	10/9/2002	\$107,792.00	\$0.00

Figure 8. Purchase Order, Invoice and Payment Information page

PIPES Purchase Order, Invoice and Payment Information - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Line	Description	Unit	Unit Cost	Order Qty	Total Cost	Rel Qty	Last Unit Cost	Inv Qty	Total Invoiced
001	TELEPHONE EQUIPMENT PER THE ATTACHED QUOTE DATED 10/2/02	LOT	\$107,792.00	1.00	\$107,792.00	0.86	\$110,370.36	1.02	\$107,792.00

Change Order Modifications (MPPH):

Mod No	Date	PO Amt	CO Amt
00	10/9/2002	\$107,792.00	\$0.00

Invoices (OVH):

Invoice Date	Invoice Number & Image	Archive Reference	Total Invoice Amt	Disc Type	Discount	Num Days	Estimated Paydate	Actual Paydate
12/16/2002	20026377	EP00311456	\$78,596.15	09	0	30	1/15/2003	1/15/2003
1/2/2003	20026753	EP00313661	\$13,744.00	09	0	30	2/1/2003	1/31/2003
2/25/2003	200310834	EP00322078	\$15,451.85	09	0	30	3/27/2003	3/27/2003

Payment Transactions (General Ledger):

Transaction ID	Record Date	Amount	Account	Func	Objt	So	Reference Number	Invoice Number & Image
PV03010800003	12/16/2002	\$78,596.15	A399-050-3770	5732	T300	03	PC03A-01000	20026377
AD00000448230	1/15/2003	\$78,596.15	A399- -				03010800003	20026377
PV03010800004	1/2/2003	\$13,744.00	A399-050-3770	5732	T300	03	PC03A-01000	20026753
AD00000450906	1/31/2003	\$13,744.00	A399- -				03010800004	20026753
PV03022600190	2/25/2003	\$15,451.85	A399-050-3770	5732	T300	03	PC03A-01000	200310834
AD00000460884	3/27/2003	\$15,451.85	A399- -				03022600190	200310834

[Return to top](#)

Figure 9. Purchase Order, Invoice and Payment Information page

4.2.1.3 Purchase Order Lookup Page

The *Purchase Order Lookup* page contains the following:

- A title area that includes the University of Cincinnati branding logo
- A descriptive section that explains the selection criteria for looking up a purchase order number by account
- A functional section that has the following elements:
 - 1) Three textboxes of required input, where the user enters, respectively, the Fund, Area and Organization code of the account number used in the purchase order
 - 2) For a more refined search, two textboxes of optional input, where the user can enter the Function and/or Object of the account number
 - 3) A command button that the user clicks with the mouse cursor to get to the associated information

- A command button that returns the user to the *Home* page
- A detail section for the results of the lookup that indicates: the PO Number, the PO's Create Date, the Function, the Object, the Line Amount, the Line Balance, the Vendor Name, and the Closed Date.

Figure 10 shows a screen capture of the *Purchase Order Lookup* page.

PO Number	PO Date	Fund	Area	Orgn	Func	Objt	Subj	Line Amount	Balance	Vendor Name	Close Date	Responsible Person
PC02A-02193	4/12/2002	A001	060	3670	5710	3002	15	\$11,187.20	\$0.00	BGI TEMPORARY SERVICES	8/5/2002	LARRY CARAS
PB000229654	4/4/2002	A001	060	3670	5710	3002	15	\$4,992.00	\$0.00	KELLY SERVICES INC.	7/11/2002	LARRY CARAS
PC04A-00041	7/1/2003	A001	060	3670	5710	3002	25	\$228.00	\$0.00	MILLENNIUM BUSINESS SYSTEMS	10/17/2003	LARRY CARAS
PC04A-00041	7/1/2003	A001	060	3670	5710	3002	25	\$620.40	\$378.69	MILLENNIUM BUSINESS SYSTEMS		LARRY CARAS
PC04A-00042	7/1/2003	A001	060	3670	5710	3002	25	\$228.00	\$0.00	MILLENNIUM BUSINESS SYSTEMS	10/17/2003	LARRY CARAS
PC03A-00048	7/1/2002	A001	060	3670	5710	3002	25	\$217.00	\$0.00	MILLENNIUM BUSINESS SYSTEMS	8/9/2002	LARRY CARAS
PC03A-00051	7/1/2002	A001	060	3670	5710	3002	25	\$639.41	\$0.00	MILLENNIUM BUSINESS SYSTEMS	7/30/2003	LARRY CARAS
PC03A-00054	7/1/2002	A001	060	3670	5710	3002	25	\$217.00	\$0.00	MILLENNIUM BUSINESS SYSTEMS	8/9/2002	LARRY CARAS
PC02A-00075	7/1/2001	A001	060	3670	5710	3002	25	\$422.00	\$0.00	MILLENNIUM BUSINESS SYSTEMS	6/14/2002	L. CARAS
PVROBOT01	3/20/2002	A001	060	3670	5710	3008	04	\$10.00	\$0.00	ROBOT TEST 1	3/21/2002	
PVROBOT02	3/20/2002	A001	060	3670	5710	3008	04	\$5.00	\$0.00	ROBOT TEST 2	3/21/2002	
PVROBOT03	3/20/2002	A001	060	3670	5710	3008	04	\$15.00	\$0.00	ROBOT TEST 3	3/21/2002	
PVROBOT04	3/20/2002	A001	060	3670	5710	3008	04	\$20.00	\$0.00	ROBOT TEST 4	3/21/2002	

Figure 10. Purchase Order Lookup page

4.2.1.4 Standards and Guidelines for Web Interface

Each of the pages meets the standards established by the University of Cincinnati and described in its *Web Communication and Design Standards* (21) and *Branding Standards Manual* (15). The list below highlights some of the standards and guidelines that were met by PIPES.

- Designed Web pages for 800 x 600 resolution (21, page 5)

- Made use of the University’s primary color palette with applied secondary palette as appropriate (21, page 5)
- Kept page designs as short and simple as possible (21, page 5)
- Used high contrast combinations of text vs. background color for best readability. Black text on a white background is preferred for body copy (21, page 5)
- Used Sans serif--Arial font, which is one of the recommended font sets (21, page 5)
- Created and maintained a consistent look and feel on all pages and sub-sites (21, page 5)
- Used navigational aids such as “Next/Previous Page,” “Return to Top,” “Table of Contents,” and jump links to help user work through the site (21, page 5).

4.2.2 Automated Process for E-mail Notification

PIPES has an automated process or program that sends out e-mails to departmental contacts, listing invoices that Accounts Payable has recently received and processed for each contact’s department. Each invoice in the e-mail contains a hyperlink to the document images of the invoice. Additionally, the following information is provided for each invoice: the Invoice Number (with the hyperlink), Invoice Date, Reference Number, Purchase Order Number, Requestor, Vendor Code, and Vendor. Figure 11 shows an example of an e-mail sent by PIPES E-mail Notification program.

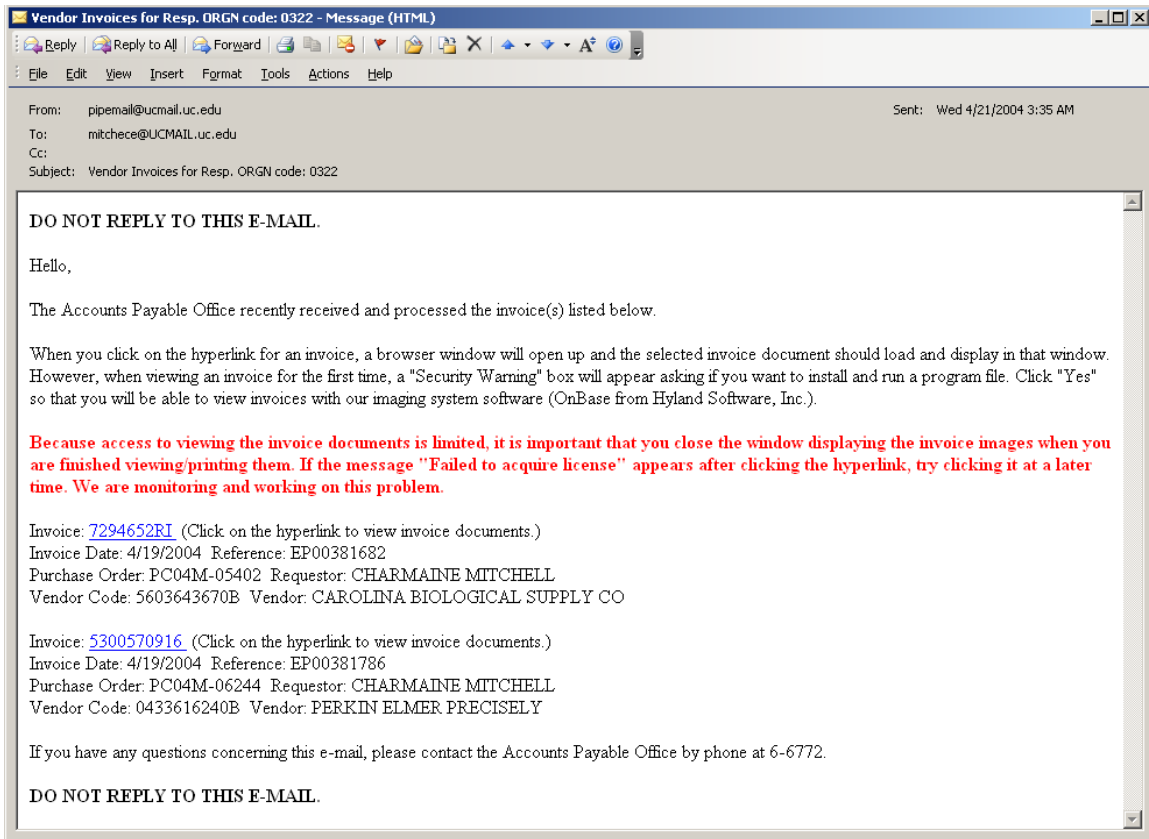


Figure 11. Example of e-mail sent by PIPES E-mail Notification program

4.2.3 Back-End Interface for Invoice E-mail Administration

PIPES has a back-end interface used by Accounts Payable staff for the administration of the vendor invoice notifications and the e-mail contacts. The PIPES Invoice E-mail Administration application has an Outlook-style interface with the following screen elements: a menu bar, a tree view control for navigation, and a panel control for displaying a *Start Menu* and four different functional screens. The four different functional screens are *Contacts*, *E-mail Content*, *E-mail Verify*, and *Pending E-mail*. These screens will be described in the sections that follow. The *Start Menu* is the initial screen that shows up in the panel control; it lists and provides a brief description of the four different functional screens as well as providing a “hyperlink” to each screen. The menu bar has 3 menus: *Get*, *Help*, and *Exit*. In the event that the user’s mouse fails,

a short-cut key can be activated with the *Alt* key on the keyboard to use keystrokes that access these menus; the shortcut keys in combination with the arrow keys allow the user to navigate without the mouse. The *Get* menu provides an alternative way for navigating to the *Start Menu* and four different functional screens. The *Help* menu provides information about PIPES Invoice E-mail Administration and instructions on using each of the functional screens. *Exit* allows the user to exit the application via the menu bar. Figure 12 shows the PIPES Invoice E-mail Administration interface with the menu bar at the top, the tree view control on the left, and the panel control (displaying the *Start Menu*) on the right. The gray bar, a splitter control, between the tree view control and panel control can be moved to the right or left with the left mouse button by the user to increase/decrease the display area for either control; increasing the display area for one control decreases the display area for the other.

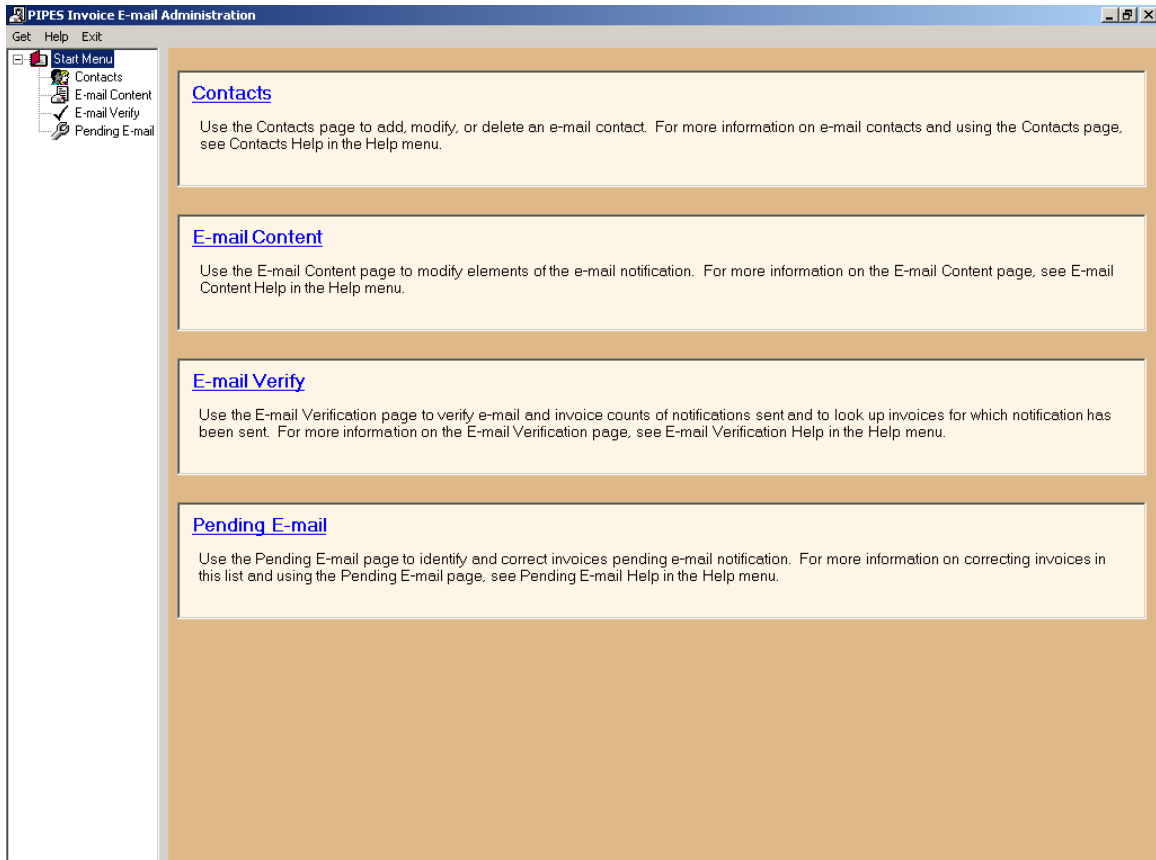


Figure 12. Start Menu screen

4.2.3.1 Contacts Screen

The *Contacts* screen allows certain users in the Accounts Payable Office to add, modify or delete a contact. The contacts list is used by the automated e-mail notification feature to associate, by responsible department number (RespOrg), an invoice with a contact and e-mail address. This screen contains the following:

- A label or title for each screen element
- A functional section with the following features and elements for searching, modifying or deleting a contact record:
 - A datagrid with re-sizeable columns that lists the contact information for all contacts and allows the user to modify the information within the datagrid

- A combo-box, where the user can select or type the Responsible Organization number of the contact and then move to the associated record in the datagrid
- A combo-box, where the user can select the primary contact's last name and then move to the associated record in the datagrid
- A combo-box, where the user can select the name of the responsible department and then move to the associated record in the datagrid
- A button that the user can click to export the data in the datagrid to an Excel spreadsheet
- A button that the user can click to delete the currently selected Responsible Organization number and contact information
- A button that the user can click to save modifications made to contact records in the datagrid
- Another functional section with the following features and elements for adding a contact record:
 - Text boxes where the user can input the contact information
 - A button that the user can click to add contact information for a new Responsible Organization.

Figure 13 shows the *Contacts* screen.

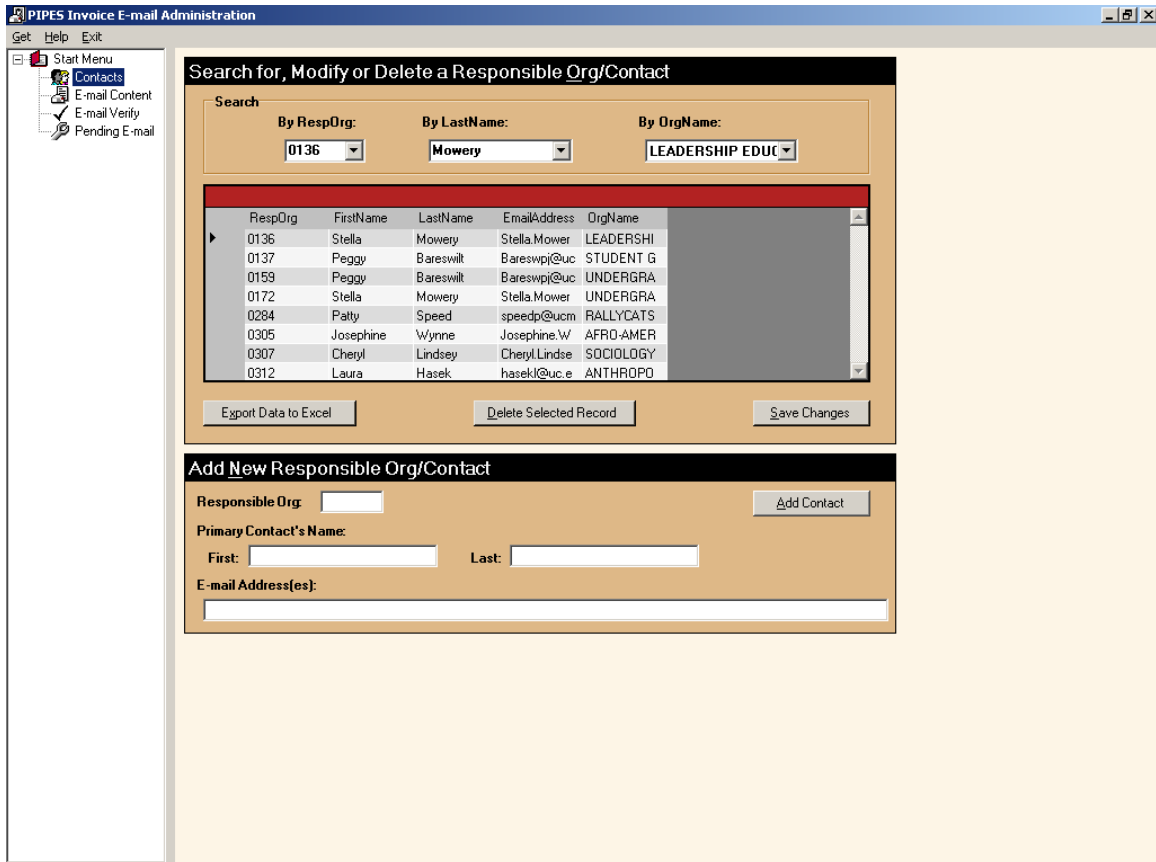


Figure 13. Contacts screen

4.2.3.2 E-mail Content Screen

The *E-mail Content* screen allows the E-mail Contacts administrators to modify several elements of the notification e-mail. The screen has the following features and elements:

- A label or title for each screen element
- A functional area for the e-mail header with the following modifiable elements:
 - A text box for the e-mail address of the sender(s) – *From*
 - A text box for the e-mail address of carbon copy recipient(s) – *Cc*
 - A text box for the e-mail address of blind carbon copy recipient(s) – *Bcc*
 - A text box for information about the e-mail subject – *Subject*

- A functional area for the e-mail body with the following modifiable elements:
 - An combo-box for choosing the level of importance of the e-mail – *Importance*
 - A text box for the introductory greeting and information that precedes the actual invoice information in the e-mail - *Introduction/Greeting*
 - A text box for the closing remarks or comments at the end of the e-mail – *Closing*
- A button that the user can click to save modifications.

Since the message format of the notification e-mail is HTML, HTML tags must be used to create special formatting in the *Subject*, *Introduction/Greeting* and *Closing* text boxes.

Figure 14 shows the *E-mail Content* screen.

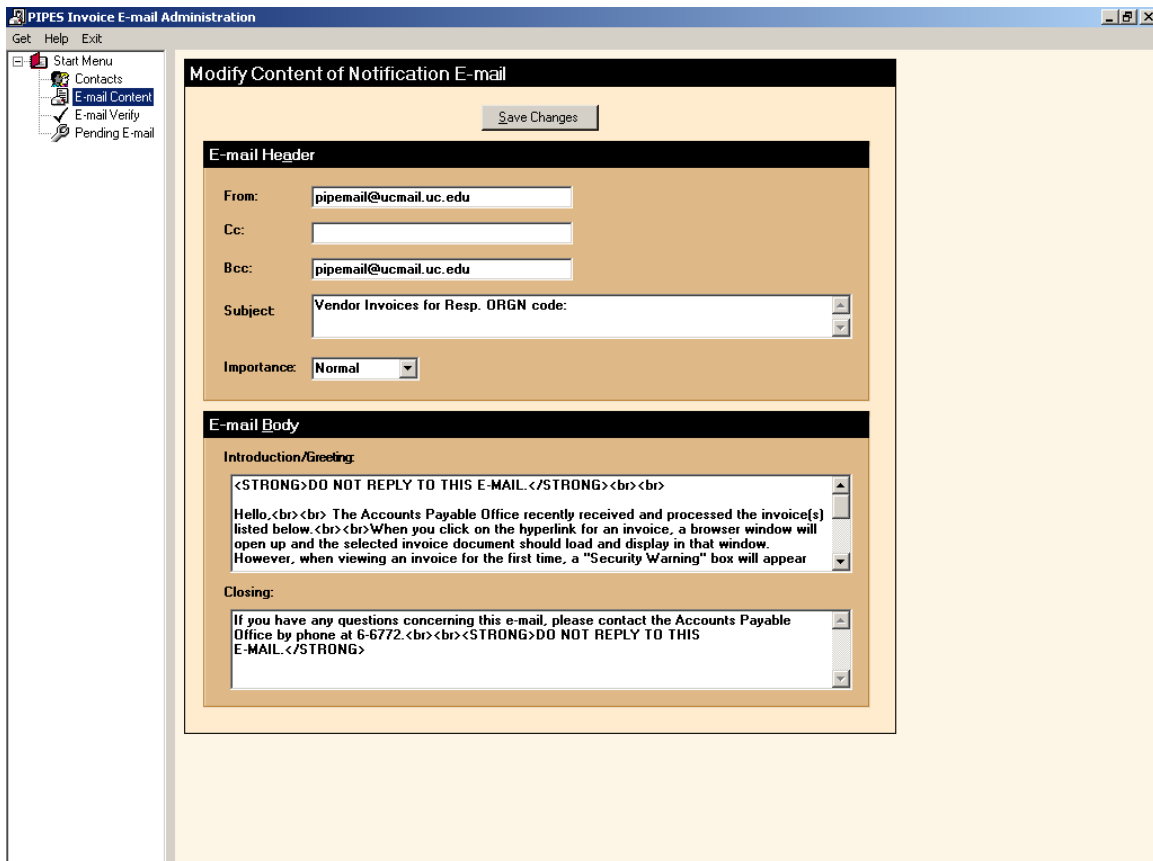


Figure 14. E-mail Content screen

4.2.3.3 E-mail Verify Screen

The *E-mail Verify* screen provides information to the administrators about the e-mail notifications that have been sent. It contains the follow features and elements:

- A label or title for each screen element
- A section of count information with the following elements:
 - A datagrid with re-sizeable columns that lists by date and time a count of invoices and e-mails before they are sent out and a count of invoices and e-mails after they are sent. The count before the e-mail notification program runs is based on a query of the OVIH_PLUS table. The count after the e-mails are sent out is created by the E-mail Notification process.
 - A button that the user can click to export the data in the datagrid to an Excel spreadsheet
 - A button to refresh the datagrid after the user runs the E-mail Notification program via the *Pending E-mail* screen
- A section of invoice information (the Responsible Organization, Invoice Number, Invoice Date, Vendor Name, Vendor Code, Imaging Date, Date Sent, and PO Number) with the following elements:
 - A radio button, a combo-box and text box that allows the user to specify a calendar month and year for retrieving a list of invoices sent in that time period
 - A radio button and text box that allows the user to specify a Responsible Organization code for retrieving a list of invoices sent for that code
 - A radio button and text box that allows the user to specify a purchase order number for retrieving a list of invoices sent for that purchase order
 - A button that the user clicks, after specifying search criteria, to retrieve the invoice information
 - A datagrid with re-sizeable columns that lists the invoice information
 - A button that the user can click to export the data in the datagrid to an Excel spreadsheet.

Figure 15 shows the *E-mail Verify* screen.

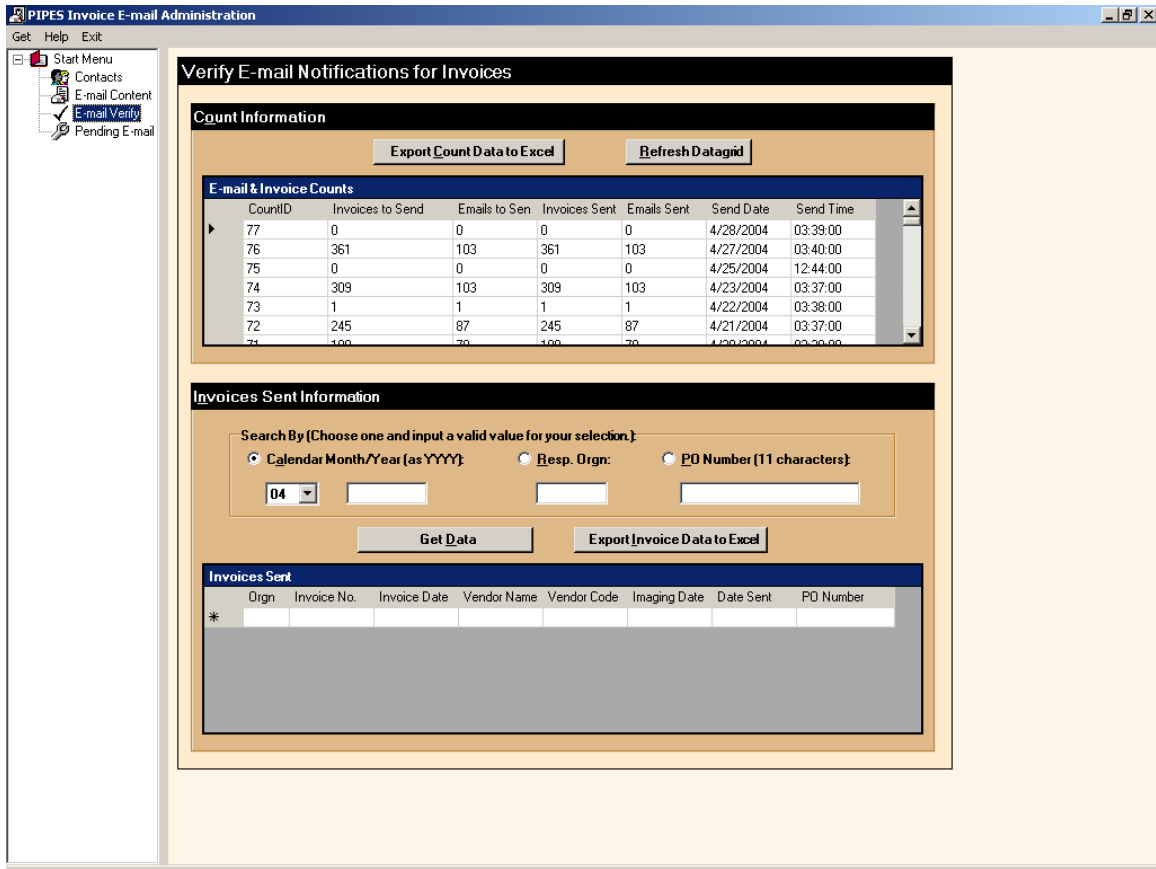


Figure 15. E-mail Verify screen

4.2.3.4 Pending E-mail Screen

The *Pending E-mail* screen helps the administrators in Accounts Payable resolve errors with the e-mail notifications. It mainly helps them resolve or identify three types of problem situations. 1) One situation occurs when a notification should be sent to a department for one or more invoices, but a record with contact information does not exist in the Contacts table for that department. The E-mail Notification program therefore cannot send an e-mail for these invoices. These invoices are marked as pending because they exist in the system and have document images associated with them. 2) A second situation occurs when an incorrect value is entered for an invoice's EP reference number during invoice processing. Because of this incorrect reference, the invoice does not get

associated with a document image. The E-mail Notification program does not include an invoice in an e-mail if it does not have a document image. Invoices in this situation are also marked as pending. 3) A third situation occurs when a record in the Contacts table has an invalid e-mail address. In this third situation, an e-mail is sent but a message is returned to the PIPES e-mail account, indicating that the e-mail was undeliverable to the invalid e-mail address. The invoices in this situation are marked as sent.

The *Pending E-mail* screen thus has the following features and elements for dealing with these situations:

- A label or title for each screen element
- A functional section, *Invoices Pending E-mail Notification on OVIH_Plus*, with the following features and elements for identifying invoices that are pending:
 - A datagrid that lists invoices pending e-mail notification. This list includes any invoices that fall into the first situation. Since many invoices are processed each day before they are scanned into the imaging system, invoices not yet associated with document images and more than 14 days old will also appear in the datagrid; invoices that fall into the second situation would therefore be identified in this group.
 - The ability to modify the InvCopySent indicator and DateInvCopySent field in the datagrid.
 - A button that the user can click to save modifications made to the InvCopySent indicator and DateInvCopySent fields in the datagrid
 - A button that the user can click to export the data in the datagrid to an Excel spreadsheet
 - A button that the user can click to send notifications after a contact record has been added
 - A button to refresh the datagrid after the user runs the E-mail Notification program by clicking the *Send Notifications* button

- Another functional section, *Resend Invoices for Specified ORG and Date*, with the following features and elements for resending invoices after correcting the e-mail address in the Contacts table:
 - A text box where the user can input the Responsible Organization code pertaining to the undeliverable e-mail message
 - A text box where the user can input the date pertaining to when the undelivered e-mail was sent
 - A button that the user can click to retrieve the invoices associated with the undelivered e-mail
 - A datagrid with re-sizeable columns that lists the invoices retrieved from the search
 - A button that the user can click to resend a notification for the invoices listed in the datagrid.

Figure 16 shows the *Pending E-mail* screen.

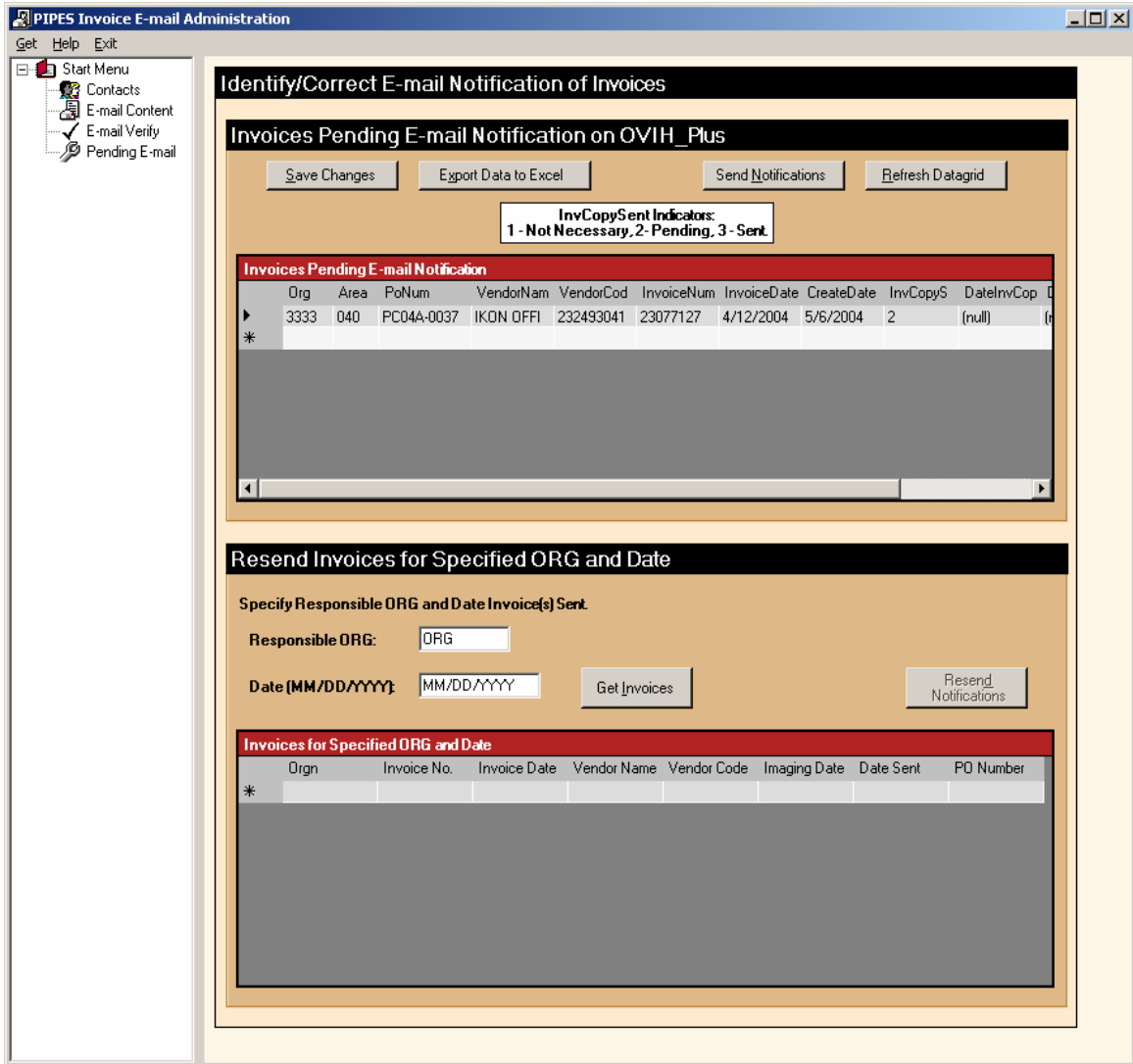


Figure 16. Pending E-mail screen

5. Timeline

Task	Start Date	End Date
Researched the Project	10/1/2002	11/18/2002
Worked on the Proposal Document	11/7/2002	12/12/2002
Submitted and Presented the Proposal	12/12/2002	12/12/2002
Developed the Database Components	3/31/2003	4/13/2003
Developed the Web Interface	4/14/2003	4/27/2003
Finalized the Web interface into a Working Quick Prototype	4/28/2003	5/4/2003
Tested the Working Quick Prototype and Resolved any Problems	5/5/2003	5/11/2003
Worked on the Design Freeze Document	5/12/2003	5/22/2003
Submitted a Draft of the Design Freeze Document	5/22/2003	5/22/2003
Prepared the Presentation of the Working Quick Prototype	5/23/2003	6/4/2003
Submitted the Design Freeze Document and Demonstrated the Working Quick Prototype	6/6/2003	6/6/2003
Developed the Automation for E-mail Notification	6/9/2003	7/31/2003
Migrated the Web Interface to the UC Web Server	7/7/2003	7/13/2003
Tested the Working Quick Prototype of the Web Interface with a Group of Users	8/5/2003	8/22/2003
Completed the Working Quick Prototype of the Web Interface for the Final Product	8/22/2003	8/28/2003
Developed the Interface for E-mail Contacts Administration	8/1/2003	2/15/2004
Finalized the Automation for E-mail Notification into a Working Quick Prototype	3/1/2004	3/15/2004
Finalized the Interface for E-mail Contacts Administration into a Working Quick Prototype	3/1/2004	3/15/2004
Tested the Working Quick Prototypes of the E-mail Contacts Administration and E-mail Notification Interfaces	3/15/2004	3/22/2004
Revised the E-mail Contacts Administration and E-mail Notification Interfaces to Resolve any Problems Discovered through Testing	3/17/2004	3/26/2004
Completed the Working Quick Prototypes of the E-mail Contacts Administration and E-mail Notification Interfaces for the Final Product. PIPES fully implemented.	3/26/2004	3/31/2004
Worked on the Final Document	4/1/2004	5/6/2004
Submitted the Final Document Draft	5/6/2004	5/6/2004
Prepared the Presentation of the Final Product	5/7/2004	5/31/2004
Submitted the Final Document and Demonstrated the Final Product	6/3/2004	6/3/2004

Figure 17. Timeline for PIPES project

6. Budget

Costs of Items Directly Associated with the Project:		
Visual Studio.Net Professional 2002	\$	549.00 (13)
Hosting of Web Pages *		<u>600.00 (14)</u>
Total Cost of Items Directly Associated with the Project	<u>\$</u>	<u>1,149.00</u>

* The cost for Web Page Hosting is for four years. It is estimated that this web application will be replaced in approximately four years by features of a new financial system (7).

Infrastructure Costs of New Items that Contribute to the Project:		
OnBase Imaging System	\$	29,180.00 (16)
Maintenance Agreement for OnBase Imaging System		15,000.00 (16)
SNAP Server		<u>4,200.00 (16)</u>
Total Infrastructure Cost of New Items that Contribute to the Project	<u>\$</u>	<u>48,380.00</u>

Figure 18. Budget for PIPES project

7. Deliverables

1. The Purchase order, Invoice and Payment Enquiry System simplifies the retrieval of information pertaining to purchase orders.
2. PIPES has a Web interface that does the following:
 - a) Brings together CUFS data associated with a purchase order number from the MPPH/MPPL, OPPH, OPPC, OPPL, and OVIH tables as well as the Current and Closed Ledgers
 - b) Allows the user to select an invoice transaction and then click a button or hyperlink that brings up an image of the selected invoice
 - c) Allows the user to look up purchase order numbers by account (e.g., FUND-AREA-ORGN).
3. The Web interface has the following hardware and software specifications:
 - a) It is developed using ADO.NET, ASP.NET, and VB.NET technology.
 - b) It is hosted on the UC Web Server.
 - c) It connects to the SQL Server, running version 7.0, in Financial Systems for its data content.
 - d) It connects to the Internet Information Server, running on Financial Systems' SQL Server, for the image retrieval. The images are actually stored on a SNAP Server in Financial Systems.
4. PIPES has an automated feature, a VB.NET Windows interface program, that sends an e-mail to designated users, notifying them that the Accounts Payable Office has received and processed a referenced invoice. Additionally, the e-mail contains a link to the document images of the referenced invoice. When the user clicks the link, the document images are pulled into a browser where the user can view and print the documents. The image retrieval is handled by a system outside of PIPES.
5. PIPES has a VB.NET Windows interface for the management of the e-mail notifications and the administration of the e-mail contacts identified for the notification. This interface allows certain Accounts Payable employees to do the following:
 - a) Add a new contact
 - b) Edit an existing contact

- c) Delete a contact
- d) Search for a contact
- e) Modify elements (To, From, Cc, Bcc, Subject, Introduction/Greeting, Closing) of the e-mail notification content
- f) Verify counts of e-mails and invoices that were sent
- g) Look up invoices that were sent for a particular calendar month, Responsible Organization, or purchase order
- h) Manage problems with invoices that are pending e-mail notifications because a contact record does not exist or because an invalid EP reference number was entered into CUFS
- i) Resend e-mail notifications for invoices where the e-mail was returned undeliverable due to an invalid e-mail address.

8. Testing Plan and Procedures

Testing was conducted by participants from three different areas. Those areas are the Accounts Payable Office, users from various departmental business units, and Financial Systems. Testing focused on three components: the VB.NET Windows interface for the automated feature, the VB.NET Windows interface for the e-mail contacts administration and management of the notifications, and the Web interface. A set of tests was defined by Financial Systems for each area. The areas testing each component were the following:

- VB.NET Windows Application Interface: Accounts Payable Office, Financial Systems
- E-mail Notification Feature: Accounts Payable Office, Departmental Business Unit, Financial Systems
- Web Interface: Accounts Payable Office, Departmental Business Unit, Financial Systems.

8.1 E-mail Notification Feature

1. Did you receive an e-mail notifying you of an invoice?
2. Does the body of the e-mail contain a hyperlink?
3. Does the hyperlink in the body of the e-mail open an Internet browser window and display images of the referenced invoice documents?
 - Click on the hyperlink.
4. When the e-mail notification feature is scheduled to run, did any participants receive an e-mail? **(Question for Financial Systems only)**
5. The invoices referenced in the e-mails sent, were they the ones that you expected? **(Question for Financial Systems only)**
 - Identify what invoices should be included in the e-mail notification run through SQL queries.

8.2 VB.NET Windows Application Interface

Below are some of the tests that were performed by the Applications Analyst in Financial Systems and the Assistant Director of Accounts Payable. Each screen had its own set of tests.

1. When you open the interface for E-mail Contacts Administration, does the administration form display properly?
 - Click on the icon that opens this interface.
2. Do contacts appear in the datagrid?
 - Navigate to the *Contacts* screens.
3. When you click the combo-box, does a drop-down list of Responsible Organization codes and department names appear?
 - Click on the down-arrow of the combo-box and select a Responsible Organization code.
4. When you select a Responsible Organization code, are you able to bring up the contact information?
 - After selecting a Responsible Organization code, click the down arrow on your keyboard.
5. Are you able to modify the contact information?
 - After retrieving the contact information for an existing contact, change the values in the datagrid row of the contact and click *Save*.
6. Are you able to add a new contact?

- Click the “Add New Contact” button and complete the textboxes of contact information. Confirm that the contact was added by clicking on the combo-box and retrieving the information for the contact that was just added.
7. Are you able to delete a contact?
 - Select a contact in the datagrid and then click on the “Delete” button.
 8. When you clicked the “Delete” button, did a message appear indicating how many records were deleted?

8.3 Web Interface

Please see Appendix B for a sample copy of the Web interface’s testing procedures form that was completed by one of departmental users.

1. Does the link to the *Home* page of the Web interface appear on the Controller’s “Query Page”?
 - Open up Internet Explorer and navigate to the Controller’s “Query Page”.
2. Does the link connect to the *Home* page and is the *Home* page displayed properly?
 - Click on the link.
3. Does the PO number textbox accept input?
 - Type a valid PO number into the textbox.
4. After clicking the *Get PO Information* button, does the *Purchase Order, Invoice and Payment Information* page appear with correct data?
 - Click the *Get PO Information* button and compare the information on the page with the information on CUFS.
5. Does a message, indicating that the PO was not found, appear if you search for an invalid PO number?
 - Type an invalid PO number into the textbox on the *Home* page, and click the *Get PO Information* button.
6. Does the navigation button that goes to the *Purchase Order Lookup* page from the *Home* page work?
 - Click on the navigation button that takes you to the page.
7. On the *Purchase Order, Invoice and Payment Information* page, do hyperlinks appear for the invoices listed in the payment section?
 - Identify a PO that has invoice activity, enter the number in the *Home* page textbox and click the *Get PO Information* button. Look for invoices in the payment section and confirm that a hyperlink appears.
8. Does the hyperlink work?

- Click the hyperlink. Another browser window should open and the invoice images should appear. Test the functionality of the controls for viewing the images.
9. Do the navigation buttons that return the user to the *Home* page work?
- Click on the navigation button from both the *Purchase Order Lookup* page and the *Purchase Order, Invoice and Payment Information* page.

9. Proof of Design

Design, development, and implementation of PIPES occurred in two phases. The first phase involved the design, development and implementation of the Web interface. This phase took approximately five months and included the redesign of the Controller Web queries, using ASP.NET. In the second phase, I designed, developed, and implemented the e-mail notification feature. Parts of this phase were begun during the first phase; however the bulk of it happened after the first phase was completed. The second phase took around eight months.

During the first phase, the Web interface of the Purchase order, Invoice and Payment Enquiry System was made available last August to a test group of 13 departmental users. For a period of two weeks, the test group was asked to evaluate the Web interface and respond to a set of questions. Please see Appendix B for a sample copy of the Web interface's testing procedures form that was completed by one of test group participants. Everyone reacted positively to the interface. I made several modifications to improve its design and content based on suggestions and comments made by some of the group members. Although the Web interface was actually put into a production environment as early as July 2003, University-wide access did not happen until January 2004; the Controller's Office did not want it made available to the entire University business community until after the purchase of additional licenses of the

OnBase Imaging System. It took approximately five months to purchase those additional licenses. Since being made widely available, PIPES Web interface has been very well received throughout the University. In February 2004, Sally Roetker, Business Services Officer in Parking Services, wrote the following about PIPES to Larry Caras, Director of Accounts Payable: “Thank you so much for the wonderful PIPES website. I love it! It works very well and makes CUFS so much easier to use” (12).

During the second phase, while the additional licenses for OnBase imaging system were being purchased, I did most of the development and realization work for the e-mail notification feature. I first started on the automated process that sends the e-mails out because I felt that conquering this aspect of the feature was critical to its success. Fortunately, I found some VB.NET code on Microsoft’s MSDN Website to help me develop this process very early on (10).

Determining exactly what data to use and how to use it was the next thing that I tackled after figuring out the automate e-mail process. I worked closely with Larry Caras, the Director of Accounts Payable, on this part of the project. One of the things that we had to decide was what piece of data in CUFS could be used to determine where the notification for a vendor invoice should be sent. Since the vendor invoices, ones requiring e-mail notification, are associated with a purchase order on the OPPH (Open Purchase Order Header) table in CUFS, we finally determined that the Responsible Organization (RespOrg) code on this table was the best piece of data to use. Even in the OPPH table, that code is also associated with a Responsible Person. We could then use the RespOrg code as the primary key upon which to build the table of contacts. I therefore created a SQL table that consists simply of the following four fields: RespOrg,

FirstName, LastName, and EmailAddress; this table is called Pipes_EmailContacts.

While the FirstName and LastName fields are intended for a primary departmental contact's name, the EmailAddress field can hold more than one e-mail address. Larry Caras decided for management purposes, however, that a RespOrg could have no more than two e-mail addresses. He did not want to create the extra work of maintaining/updating e-mail addresses for more than two recipients.

Another challenge associated with the data concerned the vendor invoice data itself. For the Web interface, I built the OVIH_PLUS table. This table originally was based only on the OVIH (Open Vendor Invoice Header) table in CUFS with an added field to indicate the date when a document was stored in the OnBase imaging system. I quickly learned that the OVIH_PLUS had to be re-engineered to serve the needs of the e-mail notification feature. One of the problems with the OVIH table in CUFS is that it is purged on a monthly basis; over time, it therefore allows for the duplicate use of its key fields; this would have eventually caused problems for OVIH_PLUS, which is not purged (1). For this reason and also for SQL performance reasons, I made the purchase order number a part of OVIH_PLUS's key field. To further improve performance of various SQL stored procedures, I made the RespOrg a part of the OVIH_PLUS table. In a previous design, the RespOrg had to be obtained from the OPPH table through a join with the OVIL (Open Vendor Invoice Line) table, which contains the purchase order number in its REFTRNSCOD field; this join performed poorly and negatively affected the automated e-mail process.

I also added the InvCopySent and DateInvCopySent fields to the OVIH_PLUS table to facilitate the automated e-mail process and its administration. The InvCopySent

field can have four integer values: 0, 1, 2, and 3; it is initially set to 0 when the record is added to the table. 1, 2, and 3 have the following meanings respectively: No (Not Necessary), Not Yet (Pending), and Yes. Eventually, I defined the nightly process below for the e-mail notification feature:

1. Append records to OVIH_PLUS based on the OVIH download from CUFS.
2. Update records on OVIH_PLUS based on the OVIH & OPPH downloads from CUFS.
3. Test for the existence of the EP number in OnBase. If newly added, update the DocDatestored field in OVIH_PLUS.
4. Set the InvCopySent field to 2 for invoices where notification is necessary.
5. Set the InvCopySent field to 1 for invoices where notification is not necessary.
6. Send e-mail notification for invoices where: the DocDatestored field has a valid date, and the InvCopySent field in OVIH_PLUS has a value of 2.

While developing this process, I realized that the Accounts Payable staff would need some verification of the number of invoices and e-mails that were being sent each day and a way to manage certain problems that may occur in the notification process. I developed the *E-mail Verify* and *Pending E-mail* screens for these purposes. The fields that I added to OVIH_PLUS all helped to provide the necessary data and functionality to these screens.

After the additional licenses for OnBase were installed, we were given approval to proceed with making the Web interface widely available to the University community and to push ahead with the e-mail notification feature. For the announcement of the Web interface and e-mail notification feature, a letter and form were sent out with the monthly financial packets. In the letter, we asked departments to identify their Responsible

Organization codes, designating a contact for each code along with an e-mail address, and to return the completed form to the Accounts Payable Office. Please see Appendix C for a copy of the announcement that went out with the monthly financial packets.

At about the same time of the announcement, Larry Caras and I met with Pete Pfefferkorn, Senior Systems Engineer in charge of the Exchange Server, to discuss the best way for Accounts Payable to handle the PIPES e-mails. Mr. Pfefferkorn recommended that we set up a separate Exchange account for PIPES and then give the appropriate staff in Accounts Payable delegate access to that e-mail account (11). This account was set up in early February to facilitate testing. This account helps the PIPES E-mail Contact administrators in Accounts Payable centrally handle the e-mail notifications, especially when undeliverable notices result from a bad e-mail address in the Contacts table. We are also currently blind copying this e-mail account with the e-mail notifications as added verification of what is sent; since the E-mail Notification program runs outside of the e-mail server, the notifications do not show up in a *Sent Items* folder of the e-mail account.

Shortly after sending out the announcement and form for the PIPES e-mail notification feature, I implemented the first iteration of the PIPES E-mail Contacts Administration interface so that Accounts Payable could begin updating the contacts information in the Pipes_EmailContacts table as soon as forms were returned. Based on information in CUFS OPPH table, I was able to populate the Pipes_EmailContacts table with records for the Responsible Organizations currently processing vendor invoices. I set the interface up for the Pat Swanson, Assistant Director of Accounts Payable, and trained her on how to use it. While Pat was adding/updating contacts from the returned

forms in February and early March, I worked on completing the *E-mail Verify* and *Pending E-mail* screens. As soon as we received enough of the forms back and had contact information for 95% of the Responsible Organizations in our table, we decided to fully implement the PIPES E-mail Notification on March 15, 2004. We also made the decision to continue mailing paper copies of vendor invoices until the end of March or until the e-mail notification feature was functioning successfully. I am happy to say that Accounts Payable was able to stop mailing paper copies of vendor invoices on April 1, 2004!

10. Conclusions and Recommendations

I first recognized the need for something like PIPES from answering calls from users on the CUFS Hotline. It was a difficult task explaining to users, especially new ones, how to navigate online CUFS and put together just a part of the information found with PIPES. Pulling data from a SQL Server warehouse of CUFS information, the Web interface, built with ASP.NET and VB.NET technology, provides more historical and concise information than is available through online CUFS; in fact, it brings together information associated with purchase orders from approximately eight different CUFS sources. This information is consolidated into a logical and easy-to-read manner and greatly enhanced by linking the invoice numbers to the images of the invoice documents in the OnBase system. I also improved access to this information by including a Web page that allows departmental users to conveniently and quickly look up purchase orders by account number. With Microsoft's VB.NET technology, I further extended and enhanced PIPES' capabilities by replacing the Accounts Payable Office's manual system of mailing paper copies of vendor invoices to departments with an automated e-mail

notification system that is easy to manage. The project met all deliverables of the Design Freeze and, in several aspects, exceeded its deliverables. I performed testing in all phases to ensure the success of the product.

Although PIPES has proven to work well and appears to be a success, I would like to improve, given the time, some of its features and functionality. I have already received some good suggestions from users for improving PIPES. Someone pointed out, for example, that the purchase order summary information does not show the Responsible Person or requestor. Another user asked if a search criterion by date could be added to the *Purchase Order Lookup* page to limit the number of purchase orders listed for an account. Senior Systems Engineer in charge of the Exchange Server Senior Systems Engineer in charge of the Exchange Server One thing that I am not completely satisfied with in the E-mail Contacts Administration interface is the feature that allows the administrator to resend an e-mail notification. In order to avoid more coding and testing, I came up with a quick-and-dirty solution to provide this feature; as a result, it takes a bit longer to run than I care for. I would also like to improve that feature by giving the administrator the ability to specify the e-mail address to which the notification is resent.

At this point in time, I probably will not be making any modifications to PIPES because the current financial system, CUFS, is expected to be replaced on July 1, 2005 with SAP Financials. One of best complements and recommendations for PIPES that I have received is that users want something like it to exist in the new financial system.

Larry Caras wrote the following to me recently:

Chris, as I sit in the SAP workshops I am just inundated with positive comments about the functionality of PIPES. Our user community could not be more pleased

with this very useful and intuitive interface that you've designed. I, too, have heard comments much like the following, "I hope that functionality exists in SAP that is as easy to use and interpret as the functionality in PIPES" (2).

PIPES will thus hopefully serve as a model for some of the related configuration of SAP Financials and the kind of functionality that users expect in the new financial system.

Appendix A.

F/B	REF TC NUMBER	ACCEPT DATE	DOC ACT	VENDOR	VENDOR NAME
01-		07 01 02	E	6109094140	ABELL ELEVATOR INT'L
02- B	RX 38050000068	07 01 02	E	6109094140	ABELL ELEVATOR INT'L
03- F	AD 00000421739	08 16 02	E	6109094140	ABELL ELEVATOR INT'L
04- F	PV 03071900131	07 20 02	E	6109094140	ABELL ELEVATOR INT'L
05- F	RC APO03001031	07 19 02	E	6109094140	ABELL ELEVATOR INT'L
06- F	VI 125584	07 19 02	E	6109094140	ABELL ELEVATOR INT'L
07-					
08-					
09-					
10-					
11-					
12-					
13-					
07-*L009 HEADER CHANGE					

Figure 19. Example of a TCP3270 screen and a CUFS table

This particular table is usually the starting point for researching a question about an invoice. The DXRF in TABLEID stands for “Document Cross Reference” (17).

Appendix B.

Sample Copy of Testing Procedures for Web Interface

Testing Procedures for PIPES Web Interface

Return to Chris Huff at ML 0333

Questions: call Chris Huff at 6-1159 or Larry Caras at 6-6772

Name: Laurie Nelson

Please do not share this information or the link to PIPES with anyone else. The link to PIPES is <http://www.uc.edu/controller/queries/pipeshome.aspx>. Additionally, PIPES allows you to link to the Controller Office's imaging system called OnBase to view invoice documents over the Internet. Because of our limited number of licenses for this feature in OnBase, it is important that you always close the browser window of the "document viewer" after printing/viewing the invoice documents; leaving the window open can prevent others from using the "document viewer".

1. Click the link to *PIPES Home Page*.
 - Does *PIPES Home Page* display in your Web browser, or are there problems with its display?
 - If it does not display, what browser and version of browser are you using?
 - If the page displays properly, are the design and layout of the page clear and logical to you? (Please feel free to comment.)
2. Type an invalid PO number into the textbox on *PIPES Home Page*.
 - Does the PO number textbox accept input?
3. With the invalid PO number still in the textbox, click the *Get PO Information* button.
 - Does a message appear, indicating that the PO was not found?
4. Type a valid PO number into the textbox on *PIPES Home Page* and click the *Get PO Information* button.
 - Did the navigation button take you to the *Purchase Order, Invoice and Payment Information* page?
 - If the page displays properly, are the design and layout of the page clear and logical to you? (Please feel free to comment.)

8/4/2003

1

Figure 20. Page 1 of Web interface's testing procedures

Testing Procedures for PIPES Web Interface

Return to Chris Huff at ML 0333

Questions: call Chris Huff at 6-1159 or Larry Caras at 6-6772

5. Compare the information on the *Purchase Order, Invoice and Payment Information* page with the information on CUFS.

- Does the data on the page appear to be correct?

Yes

6. Click on the *New Search* button at the top of the *Purchase Order, Invoice and Payment Information* page. I don't see a "New Search" button, but

- Did the navigation button return you to the *PIPES Home Page*? there's a button for "PIPES Home Page"

Yes

7. Identify a PO that has invoice activity, enter the number in the *PIPES Home Page* textbox and click the *Get PO Information* button. Look for invoices in the *Invoices* section and confirm that a hyperlink appears.

- On the *Purchase Order, Invoice and Payment Information* page, do hyperlinks appear for the invoices listed in the *Invoices* section?

Yes

- For invoices having hyperlinks in the *Invoices* section, do the associated payment vouchers in the *Payment Transactions* section also have invoice numbers with hyperlinks?

Yes

8. Click one of the hyperlinks in either the *Invoices* section or the *Payment Transactions* section. Another browser window should open and the document images for the invoice should appear. This "document viewer" is a part of the Controller's imaging system called OnBase. Test the functionality of the controls for viewing the images.

- Did the hyperlink work—did the "document viewer" appear with the invoice document that you selected?

Yes

9. **Close the "document viewer" window. Because of our limited number of licenses for the "document viewer", it is important that you always close this window after printing/viewing the invoice documents; leaving the window open can prevent others from using the "document viewer".** On the *Purchase Order, Invoice and Payment Information* page, click the *PO Listing by Account* button.

- Did the navigation button take you to the *Purchase Order Lookup* page?

Yes

- If the page displays properly, are the design and layout of the page clear and logical to you? (Please feel free to comment.)

Design + layout are fine

8/4/2003

2

Figure 21. Page 2 of Web interface's testing procedures

Testing Procedures for PIPES Web Interface

Return to Chris Huff at ML 0333

Questions: call Chris Huff at 6-1159 or Larry Caras at 6-6772

10. Click the *Pipes Home Page* button from the *Purchase Order Lookup* page.
 - Did the navigation button return you to the *PIPES Home Page*?
11. On the *Pipes Home Page*, click the *PO Listing by Account* button.
 - Did the navigation button take you to the *Purchase Order Lookup* page?
12. Click the *Get Listing* button on the *Purchase Order Lookup* page.
 - Did error messages appear next to all three of the textboxes for *Required Input*?
13. Enter invalid codes into the *Fund*, *Area*, and *Orgn* textboxes and click the *Get Listing* button.
 - Did a message appear indicating that no records were found? The message will say, "No records found for that account. Please check your input and try again."
14. Enter a valid *Fund*, *Area*, *Orgn* combination that has purchase order activity in the *Required Input* textboxes accordingly.
 - Did a "datagrid" appear with a listing of purchase orders for that account?
15. Try adding valid codes to the *Optional Input* text boxes and then clicking the *Get Listing* button. Hint: use one of the codes that appeared in the datagrid results in the previous step.
 - Did the datagrid results only show data for the *Fund*, *Area*, *Orgn* and optional input code(s) combination?
16. Please list below any other comments you may have for improving these pages.
17. Do you think that you will find PIPES useful in your job?

8/4/2003

3

Figure 22. Page 3 of Web interface's testing procedures

Appendix C.

PIPES Announcement

TO: Departmental Users

FR: Larry Caras
Accounts Payable Office

RE: Announcement of PIPES

January 2004

For the last several months we've been working on a Web interface that brings together consolidated purchase order, invoice and payment information in one location. We have named this Web interface PIPES (Purchase Order, Invoice, Payment Enquiry System). The link to PIPES is <http://www.uc.edu/controller/queries/pipeshome.aspx>. You will also find this link on the Controller's Financial Data Queries page. I hope you can find time to take a look at and start using this interface as soon as possible. A description of PIPES is on the front page of the interface. I believe you will find that once you start tracking your purchase orders you will see that the interface is an easy tool to use. Please note that another feature of this interface is a link to your invoice images that you are able to view and print at your workstation.

As we begin to unroll the PIPES interface, we also are finishing the design portion of the PIPES feature that enables Accounts Payable to stop mailing invoice copies to departments; instead, PIPES will send an e-mail notification with a link to invoices as they become imaged. These e-mails will contain groups of links that are controlled by what is imaged by work day or batch. To that end, we will need you to complete the bottom section of this e-mail and return it. Please bear in mind that we prefer to send one e-mail notification to one unique user per CUFS organization, however, we will accept two names per organization. So be sure to indicate the e-mail addresses of individuals who will be responsible for disseminating this important information on a daily basis.

I hope you can begin to use the PIPES interface. For questions or general assistance, please call Patricia Swanson at 556-6772 or Christopher Huff at 556-1159. Please also be sure to complete the bottom section of this memo and return it per the instructions indicated. For those of you working with multiple CUFS organizations, please make copies of the tear sheet and complete one form for each organization.

Thank you!

.....
To: Pat Swanson (MAIL LOCATION 0333) Date: _____

Fr: _____ CUFS Org: _____

1) Name of individual owning e-mail address: _____

E-mail address: _____

2) Name of individual owning e-mail address: _____

E-mail address: _____

Figure 23. PIPES announcement

Appendix D.

PIPES SQL Server Database Tables, Datasources and Queries

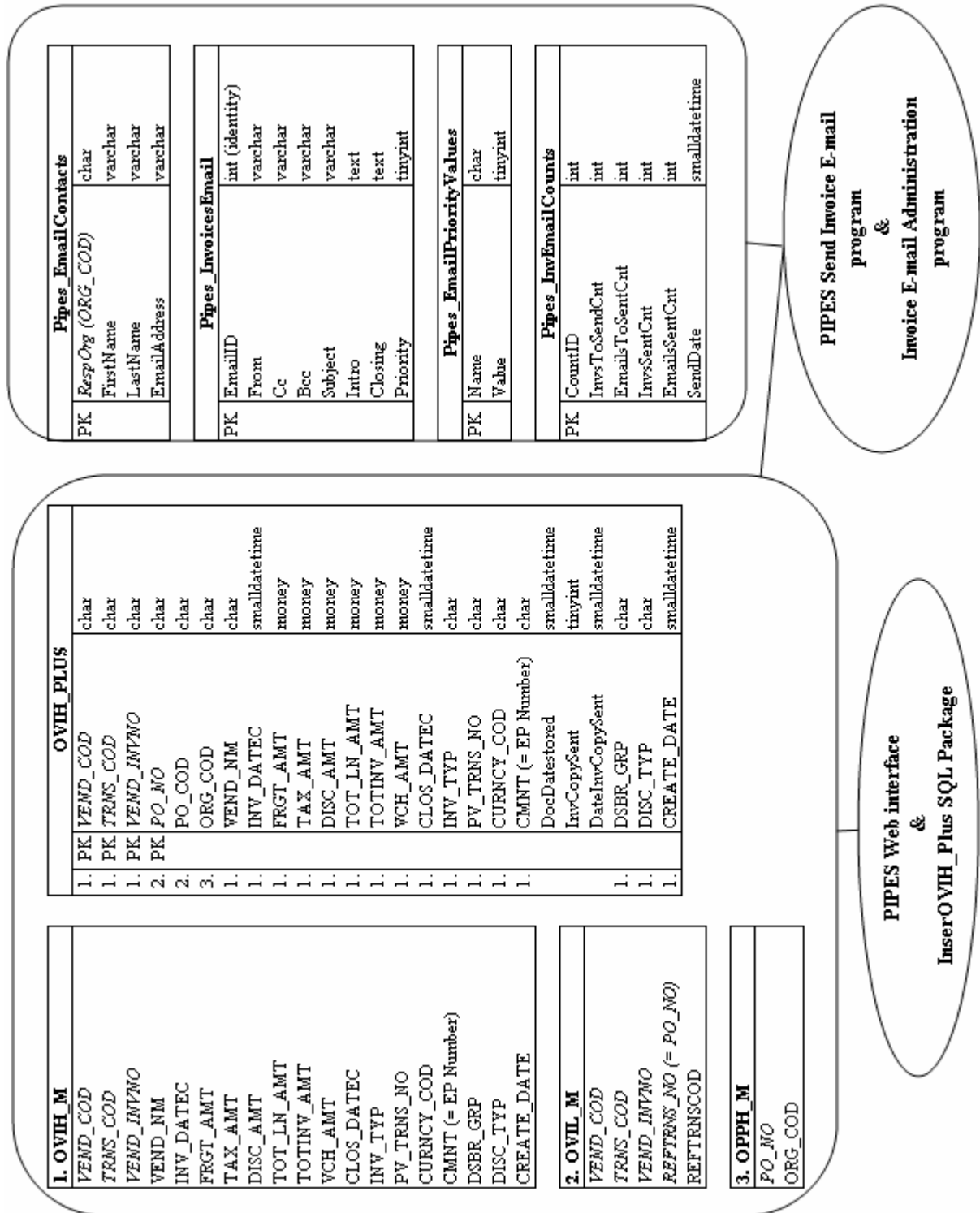


Figure 24. PIPES SQL Server Database Tables

	Data Source Name	Type	Web Interface		Insert OVIH_Plus Nightly SQL Package	Invoice E-mail Administration Program			Pending E-mail Screen	
			Purchase Order, Invoice and Payment Information	Po Lookup		Contacts Screen	E-mail Content Screen	E-mail Verify Screen		
1	WebPipes_LedgerData	Permanent table used for temporary data								
2	vwWebPipes_Ledger	view	X							
3	spWebPipes_EP_OVIHL	stored procedure	X							
4	spWebPipes_LEDGER	stored procedure	X							
5	spWebPipes_MPPHL	stored procedure	X							
6	spWebPipes_OCHT	stored procedure	X							
7	spWebPipes_OPFC	stored procedure	X							
8	spWebPipes_OPFH	stored procedure	X							
9	spWebPipes_OPPL	stored procedure	X							
10	spWebPipes_PoLookup	stored procedure		X						
11	spWebPipes_PTEX	stored procedure	X							
12	spWebPipes_RemitAddress	stored procedure	X							
13	usp_OVIH_PLUS_EpRefChanges	stored procedure			X					
14	usp_OVIH_PLUS_Update	stored procedure			X					
15	usp_OVIH_PLUS_Insert	stored procedure			X					
16	usp_Update DocDatestored OVIH_PLUS	stored procedure			X					
17	usp_OVIH_PLUS_UpdateORG	stored procedure			X					
18	spPipes_UpdateInvCopySent21	stored procedure			X					
19	vwPipes_EP_OVIHL	view							X	
20	vwPipes_InvEmails_ToSend	view							X	
21	spPipes_InvEmailsToSent	stored procedure							X	
22	spPipes_InsertInvEmailCnts	stored procedure							X	
23	spPipes_MarkInvoicesSent	stored procedure							X	
24	spPipes_UpdateInvEmailCnts	stored procedure							X	
25	spPipes_DeleteEmailContacts	stored procedure							X	
26	spPipes_EmailContactsOrgNmns	stored procedure							X	
27	spPipes_InsertEmailContacts	stored procedure							X	
28	spPipes_UpdateEmailContacts	stored procedure							X	
29	spPipes_InvoicesEmail	stored procedure							X	
30	spPipes_UpdateEmailContent	stored procedure							X	
31	spPipes_Pipes_InvCopiesSent	stored procedure							X	
32	spPipes_InvEmailCounts	stored procedure							X	
33	spPipes_InvCopiesPending	stored procedure							X	
34	spPipes_ResetInvCopySent	stored procedure							X	
35	spPipes_UpdateInvsPending	stored procedure							X	

Figure 25. PIPES SQL Server Data Sources & Queries

References

1. Caras, Larry. Director of Accounts Payable at the University of Cincinnati. Personal Interview. November 14, 2002.
2. Caras, Larry. Director of Accounts Payable at the University of Cincinnati. E-mail communication. May 13, 2004.
3. Diers, Rich. Digital Products Manager with Assured Micro-Services. Personal Interview. October 15, 2002.
4. Heilman, Carl. Senior Financial Systems Analyst at the University of Cincinnati. Personal Interview. November 14, 2002.
5. Holmes, Mark. Applications Analyst for University of Cincinnati's Information Technology Department. E-mail communication. July 26, 2002.
6. Holmes, Mark. Applications Analyst for University of Cincinnati's Information Technology Department. E-mail communication. November 20, 2002.
7. Kelley, William. Director of Financial Systems at the University of Cincinnati. Personal Interview. November 12, 2002.
8. Microsoft.com. "HOW TO: Wait for a Shelled Application to Finish by Using Visual Basic .NET". <http://support.microsoft.com/default.aspx?scid=kb;EN-US;305368>. March 25, 2004.
9. MSDN.Microsoft.com. "Code: Creating an E-mail Message (Visual Basic)". http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dv_vbcode/html/vbtskcodeexamplecreatingemailmessage.asp. July 1, 2003.
10. MSDN.Microsoft.com. "Walkthrough: Creating a Multipane User Interface with Window Forms (Visual Basic)". <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/vbcon/html/vbtskcreatingintricateuserinterfacewithwindowsapplicationobjects.asp>. September 29, 2003.
11. Pfefferkorn, Pete. Senior Systems Engineer in charge of the Exchange Server. Personal Interview. February 5, 2004.
12. Roetker, Sally. Business Services Officer in Parking Services. E-mail communication. February 3, 2004.
13. Shop.Microsoft.com. "Visual Studio.NET Professional, Upgrade & Pricing Details". <http://shop.microsoft.com/Referral/productInfo.asp?siteID=11128>. November 14, 2002.

14. UCVHOST.com. "Web Developer Package". http://www.ucvhost.com/web_dev.htm. November 18, 2002.
15. University of Cincinnati. *Branding Standards Manual*. <http://staging.uc.edu/branding/brandingstandards.pdf>. March 2002.
16. University of Cincinnati. Purchase Order Number: PC02A-02621. June 30, 2002.
17. University of Cincinnati's College University Financial System (CUFS). "DXRF" table: CUFS online. November 14, 2002.
18. University of Cincinnati's College University Financial System (CUFS). "End-Users" table: CUFS Security database. November 13, 2002.
19. University of Cincinnati's College University Financial System (CUFS). "OPPH_M" table: CUFS_TBL_M database on Financial Systems' SQL Server. November 13, 2002.
20. University of Cincinnati's College University Financial System (CUFS). "PKR2_M" table: CUFS_TBL_M database on Financial Systems' SQL Server. November 13, 2002.
21. University of Cincinnati. *Web Communication and Design Standards*. <http://staging.uc.edu/branding/webguide.pdf>. May 2002.
22. Yockey, Dennis. Associate Controller for the University of Cincinnati. Personal Interview. November 12, 2002.