The Progress Electronic Magazine

In this issue:

Did you sign up to receive this E-Zine? Send email to sauge@amduus.com to subscribe or fill out the forms at http://www.amduus.com/online/dev/ezine/EZineHome.html ! It's free! (Though donations are certainly welcome – whatever you feel is fair!)

Though intended for users of the software tools provided by Progress Software Corporation, this document is NOT a product of Progress Software Corporation.

Publisher's Statement:

In this issue we examine three ways of sending email from one's progress applications on UNIX and Windows computers. This is a common question in the Progress community, and use of the smtpmail.p program from the FreeFrameWork.org and Amduus.com web sites can be tricky.

Also in the issue is a discussion of creating Requests for Proposals (aka RFPs or Invitation To Tender.) These documents can be sent out to companies to inform them there is something to bid on. This is the first step for getting a practical approach to getting work done with outsourced activities.

I would like to alert everyone that I am looking for work. If any of you have tips about potential projects, I would appreciate hearing from you! I am including my resume with this E-Zine.

More exciting news! I am about to release Service Express which is a pure open source work flow oriented help desk/work order dispatching system. I will be posting it to the Open Source area at Amduus Information Works, Inc. This is a real full blown Webspeed application written by someone who has created similar systems for Fortune 100 companies. I am providing maintenance for the application for $1,000.00 per year. This means your problems will have higher priority, you will receive updates more frequently, there will be special tools available for configuring the application not included in the usual release, as well documentation. Also contributing to the fund will allow there to be money to spend on other developers who can add more stability and features to the code full time.

To your success,

Scott Auge
Founder, Amduus Information Works, Inc.
sauge@amduus.com

Coding Article: Sending E-mail from Progress Applications

Written by Scott Auge

Often times there are requests to send email from one's Progress application. This article examines three means of sending email, each means growing more and more advanced with additional features.

Calling mail

Under UNIX/Linux, there are many tools for achieving actions on the system. One of these is the mail command which can be called directly from the Progress program with arguments to issue E-mail.

The arguments are basic and include: the subject, who the message is meant for, and the body of the message. The from will always be the account running the progress program so this technique has some limitations.

```
/*
 * Written by Scott Auge scott_auge@yahoo.com sauge@amduus.com
 * Copyright (c) 2001 Amduus Information Works, Inc.  www.amduus.com
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. All advertising materials mentioning features or use of this software
 *    must display the following acknowledgement:
 *      This product includes software developed by Amduus Information Works
 *      Inc. and its contributors.
 * 4. Neither the name of Amduus Information Works, Inc. nor the names of
 *    its contributors may be used to endorse or promote products derived
 *    from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY AMDUUS AND CONTRIBUTORS ``AS IS'' AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED.  IN NO EVENT SHALL THE AMDUUS OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 *
 */

DEF VAR RCSVersion AS CHARACTER INIT "$Header:
/home/sauge/code/progress/misc/RCS/mail.p,v 1.1 2001/11/12 18:10:24 sauge Exp
sauge $" NO-UNDO.

DEF INPUT PARAMETER pSubject   AS CHARACTER NO-UNDO.
DEF INPUT PARAMETER pTo        AS CHARACTER NO-UNDO.
DEF INPUT PARAMETER pMessage   AS CHARACTER NO-UNDO.

DEF VAR lFileName    AS CHARACTER NO-UNDO.
DEF VAR lMailCommand AS CHARACTER NO-UNDO.

{MakeID2.i}

ASSIGN lFileName = "/tmp/" + MakeID2(10) + ".mail".

OUTPUT TO VALUE(lFileName).
PUT UNFORMATTED pMessage.
OUTPUT CLOSE.

ASSIGN lMailCommand = "cat " + lFileName + " | mail -s ~"" + pSubject + "~" " +
pTo.

UNIX VALUE(lMailCommand).


OS-DELETE VALUE(lFileName).
```

The following code is used to call the program to generate a mail message.

```
run mail.p (
input "test",
input "sauge@amduus.com",
input "This is the message!"
).
```

The programmer can have the message go to multiple people by issuing a space delimited address list.

Calling Sendmail

The following program is a bit more flexible as the sender's ID is does not need to be the account running the application. This can be useful for such things as: automatic replying of messages from an e-commerce site, to confirm to a customer that something like an order has been shipped with the return address as to a certain support person, or for warnings about some condition that has occurred within the system.

An added benefit, is that the program can also include attachments that are uuencoded. Be aware, that some mail client programs do not handle uuencoded attachments very well, leaving the user with base 64 encoded slop in the email body. Most major email clients support this sort of encoding, but some web based email clients (such as yahoo) do not. They require their attachments to be mime encoded[1].

In addition, the program running the sendmail.p will need in it's path the location of sendmail executable. This is usually not part of the path for a common user – and if need be, the call to sendmail can be hard coded or configuration coded into the source code to help the program find it.

```
[~/code/progress/misc]$ export PATH=$PATH:/usr/sbin
[~/code/progress/misc]$ which sendmail
/usr/sbin/sendmail
```

The program calls sendmail which is a mail transport delivery agent. This means sendmail knows how to find what host out on the internet will receive email for a given user. If that host is down for a while, sendmail will continue to attempt to deliver the mail (if so configured) so there is more assured delivery of the mail.

Below is the listing for the sendmail.p program[2]:

```
/*
 * Written by Scott Auge scott_auge@yahoo.com sauge@amduus.com
 * Copyright (c) 2001 Amduus Information Works, Inc.  www.amduus.com
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
```

[1] It is possible to adapt this program to mime encode multiple file attachments

[2] Note this program does not check for error conditions from the call into sendmail. It assumes everything went smoothly. If one needs a recording of the messages sent out for resending, do not delete the message text file as the program does at the end of the listing.

```
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. All advertising materials mentioning features or use of this software
 *    must display the following acknowledgement:
 *      This product includes software developed by Amduus Information Works
 *      Inc. and its contributors.
 * 4. Neither the name of Amduus Information Works, Inc. nor the names of
 *    its contributors may be used to endorse or promote products derived
 *    from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY AMDUUS AND CONTRIBUTORS ``AS IS'' AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED.  IN NO EVENT SHALL THE AMDUUS OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 *
 */


DEF INPUT PARAMETER pEmailFrom        AS CHAR NO-UNDO.
DEF INPUT PARAMETER pEmailTo          AS CHAR NO-UNDO.
DEF INPUT PARAMETER pEmailCC          AS CHAR NO-UNDO.
DEF INPUT PARAMETER pEmailCCDelimiter AS CHAR NO-UNDO.
DEF INPUT PARAMETER pAttachment       AS CHAR NO-UNDO.
DEF INPUT PARAMETER pAttachmentName   AS CHAR NO-UNDO.
DEF INPUT PARAMETER pSubject          AS CHAR NO-UNDO.
DEF INPUT PARAMETER pMessageData      AS CHAR NO-UNDO.

DEF VAR LSystemName                   AS CHAR NO-UNDO FORMAT "X(50)".
DEF VAR LTempFile                     AS CHAR NO-UNDO INIT "/tmp/".
DEF VAR LEmailScript                  AS CHAR NO-UNDO INIT "/tmp/".
DEF VAR i_EmailCCCounter              AS INT  NO-UNDO.
DEF VAR cDelimiter                    AS CHAR NO-UNDO.
DEF VAR iCounter                      AS INT  NO-UNDO.

DEF STREAM email.
```

```
/*** Message bodies containing ~n do not work.  Change them to ~r ***/
ASSIGN
  cDelimiter   = pEmailCCDelimiter
  pMessageData = REPLACE(pMessageData,"~n","~r")
  LEmailScript = LEmailScript + USERID("dictdb") +
                 STRING(ABSOLUTE(etime), "999999999999").


  /*** If Attachment and can find the Attachment ***/
  DO iCounter = 1 TO NUM-ENTRIES(pAttachment,cDelimiter):
    IF pAttachment <> "" AND
      SEARCH(ENTRY(iCounter,pAttachment,cDelimiter)) <> ? THEN DO:
      /*** If no Attachment Name then Use a default name ***/
      IF ENTRY(iCounter,pAttachmentName,cDelimiter) = "" THEN
        ASSIGN
          ENTRY(iCounter,pAttachmentName,cDelimiter) = "Unnamed.txt".
      ELSE /*** Replace Spaces in Name with _ (uuencode fails with spaces) ***/
        ASSIGN
          ENTRY(iCounter,pAttachmentName,cDelimiter) =
             REPLACE(ENTRY(iCounter,pAttachmentName,cDelimiter)," ","_").
    END.
  END. /*** DO iCounter ***/

INPUT THROUGH VALUE("uname -n") NO-ECHO.
set LSystemName.
INPUT CLOSE.

OUTPUT STREAM email TO VALUE(LEmailScript).

PUT STREAM email UNFORMATTED
  "HELO " LSystemName           SKIP
  "MAIL FROM:<" pEmailFrom ">" SKIP.

/* Loop through all To's */
IF pEmailTo <> "" THEN
DO iCounter = 1 TO NUM-ENTRIES(pEmailTo,cDelimiter):
  PUT STREAM email UNFORMATTED
    "RCPT TO:<" ENTRY(iCounter,pEmailTo,cDelimiter) ">" SKIP.
END.
/* Loop through all CC's */
IF pEmailCC <> "" THEN
DO i_EMailCCCounter = 1 TO NUM-ENTRIES(pEmailCC,pEmailCCDelimiter):
  PUT STREAM email UNFORMATTED
    "RCPT TO:<" ENTRY(i_EmailCCCounter,pEmailCC,pEmailCCDelimiter) ">" SKIP.
END.

PUT STREAM email UNFORMATTED
```

```
"DATA"                       SKIP
"Subject: "  pSubject        SKIP
"From: " pEmailFrom          SKIP.

/* Loop through all To's */
IF pEmailTo <> "" THEN
DO iCounter = 1 TO NUM-ENTRIES(pEmailTo,cDelimiter):
  PUT STREAM email UNFORMATTED
    "To:<" ENTRY(iCounter,pEmailTo,cDelimiter) ">" SKIP.
END.
/* Loop through all CC's */
IF pEmailCC <> "" THEN
DO i_EMailCCCounter = 1 TO NUM-ENTRIES(pEmailCC,pEmailCCDelimiter):
  PUT STREAM email UNFORMATTED
    "Cc:<" ENTRY(i_EmailCCCounter,pEmailCC,pEmailCCDelimiter) ">" SKIP.
END.

/*** You can Now send multiple attachments
IF pAttachment <> "" AND SEARCH(pAttachment) <> ? THEN DO:
  OUTPUT STREAM email CLOSE.
  OS-COMMAND uuencode
     VALUE(pAttachmentName + " < " + pAttachment + " >> " + LEmailScript).
  OUTPUT STREAM email TO VALUE(LEmailScript) APPEND.
END. /*** IF pAttachment <> "" THEN DO: ***/
 *** You can now send multple attachments ***/

PUT STREAM email UNFORMATTED
  SKIP(1) pMessageData SKIP(1).

/*** Handle Multpile Attachments      ***/
/*** Must Use delimiter for CC's      ***/
DO iCounter = 1 TO NUM-ENTRIES(pAttachment,cDelimiter):
  IF ENTRY(iCounter,pAttachment,cDelimiter) <> "" AND
    SEARCH(ENTRY(iCounter,pAttachment,cDelimiter)) <> ?
  THEN DO:

    OUTPUT STREAM email CLOSE.
    OS-COMMAND SILENT uuencode
      VALUE(ENTRY(iCounter,pAttachmentName,cDelimiter)
      + " < "
      + ENTRY(iCounter,pAttachment,cDelimiter)
      + " >> " + LEmailScript).
    OUTPUT STREAM email TO VALUE(LEmailScript) APPEND.
  END. /*** IF pAttachment <> "" THEN DO: ***/
END. /*** Do iCounter = 1 to  ***/
```

```
PUT STREAM email UNFORMATTED
  "."                          SKIP
  "QUIT"                       SKIP.

OUTPUT STREAM email CLOSE.

OS-COMMAND SILENT
  sendmail -bs < VALUE(LEmailScript) 1>/dev/null.
```

Here is an example call of the program:

```
RUN sendmail.p (
INPUT "sauge@amduus.com",
INPUT "sauge@amduus.com",
INPUT "sales@amduus.com",
INPUT ",",
INPUT "/tmp/logfile",
INPUT "logfile.txt",
INPUT "test sendmail",
INPUT "This is a test of the sendmail.p program."
).
```

Calling smtpmail.p

The smtpmail.p program was the combination of programmers who’s names can be found in the comment section. In my opinion, this is one of the most powerful routines available in the open source arena for Progress applications.

The smtpmail.p program can be found at the FreeFrameWorks.org website: http://freeframework.org. and Amduus Information Works Inc. website http://www.amduus.com. You will need two files: smtpmail.p and b64encode.p. You will need to place b64encode.p into a subdirectory called ffwlib or touch up smtpmail.p to find it where you place it.

By using an SMTP connection, the program can communicate with TDAs (Transport Delivery Agent) that are not on the computer running the Progress application. This means TDAs on different OS’s (AIX, Linux, Apple, Microsoft, UNIX) and of different types (Exchange, sendmail, Portmailer, etc) can be used with this code.

The following are some sample calls to the program to achieve different things[3].

Call to send a simple text based message

```
/* Simple text message with smtpmail.p */

DEF VAR oSuccessful   AS LOGICAL NO-UNDO.
DEF VAR vMessage      AS CHARACTER NO-UNDO.

RUN smtpmail.p (
INPUT "localhost",
INPUT "sauge@amduus.com",
INPUT "Automail@amduus.com",
INPUT "",
INPUT "",
INPUT "",
INPUT "Test 1",
INPUT "Simple text message",
INPUT "",
INPUT "text",
OUTPUT oSuccessful,
OUTPUT vMessage
).

MESSAGE oSuccessful.
MESSAGE vMessage.
```

Call to send an HTML based message

```
/* Simple html message with smtpmail.p */

DEF VAR oSuccessful   AS LOGICAL NO-UNDO.
DEF VAR vMessage      AS CHARACTER NO-UNDO.

RUN smtpmail.p (
INPUT "localhost",
INPUT "sauge@amduus.com",
INPUT "Automail@amduus.com",
INPUT "",
INPUT "",
INPUT "",
INPUT "Test 2",
INPUT "<html><body><b>Simple HTML message</b></body></html>",
INPUT "type=text/html:charset=sascii",
INPUT "text",
OUTPUT oSuccessful,
```

[3] I have code that given an attachment name and a commonly used description will generate the proper MIME header information. This can be a bit of a nuisance to sort out by hand in the call.

```
OUTPUT vMessage
).

MESSAGE oSuccessful.
MESSAGE vMessage.
```

Note that this program does not yet handle CIDs, which allows one to send graphics or other types of embedded references within the email. Hence, if you are sending graphics in your HTML formatted email, you will need to SRC from the machine completely, as in `<IMG SRC=”http://www.amduus.com/images/myimage.gif”>`.

Call to send a simple text based message with an attachment

```
/* Simple text message including file attachment with smtpmail.p */

DEF VAR oSuccessful   AS LOGICAL NO-UNDO.
DEF VAR vMessage      AS CHARACTER NO-UNDO.

RUN smtpmail.p (
INPUT "localhost",
INPUT "sauge@amduus.com",
INPUT "Automail@amduus.com",
INPUT "",
INPUT "ezine1.doc:type=application/msword:filetype=binary",
INPUT "/home/sauge/ezine/issues/ezine1.doc",
INPUT "Test 3",
INPUT "Text message with file attachment~nHave fun!",
INPUT "",
INPUT "multipart/related",
OUTPUT oSuccessful,
OUTPUT vMessage
).

MESSAGE oSuccessful.
MESSAGE vMessage.
```

Call to send an HTML based message with an attachment

```
/* Simple HTML message including file attachment with smtpmail.p */

DEF VAR oSuccessful   AS LOGICAL NO-UNDO.
DEF VAR vMessage      AS CHARACTER NO-UNDO.

RUN smtpmail.p (
INPUT "localhost",
INPUT "sauge@amduus.com",
INPUT "Automail@amduus.com",
INPUT "",
INPUT "ezine1.doc:type=application/msword:filetype=binary",
INPUT "/home/sauge/ezine/issues/ezine1.doc",
INPUT "Test 4",
INPUT "<html><body><b>HTML Here!</b></body></html>",
INPUT "type=text/html:charset=usascii",
INPUT "multipart/related",
OUTPUT oSuccessful,
OUTPUT vMessage
).

MESSAGE oSuccessful.
MESSAGE vMessage.
```

About the author: Scott Auge is the founder of Amduus Information Works, Inc. He has been programming in the Progress environment since 1994. His works have included E-Business initiatives and focuses on web applications on UNIX platforms.
sauge@amduus.com

New Open Source Application: Service Express

Written by Scott Auge sauge@amduus.com

Below are some screen prints for the new Open Source application developed at Amduus Information Works, Inc. These are not all the available screens, but will give you some ideas of what the system is about.

The application runs on Webspeed and Blue Diamond.

The application being Open Source, of course means you get all the source code. Maintenance and support is available by contacting sales@amduus.com. There will be releases without maintenance, but those with maintenance will have them first.

It allows a web based call taking system to be set up. Users who place tickets into the application would use the following screens. These users (external users) would only be allowed to work with their tickets.

Users who handle the tickets would use another set of screens. These users are internal users and satisfy the tickets.

The system is fully configurable. The screen prints below are oriented to a configuration for Blue Diamond and Service Express problem reports.

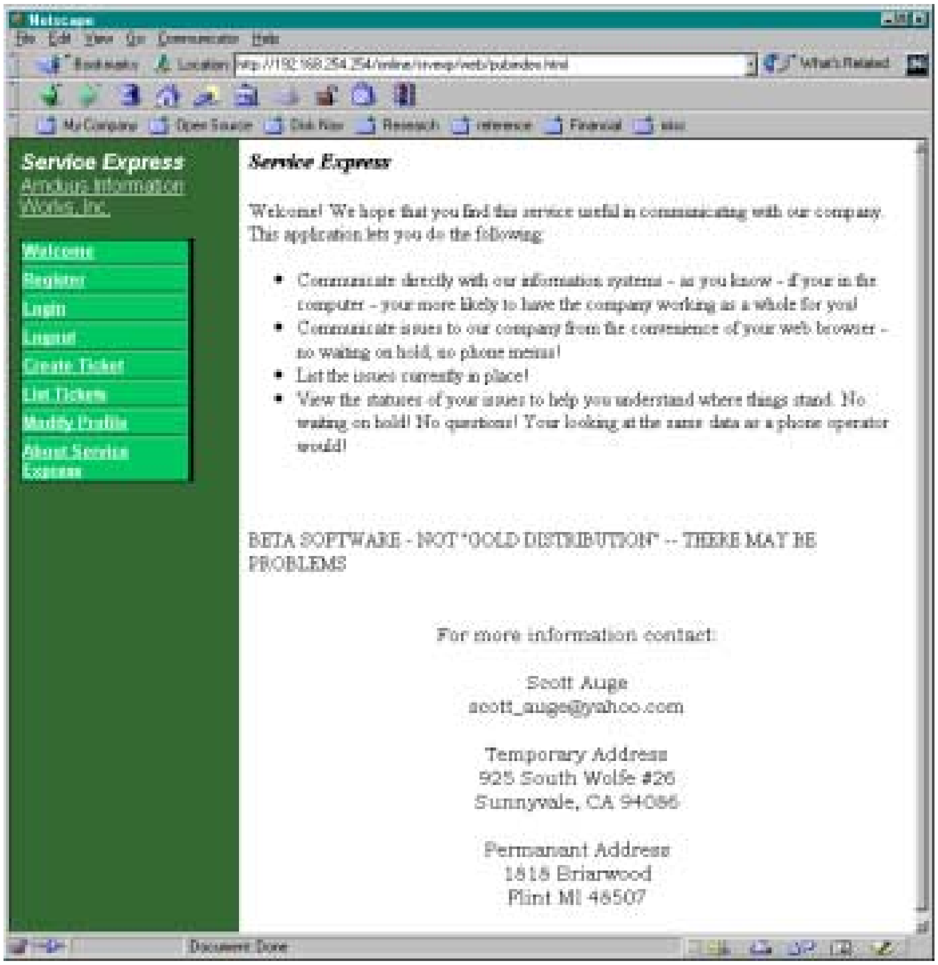
Tickets can be of a definable type. A ticket type can be thought of as a form of interaction with the organization. Some types might be accounting/billing oriented. Other types might be a request for service to be performed on some kind of equipment.

Each ticket type can have a problem type. A problem type allows further resolution to the organization of the information for easy delivery to people to interact with the customer with.

Each ticket type can have a set of definable statuses. These statuses help the organization understand where the ticket stands in the process of handling it.

Each problem type has a priority. This is so that certain problem types can be brought to the attention of the organization for more immediate resolution.

The system handles work flow, where by the status is used to determine which organizational position handles tickets of that status. When the status is changed, the ticket is immediately assigned to someone in that position.


Netscape
File Edit View Go Communicator Help
Bookmarks Location: http://192.168.254.254/online/srvexp/web/pubindex.html
What's Related
My Company Open Source Dish Nav Research reference Financial misc
Welcome
Register
Login
Logout
Create Ticket
List Tickets
Modify Profile
About Service Express
Document: Done

Service Express
Amduus Information Works, Inc.

Service Express

Welcome! We hope that you find this service useful in communicating with our company. This application lets you do the following:

- Communicate directly with our information systems – as you know - if your in the computer – your more likely to have the company working as a whole for you!
- Communicate issues to our company from the convenience of your web browser – no waiting on hold, no phone menus!
- List the issues currently in place!
- View the statuses of your issues to help you understand where things stand. No waiting on hold! No questions! Your looking at the same data as a phone operator would!

BETA SOFTWARE - NOT "GOLD DISTRIBUTION" -- THERE MAY BE PROBLEMS

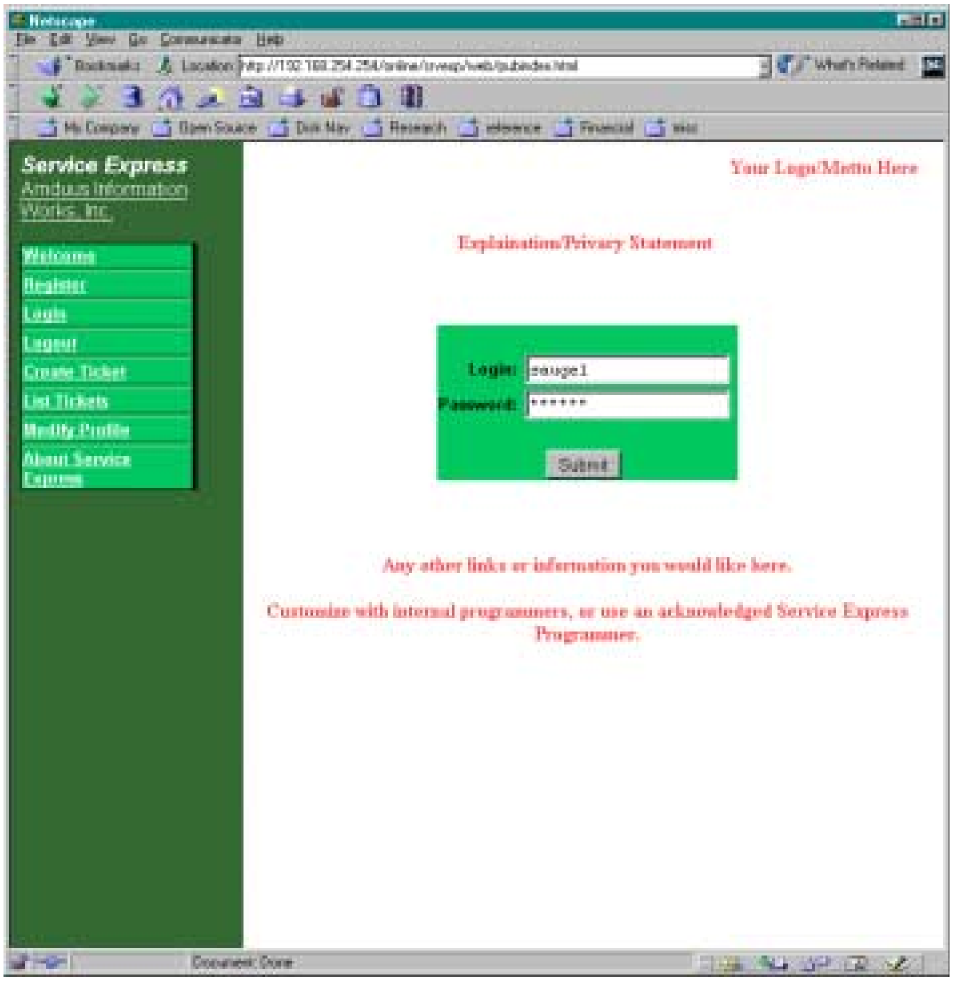
For more information contact:

Scott Auge
scott_auge@yahoo.com

Temporary Address
925 South Wolfe #26
Sunnyvale, CA 94086

Permanant Address
1818 Briarwood
Flint MI 48507

The welcome page to Service Express as seen by someone who wants to interact with your organization.



Your Logo/Motto Here

Explaination/Privacy Statement

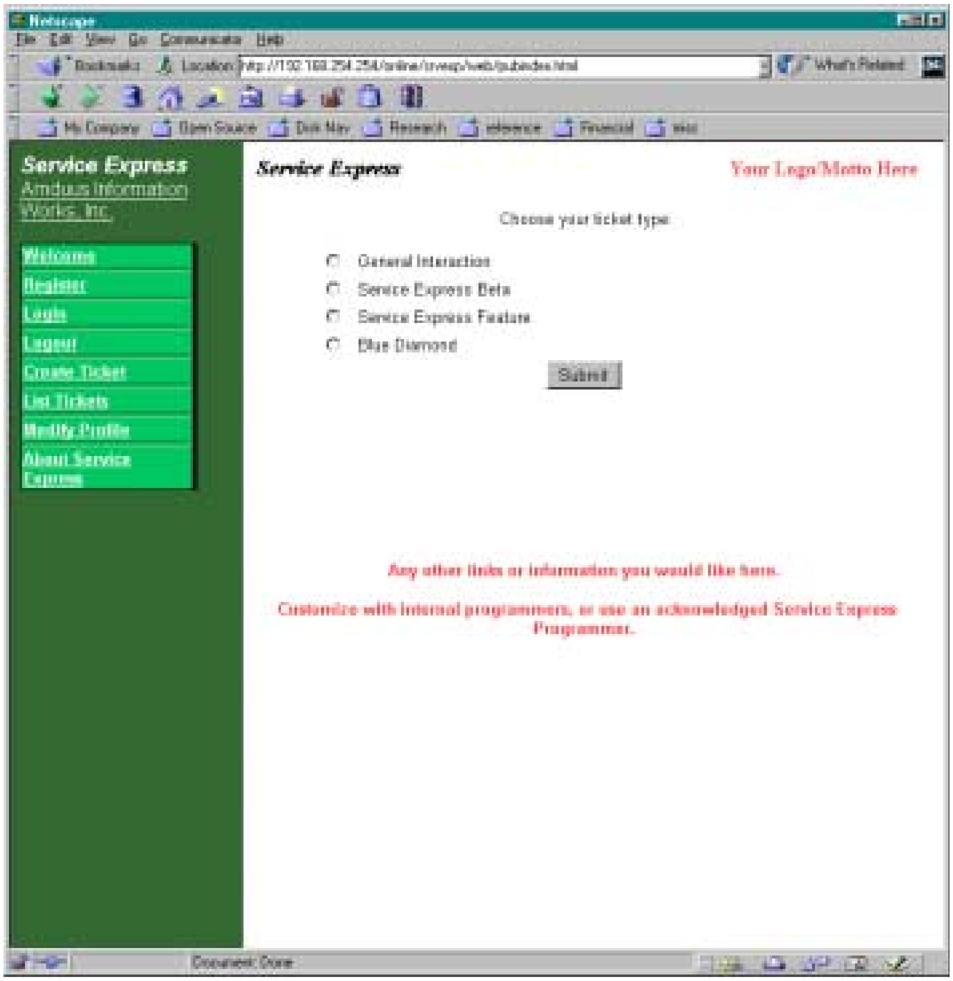
Login: sauge1

Password: ******

Any other links or information you would like here.

Customize with internal programmers, or use an acknowledged Service Express Programmer.

Once the user has registered with your company, they would log in.



Netscape
File Edit View Go Communicator Help
Bookmarks Location: http://192.168.254.254/online/srvexp/web/pubindex.html
What's Related
My Company Open Source Dist Nav Research reference Financial misc
Welcome
Register
Login
Logout
Create Ticket
List Tickets
Modify Profile
About Service Express
Document: Done

Service Express
Amduus Information Works, Inc.

Service Express

Your Logo/Motto Here

Choose your ticket type:

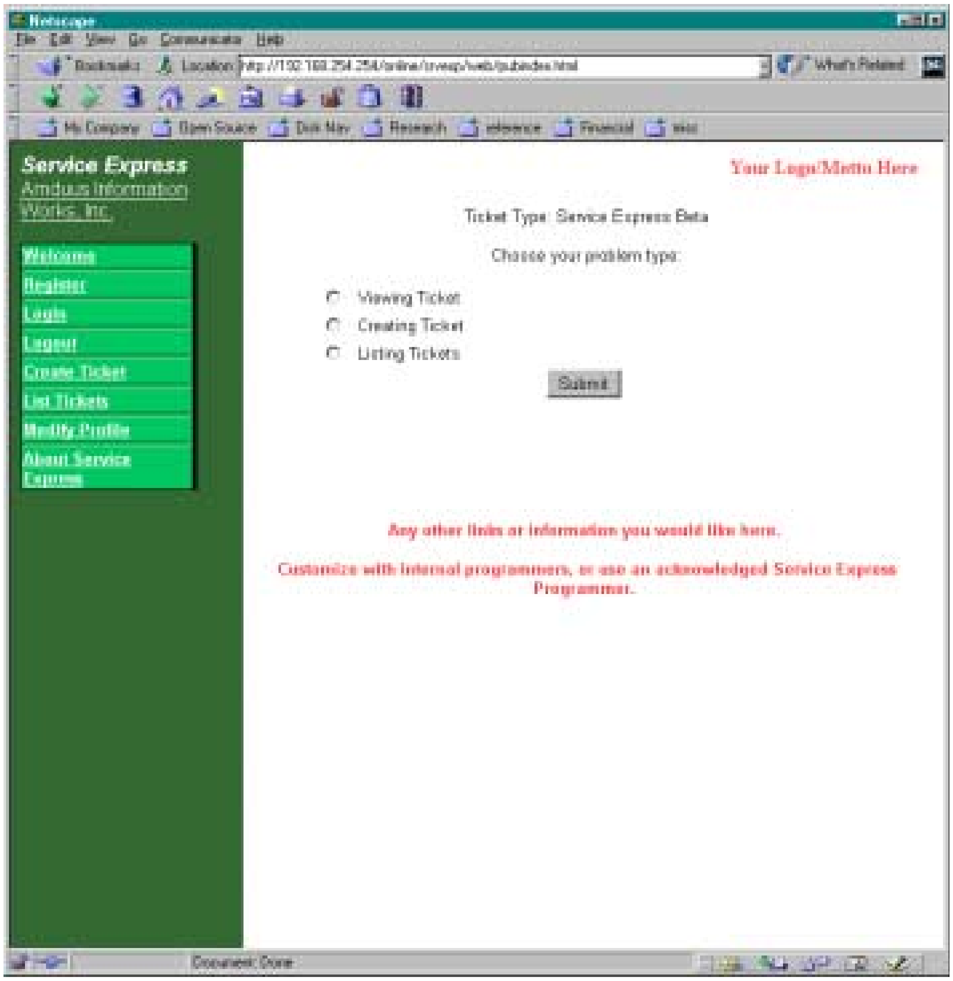
○ General Interaction
○ Service Express Beta
○ Service Express Feature
○ Blue Diamond

Submit

Any other links or information you would like here.

Customize with internal programmers, or use an acknowledged Service Express Programmer.

To create a ticket, the user chooses the ticket type. These are fully configurable within the application. Ticket types can be used for reaching different departments, handling different kinds of equipment, or requesting different kinds of services.



Netscape
File Edit View Go Communicator Help
Bookmarks Location: http://192.168.254.254/online/srvexp/web/pubindex.html
What's Related
My Company Open Source Dok Nav Research reference Financial misc
Document: Done

Service Express
Amduus Information Works, Inc.

Welcome
Register
Login
Logout
Create Ticket
List Tickets
Modify Profile
About Service Express

Your Logo/Motto Here

Ticket Type: Service Express Beta

Choose your problem type:

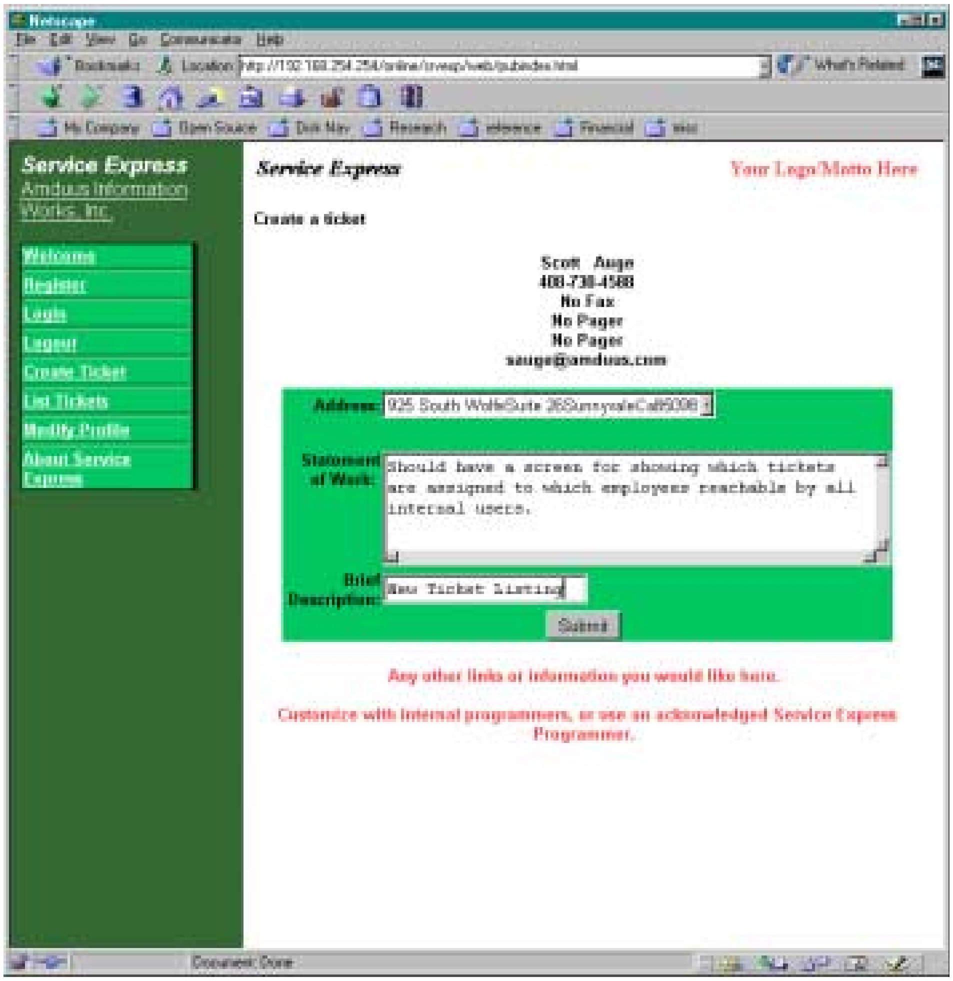
○ Viewing Ticket
○ Creating Ticket
○ Listing Tickets

Submit

Any other links or information you would like here.

Customize with internal programmers, or use an acknowledged Service Express Programmer.

Each ticket type has definable problem types. This lets problems automatically reach people who know how to handle those kinds of problems.


Netscape
File Edit View Go Communicator Help
Bookmarks Location: http://192.168.254.254/online/srvexp/web/pubindex.html
What's Related
My Company Open Source Dish Nav Research reference Financial misc
Service Express
Amduus Information Works, Inc.
Welcome
Register
Login
Logout
Create Ticket
List Tickets
Modify Profile
About Service Express
Service Express
Your Logo/Motto Here
Submit
Document: Done

Create a ticket

Scott Auge
408-738-4588
No Fax
No Pager
No Pager
sauge@amduus.com

Address: 925 South WolfeSuite 26SunnyvaleCa95086

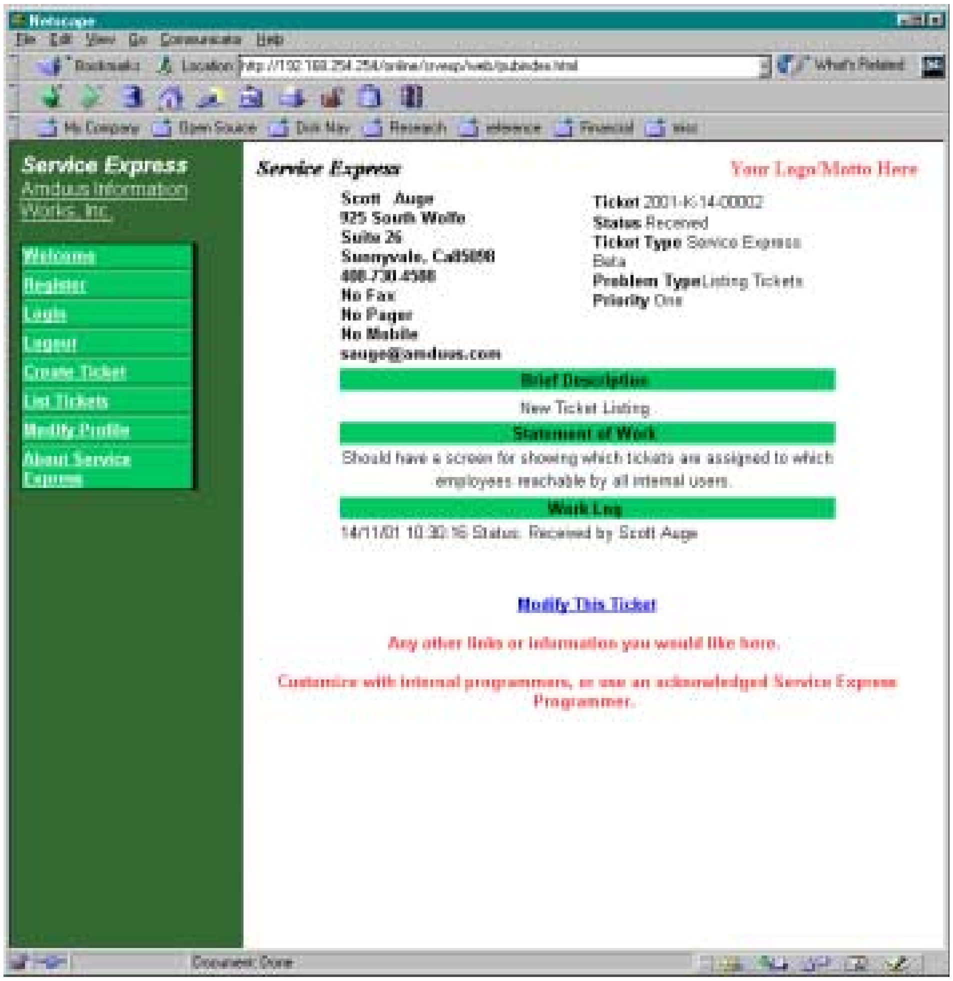
Statement of Work: Should have a screen for showing which tickets are assigned to which employees reachable by all internal users.

Brief Description: New Ticket Listing

Any other links or information you would like here.

Customize with internal programmers, or use an acknowledged Service Express Programmer.

The user would enter their statement of work and a brief description.


Netscape
File Edit View Go Communicator Help
Bookmarks Location: http://192.168.254.254/online/sveasp/web/pubindex.html
What's Related
My Company Open Source Dish Nav Research reference Financial misc
Service Express
Amduus Information Works, Inc.
Welcome
Register
Login
Logout
Create Ticket
List Tickets
Modify Profile
About Service Express
Document: Done

Service Express

Your Logo/Motto Here

Scott Auge
925 South Wolfe
Suite 26
Sunnyvale, Ca95098
408-730-4588
No Fax
No Pager
No Mobile
sauge@amduus.com

Ticket 2001-K-14-00002
Status Received
Ticket Type Service Express Beta
Problem Type Listing Tickets
Priority One

Brief Description

New Ticket Listing

Statement of Work

Should have a screen for showing which tickets are assigned to which employees reachable by all internal users.

Work Log

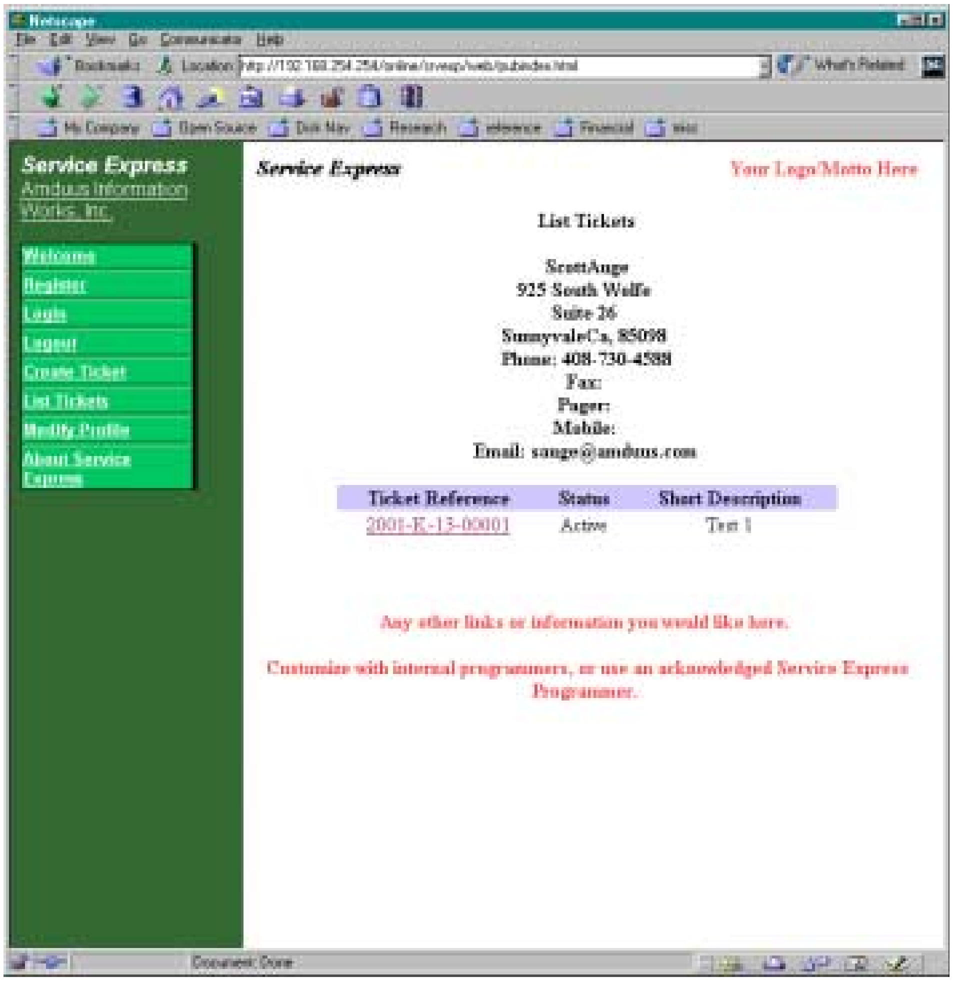
14/11/01 10:30:16 Status: Received by Scott Auge

Modify This Ticket

Any other links or information you would like here.

Customize with internal programmers, or use an acknowledged Service Express Programmer.

Upon pressing submit, their ticket would be created.


Netscape
File Edit View Go Communicator Help
Bookmarks Location: http://192.168.254.254/online/srvexp/web/pubindex.html
What's Related
My Company Open Source Dick Nav Research reference Financial misc
Welcome
Register
Login
Logout
Create Ticket
List Tickets
Modify Profile
About Service Express
Document: Done

Service Express
Amduus Information Works, Inc.

Service Express Your Logo/Motto Here

List Tickets

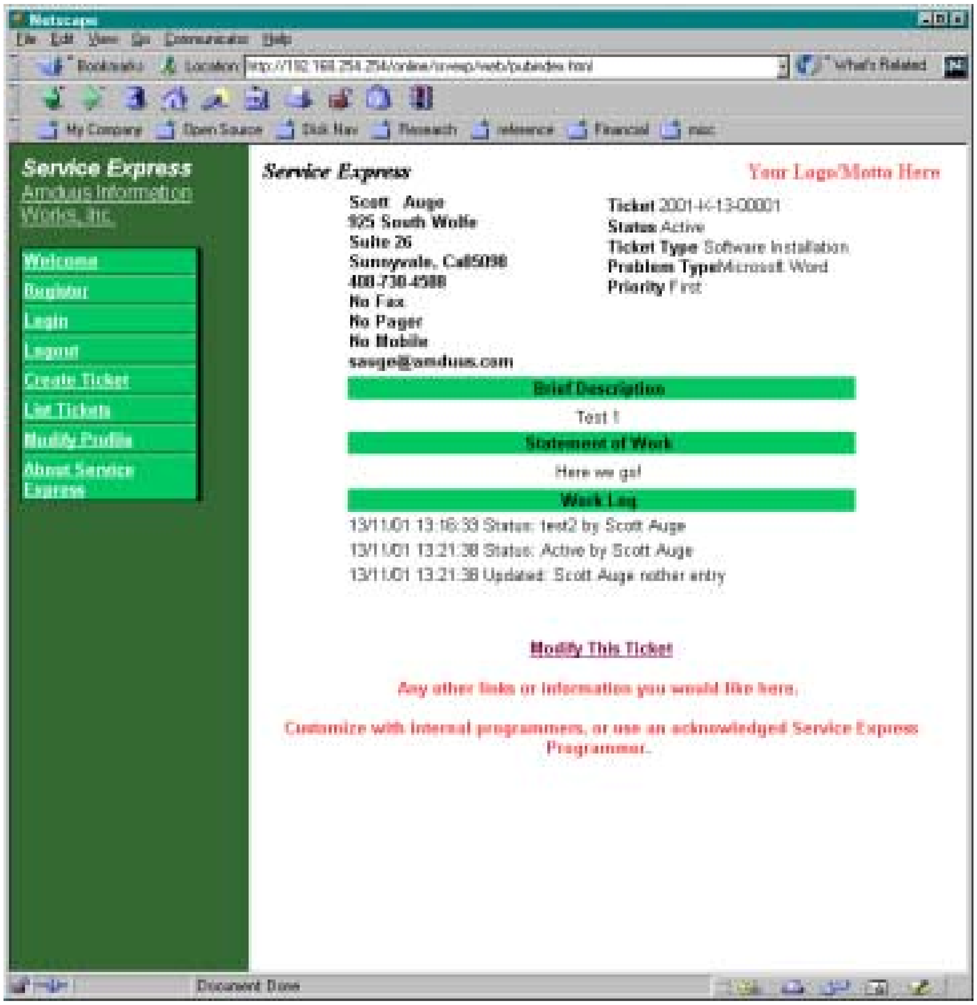
**ScottAuge
925 South Wolfe
Suite 26
SunnyvaleCa, 85098
Phone: 408-730-4588
Fax:
Pager:
Mobile:
Email: sauge@amduus.com**

Ticket Reference	Status	Short Description
2001-K-13-00001	Active	Test 1

Any other links or information you would like here.

Customize with internal programmers, or use an acknowledged Service Express Programmer.

Then they can list their current tickets in this screen.



Service Express

Your Logo/Motto Here

Scott Auge
925 South Wolfe
Suite 26
Sunnyvale, Ca85098
408-730-4588
No Fax
No Pager
No Mobile
sauge@amduus.com

Ticket 2001-K-13-00001
Status Active
Ticket Type Software Installation
Problem TypeMicrosoft Word
Priority First

Brief Description

Test 1

Statement of Work

Here we go!

Work Log

13/11/01 13:16:33 Status: test2 by Scott Auge
13/11/01 13:21:38 Status: Active by Scott Auge
13/11/01 13:21:38 Updated: Scott Auge nother entry

Modify This Ticket

Any other links or information you would like here.

Customize with internal programmers, or use an acknowledged Service Express Programmer.

And they can review the status of the ticket as well as information about the solution of the ticket.

The source and database definitions for the software will be available from http://www.amduus.com in the very near future.

Management Article: Creating Requests For Proposals

Written by Scott Auge sauge@amduus.com

There comes a time when work needs to be done on the organization’s information systems. Often this work can be outsourced to a vendor, but how to go about creating a document that vendor can use to bid on the work?

One way to get all the facts in order so as the people who might wish to perform the change can spec out the work to be done is the preparation of a Request For Proposal. (RFP)

Statement of the Problem or Project Objectives

This covers briefly what is hoped to be achieved by the solution. It would include description of the current state of the system, and description of the problem. It would include supporting documentation such as reports, diagrams, and examples. It would list the objectives and goals of the project.

Technical Requirements

Covers such items as: interfaces to existing systems, source code required, database requirements, communications and network architecture, military or government standards required, reliability requirements, stability requirements, timing constraints, programming languages used, information about the host computers used.

Administrative Information

Covers such things as: Who may respond to the RFP, how to request clarification or more information about items in the RFP, date and locations for scheduled meetings, proposal selection criteria, as well other administrative information.

Cost Requirements

This would help the proposal's provider understand the forms of payment they could expect and other solutions they could work under. Could be combinations of separate pricing for each phase, type of development contract offered, cost comparisons of alternate solutions, pricing structure for service, coding, support, etc.

Be careful on this one or you may rule out some capable contractors off the top. Be clear that your organization is flexible on this IF it is flexible.

Referenced Documents

These are documents that can be asked for or would ride along with the RFP. They include standards, existing system documentation, and product collateral.

Required Deliverables

This is the crux of what is expected of the contractor. It includes items such as documentation, software, training, relevant hardware and equipment, warranties, development and test tools.

Proposal Format

This will cover the expected items found in the proposal presented as an answer to the RFP. These proposal items will be listed in the next coming issues of the E-Zine. In short though, they would be the technical proposal, management proposal, pricing proposal and the statement of work. Expected supplements might include résumé's of the key developers, as well as case histories of the organization.

Submission Schedule and Decision Schedule

This would cover such information as the start and final dates to submit the proposal under, expected date the selection would be made under, and preferred schedule for the completion of the project.

Next issue will discuss the technical, management, and pricing proposal formats.

Portions of this article based on:

E.M. Bennatan, "On Time, Within Budget – Software Project Management Practices and Techniques" ISBN 0-89435-408-6

About the author: Scott Auge is the founder of Amduus Information Works, Inc. He has been programming in the Progress environment since 1994. His works have included E-Business initiatives and focuses on web applications on UNIX platforms.
sauge@amduus.com

Contractors/Consulting Companies Available:

Community Announcements:

A place to announce Progress User Groups, Open Source Exchanges, etc. No charge!

Wonder where to find those Progress marketing articles?

http://www.progress.com/analyst/
http://www.progress.com/profiles/index.htm
http://www.progress.com/success/index.htm

Product Announcements:

Survey Software

Amduus Information Works, Inc. is creating survey software. This software can be used on a web site to query a population of people about their views and needs. The population could be internal to a company or external to yield a better understanding of the marketplace. Documentation for the application will be available at http://www.amduus.com for free download.

The software ships with source code for better adaptability to your company's application landscape and needs. The software was developed and designed on Linux with Progress Version 9 and Webspeed 3.1. The software can operate on AIX, Linux, Solaris, HP-UX and UNIXWare. The software will also run on Blue Diamond available for free from Amduus Information Works, Inc.

Customers and resellers are welcome to contact Scott Auge at sauge@amduus.com for more information. Street price is $1,000 per machine without Progress licenses.

The software can be rented out at $100.00 per survey per week of taking results.

Mail List Software

The publishing of this E-Zine has created the need to create some software that can handle the creation of mailing lists, web pages to subscribe and unsubscribe as well as a command line to distribute this file. This software is available for $50.00 per site. The User Guide is available at http://www.amduus.com/Manuals/mls.doc . Purchase before October 15th and get a free update that will have a subscription verification feature and a list server.

Security SDK

The SDK works on E4GL source applications on Webspeed, Blue Diamond, or other Webspeed alternatives that use E4GL coding. Just one more thing brewing up at Amduus Information Works!

Publishing Information:

Scott Auge publishes this document. I can be reached at scott_auge@yahoo.com.

Currently there are over 400 subscribers and companies that receive this mailing! This mailing is not sent unsolicited, so it is not SPAM.

Amduus Information Works, Inc. assists in the publication of this document:

Amduus Information Works, Inc.
925 South Wolfe Road #26
Sunnyvale, CA 94086-8843
408-730-4588
http://www.amduus.com

Article Submission Information:

Please submit your article in Microsoft Word format or as text. Please include a little bit about yourself for the About the Author paragraph.

Looking for technical articles, *marketing Progress* articles, articles about books relevant to programming/software industry, white papers, etc.

Following is my resume as well some Project Summaries of the work I have done.

I am looking for work, if you have any knowledge of potential work, I would appreciate hearing from you!

Thanks!

Scott Auge
Sunnyvale, California
E-Mail scott_auge@yahoo.com sauge@amduus.com

Skills Summary:

Experience with Internet/Intranet programming of E-Commerce and Customer Self-Serving web sites with C/C++/HTML/CGI/Javascript/Webspeed with NCSA/Apache/Netscape/IIS/Lotus Go Web Servers. Experience with manufacturing Job Costing/MRP II/Inventory/Logistics /Capacity Planning. Experience with service Industry Inventory/Service Delivery/Communications. Experience with Law Enforcement/Probation/Parole Systems. Experience with Wide Area Network programming and configuration (TCP/IP/X.25/Ethernet/FDDI) and Wide Area Packet Radio Wireless Networks (ARDIS).

Fourteen years of computer programming/analysis experience composed of RDBMS PROGRESS experience on various UNIX machines in PROGRESS versions 6, 7 and 8, in addition to Web development with PROGRESS Webspeed 2.1/C/C++ and systems level and application development in C and C++ on UNIX/MSDOS/Win95/MS NT 4.0 Workstation and Server.

Software Used:

HP-UX, AT&T UNIX, IBM AIX, NCR MP-RAS, SCO UNIXWare, SCO OpenDesktop, Sun Solaris, Windows NT Server/Workstation 4.0, Windows 3.1/Windows for Workgroups/Windows 95, Linux

PROGRESS 4GL (Versions 6,7,8,9) and PROGRESS Webspeed 2.1/3.1, HTML, Java, ksh/sh/bash shell scripting, IBM AIX C & C++, Borland C/C++, Zortech C/C++, SCO UNIX C Compiler, FORTRAN 77, Various BASIC, GNU C++ Compiler, HP-UX C Compiler, NCR High Performance C/C++ Compiler (on the AT&T UNIX and MP-RAS), MS Visual C/C++ Compiler, JYACC JPL, Borland C++ Builder, Powersoft C++

IBM Net.Commerce, IBM Net.Data, IBM DB2 CLI, DB2 Universal Database, Oracle Database, Progress Database, IBM MQ Series, MS Word, MS Excel, MS Access, Reflection X Suite (X Server and Terminal Emulation), Visio, Netscape, etc.

Relevant Career:

2001-Current
Amduus Information Works, Inc.

925 South Wolfe Road #26
Sunnyvale, CA 94086-8843

Founder of the company. Web based customer interaction software for the service, manufacturing, and law enforcement industries.

1995 - 2001
Programmer/Analyst/Systems Administration
Analysts Express (Software Contracting Company)
12902 Glenyork Court, Cypress, Texas 77429

Design/Analysis of various web based projects within Siemens Information and Communications Networks. Online support for customers to contact Siemens thru the web for service requests, E-Commerce for customers to purchase thru the web, Customer Dispute Resolution for invoices under dispute. Legal Contract inventory and lookup under a web interface.

Developed for Consulting Services of Progress Software Corporation for BI Incorporated. Performed installation of PROGRESS RDBMS 8.2B/Webspeed 2.1, analysis, and programming of national probationer tracking system.

Worked as a designer, implementer, and trainer on Astea Intl.'s Remote-1 installation, troubleshooting, and use with NCR's worldwide dispatching system. Enhancements and integration of Astea's Dispatch-1 software with NCR's worldwide dispatching system.

1993 - 1995
Programmer/Analyst
Coe Press Equipment Corporation (Industrial Equipment Company)
40549 Brentwood, Sterling Heights, MI 48310

Worked as a designer, implemented, and trainer of job costing, material resource planning, capacity resource planning, and logistics software. Developed in PROGRESS and C on UNIX.

1985 - 1993
Systems Consultant
Self-Employed (Software Contracting Company)

Worked with law enforcement agencies to better their informational structure. Provided training on computer operation.

Developed automated circulation tracking system for Daily Mining Gazette, a subsidiary of Thompson Newspapers. Trained/Documented/Enhanced the system. Worked with networked MSDOS developing with Q&A Database product. Various privately developed programs.

1989
FORTRAN Tutor
Gogebic Community College (Education)
E-4946 Jackson Road, Ironwood, MI 49938

Worked with students in the application of FORTRAN to solve problems. Training of computer use.

Education:

Gogebic Community College
Ironwood, Michigan

Studies for transfer to a four-year university.

Michigan Technological University
Houghton, Michigan

Studies leading to a BS degree in Computational Mathematics.

Associations:

Computer Society IEEE
Association of Computing Machinery (ACM)
International Who's Who of Information Technology
Michigan UNIX User's Group

Project Summary

Parts Order System
Coe Press Equipment

Departments involved:

Parts Orders - processing of customer orders for customer’s equipment
Shipping/Receiving - pulling parts from inventory and shipping by shipping service
Material’s Management - controlling inventory
Job Costing - costing of the parts order
Accounting - handling accounts receivable and invoicing

Problem Description:

Various departments having trouble communicating with each other. Parts Order department was required to walk a part order through production to see it was processed.

Parts Order entry was done through software tools designed to handle processing and design of machines and lines.

Manual record keeping system used making it difficult to share information (customers could call two different people and get two different quotes), research on previous orders and machines that required specially tailored parts was time consuming, reporting types of parts, etc. nearly impossible.

No backorder shipments could be handled. If parts needed to be made, entire shipment would be held until all parts were available to shipping department.

Any member of Parts Department could not keep a customer apprised of their order. Highly dependent on a single individual to know what was happening with an order.

Credit checking of customer required a call back to customer after entry person checked customer with accounting department.

Solution Description:

Parts Order Entry Program specifically tailored to the parts order entry operators’ needs to perform entry while on the phone with a customer.

Shipping Entry Program tailored to allow ease of use, prepare shipping documents, capture shipping and handling costs, and prepare records for the tracking of back-orders.

Routines to integrate costing of parts orders into existing costing system.

Routines to integrate parts orders inventory requirements into existing inventory control system.

Automatic Invoicing System, which determines shipping of parts and prepares invoice automatically through a laser printer.

Project Summary

Newspaper Circulation System
Daily Mining Gazette

Departments involved:

Circulation Department - tracking of subscribers, dealers, carriers, and boxes.
Press Department - printing of the newspaper
Mail Room - delivery of paper bundles to postal, carrier, and motor carrier delivery systems

Problem Description:

Totally manual system. Position totally devoted to hand writing of bills and tracking subscriber payments on payment cards (that were literally disintegrating). Carrier, Dealer, and Box counts were kept on large poster board tables.

Manual calculation of all reports, daily printing report, ABC Auditing report.

Difficulties determining a customer's payment record, carrier's payment records, how many papers should be distributed among carriers, etc.

Overall stress in the department due to the workload was taking its toll in high turnover and personality conflicts. Some people would get in at 5:00 am and stay till 6:00 PM.

Easy communication of information between various departments and preparation

Solution Description:

Entry programs for carrier, dealer, and subscribers. Identified various information about the entity/individual conducive to efficient and speedy processing in system.

Various programs to handle monthly accounting transfers and reports.

Automatic preparation of bundle identification papers

Automatic printing of subscriber bills

Moved postal label printing database from specialized computer to integrated database

Training of users in computer operation, program operation and wrote manual for system.

Discovered many subscribers who should not have been receiving the paper. Was able to eliminate one position and take another to part-time.

Project Summary

Police Information System
Laurium Police Department

Departments involved:

Police Department - Law enforcement agency for village government

Problem Description:

Completely manual system for incident report typing, gun ownership records, chain of custody, budgeting and reporting

Required an extremely inexpensive system.

Officer’s lack of training on computer systems.

Solution Description:

Training of Chief of Police and other key personnel on the operation and use of a computer system.

Implementation of off the shelf word processing and spreadsheet software.

Text based database keying off keywords in the word processing document.

Project Summary

Implementation of demand independent MRP to demand dependent MRP system
Coe Press Equipment

Departments involved:

Purchasing, Assembly, Parts Orders, Service Orders, Machining, Fabrication, Shipping/Receiving, Engineering, Management

Problem Description:

Previous inventory system was demand independent using minimum stocking levels for 32,000 parts maintained by people. This resulted in an increase in inventory overhead and prevented continued expansion of the company.

Previous system did not in great enough detail; define the operations required on items in the fabricating, machining, and assembly process. Many items where processed ad-hoc when a critical part was found missing or unavailable (hence upping the inventory count for that part in the above mentioned problem.) This caused trouble with scheduling of work.

Solution Description:

A new operations manager was brought on board. Conversations began with this operations manager and the requirements of an information system for the new processes and procedures he was hoping to introduce into the company.

These processes were designed during the use of the previous system. The previous system was taken off-line (excluding accounting features) as manual records and paperwork controlled the new processes. This was for approximately thirty days. New software was written during this time.

New demand dependent system went on line controlling creation of inventory and allocation of inventory across machines and assembly. Information was recorded in a database and "factory order" paperwork was implemented as a control document.

Various changes in the engineering, purchasing, and receiving department software were introduced. Additional software for a newly created data management group was developed as well control documents and software for the movement of parts from plant to plant, work cell to work cell, and plant to vendor supplied outside process to plant.

Project Summary

Integration of Customer Access to Astea's D1

Companies Involved:

NCR Australia Division, Astea Intl. NCR Enhancements Group, NCR United States Division

Problem Description:

To better support for NCR customers, NCR Australia wished to distribute software to customers that would allow them to directly add and monitor the status of their work orders on the dispatching system. (This would reduce the need for telephone support to begin getting the solution to the customer.)

Solution Description:

It was determined that NCR would be responsible for the client portion of the software that would be distributed amongst their customers. This client would be MS Windows based using Powerbuilder developed by NCR.

Astea (of which I was a contractor with at the time) would develop the server portion of the software. This server would be AT&T UNIX (NCR's MP-RAS flavor) based.

A method of login to a UNIX system for the client program was devised using the UNIX I/O subsystem (/dev/tty) into a Progress Session (shared memory client) using a special terminal emulation to eliminate unneeded data transfer between the client and the server. This allowed a non-progress based client (actually PowerBuilder 4.) to communicate with a Progress database (via a modem and telnet) with normal 4GL programming on the Progress side.

A protocol was developed between the server and client of activity, incoming fields, delimiters, and end of record information. This allowed definite limitations to the amount of data passed between the client and the server and better control over response time.

This protocol was used between the Customer Access client program and an API Server that resides on the UNIX computer. The client would issue a request for some type of activity to be performed (an inquiry or update) which the API server would recognize, and route the data to the appropriate API on the D1 application to perform the service.

Wrote APIs to D1 for such things as work order inquiries, equipment inquiries, searching for equipment, updating work orders, etc.

Optimized the API Server program.

Project Summary

Integration of Remote-1 into NCR Customized Dispatching System

Companies Involved:

Astea Intl., NCR United States Division

Problem Description:

NCR has approximately 3800 Field Engineers to satisfy their customer base. NCR went with a customized version of Astea Intl's Dispatch-1 (D1) to replace the legacy dispatch system that was overburdening seven mainframe computers. This new dispatching system would need direct contact to not only the call takers and customer care agents, but also the field engineers who were in virtual offices though out the U.S. and the world.

Solution Description:

Astea provided a remote contact system called Remote-1 (R1). R1 provided the ability of the field engineers to interact with the D1 software as if they were in the office - allowing all kinds of activities from opening and closing work orders, to posting labor, expenses, and parts. Most important was the ability of R1 to interact with the contracting system of D1, alerting the remote users of work orders that needed to be prioritized to meet the expectations of the customers.

But this product was not available to the version of Dispatch-1 NCR was using. I was part of the team of three (with phone assistance with the actual developers of the original system) that connected R1 to the customized version of D1 NCR was running.

I was hired as a contractor to Astea Intl. and very much the end-to-end person regarding R1, familiar with the UNIX software (and code), wide area networking (implementing wireless and X.25 connectivity), and client software (and code).

Since NCR was running a customized version of D1, R1 also needed to be customized. I performed the following functions:

- Initial analysis and programming regarding the integration of R1 to NCR's D1 on the UNIX server(s).
- Building and training for Front End Processors for the Wide Area Network

- Problem Solving regarding the Wide Area Network (both wireless and wired)
- Analysis and programming regarding the client software.
- Served as a source of information for later programmers who began customizations to R1 in the next phase of customizations for D1.

This included flying to NCR headquarters in Dayton OH for installation and troubleshooting.

Project Summary

Internet Site

Companies Involved:

KodeNet, The InterNIC, Network Solutions

Problem Description:

I also wished to establish a presence and reputation as a capable consultant, and to provide a centralized point where information regarding my projects and skills could be easily located. In addition, I needed a machine on the internet where files between internet users (usually myself and other clients) could be placed and to handle E-Mail for me no matter where I was.

Solution Description:

Designed and built the hardware (off the shelf components) for a UNIX based server.

Installed UNIX and X Windows (X11R6) onto the hardware.

Performed Vendor Negotiations with an ISP (Internet Service Provider) for a dedicated link to the internet without any restrictions. Registered the domain sauge.com with the InterNIC (Network Solutions.)

Networked the system into the internet via a PPP connection using TCP/IP (routing and DNS), setup FTP, TELNET, Finger, sendmail and POP3 (used for E-Mail) daemons for internet use. Developed a means by which a dropped connection to the internet would be realized by the UNIX system, and it would begin reconnection steps.

Networked the UNIX server computer with my Windows 95 laptop in addition to the internet over a 10baseT network. Used Reflections X Suite on the laptop, which provides a TCP/IP stack with FTP, TELNET, Web Browser, etc. clients in addition to utilities for problem resolution. Set up a shared drive on the UNIX machine that allows me to map a Windows drive to the UNIX CDROM and hard drives via NFS.

Installed a web server from National Center for Supercomputer Applications (the NCSA web server). Developed web pages using animated art, diagrams, links to FTP sites and other web sites.

Wrote CGI interfacing using the CGIC library by Thomas Boutell. This is library of routines for handling GET, POST, and other HTTP commands from within a C program for Check Boxes, Numeric Input, Selection Menus, etc. Wrote CGI connections for Progress Installation Survey and other internal pages on the system.

Project Summary

Stock Alert - Online Stock Market Analysis

Companies Involved:

NASDAQ Web Site, NYSE Web Site
(This tool is for my own investment strategies as well those who wanted access to the system)

Problem Description:

Software solution to automate the analysis of stock data in the NASDAQ stock market. Needed a tool that acquires data from the market, applies rules to the data for the strategies involved, and then E-Mail's alerts (stock rising, stock falling, current value, and strategy alerts) to subscribers of the system about market values of interest to them.

Solution Description:

Developed software to obtain stock data from NASDAQ's web site. This software was written in C++ and involved socket programming. Incidentals were scripted using the Korn Shell to prepare files that would be loaded into the Progress database through IMPORT statements.

This data was then placed into a database managed by Progress Software's RDBMS. An easy to use front end was developed for analyzing the stock interactively, as well batch based programs that would analyze the data with programmed strategies. The batch processes provided the means of creating the afore mentioned alerts, via interfacing with the UNIX E-Mail system.

The strategies were based on Technical Analysis of the stock market data.

After receiving alerts, trades by subscribers were then done by their own accounts (usually web based such as www.eschwab.com, etc. but some had regular brokers) on their own terms.

Project Summary

Probationer Tracking System

Companies Involved:

Progress Software Corp. Professional Services, BI Inc., Community Corrections Corporation, and Peregrine Corporation

Problem Description:

BI Inc. bought out Community Corrections Corporation and Peregrine Corporation. BI Inc. then wanted to implement a common information system between it and the new offices it had acquired.

Solution Description:

It was determined that removal of the existing systems in the company was needed. CCC's was MSDOS based and on an ancient database platform that required a great deal of manual sorting and cross checking. Peregrine's system was based on MS Access and would not support the additional burdens placed on it.

Installation of UNIXWare as the database server and report generating operating system with Windows NT as the print server. This allows for a scalable network operating system specifically designed for multiple users that were open and integrated with other systems as well acts as the standard operating system of most World Wide Web transactional web sites.

The use of Webspeed to provide a WWW interface to the application as some offices would be dialed in over low bandwidth telephone lines, and to allow easy implementation of the system into 50+ offices. This allowed users to access the system easily and with little configuration using Netscape Navigator and Microsoft's Internet Explorer client software.

This allowed an infrastructure to be set-up for network computers, or X-terminals, or PCs - whatever the future structure of computing would be regarding user devices, as well a means of user interface that required very little training to operate and maintain.

Use of Progress Software's Webspeed development tool allowed fast efficient integration of the web interface with the database.

The application allowed BI to provide drug testing services, education and counseling services, restitution collection as well court cost and surcharge collections, the automatic issuing of warrants among other legal documents, as well report information for various courts throughout the United States.

Project Summary

Customer On-Line Support

Companies Involved:

Siemens Information and Communications Networks (SICN), Siemens Business Systems (SBS)

Problem Description:

SICN wanted to reduce customer calls into their company for information about the status of work orders. In addition, they wanted to open up to the customer pieces of their information system, but not parts that should remain internal to Siemens.

Solution Description:

A WWW application was created that users could log into that interacted with the legacy Dispatch-1 application. Accounts and passwords were implemented to prevent any unauthorized access to the application. Features included not only looking up the status of current orders and work orders, but also the creation of help calls that would allow the customer to be in contact with Siemens without having to call and possibly be placed on hold at the support number.

The goal for Siemens was to make the customer as self-serving and in control as possible, as well reduce 800 call expenses.

Sets of AIX RS/6000 computers were purchased to support the On-Line Support system as well another called Electronic Commerce, which was integrated into the site. The Progress RDBMS was used for On-Line Support and for the D1 application.

One RS/6000 was setup outside a firewall and equipped with a Progress Webspeed messenger for interaction with a Webspeed Transaction Server on another computer within the firewall. The transaction server computer in on an RS/6000 running AIX that interacted with a Progress database server on a third RS/6000 computer.

The application was then integrated with additional applications via IBM's MQ Series product used by Siemens for order taking and review, as well information about products available thru Siemens.

Project Summary

Electronic Catalog

Companies Involved:

Siemens Information and Communications Networks (SICN), Siemens Business Systems (SBS), IBM Global Services

Problem Description:

SICN wished to have an ability to service (via MAC) their customers of telephone switching and PBX equipment over the World Wide Web. This would ease the need of phone call takers, and reduce the cost of having manual systems between the customer and the company's information systems. SICN also felt it important as a competitive advantage to differentiate from other companies in its business and catch up to other companies in the industry.

In addition, changes would need to be made to the electronic catalog that allowed creation of work orders (as opposed to simple product shipping) that required service personnel to come to the site for installation or skilled maintenance on the equipment for the additional product to be added.

Solution Description:

The electronic catalog was implemented using a product called Net.Commerce available from IBM. The application would need to be modified to support "special pricing" that was available to each customer, as well as identifying which products have required additional products. It would need to interface with D1 for products that required a service technician for installation.

The service dispatching application was the already available as D1 from Astea Intl. Creating data in the tables by integration applications would be required with this application, as it does not lend it's self to easy integration with other systems.

An inter-operating system network based messaging system was chosen from IBM called MQ Series. This allowed real time transfer of data based on events in one application to effect another application on another OS or machine. This provided the basic infrastructure of allowing the web-based applications to interact with other applications in other languages.

Examples of messages were special prices for given customers and creation of MAC orders for equipment ordered thru the web into the service application, which would then feed into the manufacturing and order processing applications.

Net.Commerce was then integrated into a web page of applications available to the customer for use.

Net.Commerce, DB2, Progress RDBMS, Progress WebSpeed 2.1 and IBM's MQ Series were used. All software (excluding the browsers) ran with AIX on IBM RS/6000 machines.

Languages included Net.Data, Korn shell scripting, C, C++, Progress 4GL, Javascript and HTML.

Project Summary

Customer Dispute Resolution

Companies Involved:

Siemens Information and Communications Networks (SICN), Siemens Business Systems (SBS), INFOSys Solutions

Problem Description:

SICN Accounts Receivable department wished to update and change the processing of invoice disputes received from customers. In addition, a means was desired to match the corporation's goal of using thin client technology to reduce maintenance costs of applications and to enable a new computer to become functional out of the box (via a web browser).

Solution Description:

A WWW interface was agreed upon as the means of user interaction to the application. In addition, the application would heavily use E-Mail to inform users of activities requiring their attention. Since web address can be embedded in E-Mails, as well as reports, Email was determined to be a strong component of the application.

A set of APIs were created that would work with ideas of future WWW interfaces that would allow customers to directly input the invoices under question from an external web site, as well as allow easy integration of the application with other systems in the company (SAP R/3).

The application allowed entry of disputes, automatic routing of disputes to certain people in the organization that had the authority to solve the problem, resolution of the dispute, escalation of disputes that "fell by the way" to the dispute owner and others, data transfer between UNIX and VMS systems, generation of E-mails from the UNIX email subsystem, as well various reports about disputes and collections of disputes for areas in the country.

Progress RDBMS and Webspeed on RS/6000-AIX were used to store and manipulate the data. Netscape Navigator and Microsoft Internet Explorer are the thin client interfaces to the application.

Languages used: HTML, Javascript, Korn shell scripting, and Progress Webspeed 2.1 were used to implement the application. Protocols include SMTP, as well MIME 1.0 and mail attachments (UUENCODE).

Project Summary

Contracts Online

Companies Involved:

Siemens Information and Communications Networks (SICN), Siemens Business Systems (SBS)

Problem Description:

Siemens ICN provides service for the telephone and data needs of over 38,000 customers throughout the US. These service offers are controlled by contracts with the customer about what will be performed, as well the cost/warranties/etc associated with the location and equipment.

In order to handle customer questions about how problems are addressed, a need arose for multiple Siemens internal employees located across the United States to access these contracts.

Contract information included a wide variety of standard legal documents such as Terms & Conditions, Schedule A’s, as well as miscellaneous documents that were developed in deals such as adjunct memos and supporting documents of agreements.

Solution Description:

The Contracts Online program was developed for use on Siemens intranet web sites that would act as an index and search mechanism to an Adobe PDF image of the contract. Composed of 76 screens and 80 source code files, the application allowed quick and easy identification of contracts covering a site, multiple sites, single pieces of equipment, multiple pieces of equipment, as well as buying consortiums composed of multiple customers.

Oracle RDBMS, Progress WebSpeed 2.1 and Oracle Data Server on RS/6000-AIX was used to store and manipulate the data. Netscape Navigator and Microsoft Internet Explorer were used as the thin client interface.

The application use of HTML, Javascript, Korn shell scripting, and Progress 4GL was used to implement the application.