



Using Old Technology for New Problems

Alan Heter and
Alan Wyman
U.S. Foodservice - Eagan

HP e3000 Solutions Symposium
April 2002

Abstract:

When you need to interface with an existing application, old tools can be useful. An intranet client-server or application-server relationship can be written across multiple platforms using FTP, COBOL, and batch jobs on the HP e3000. This session will include a real-life example using the FTP Site Stream command, batch jobs, COBOL, and Byte Stream files. The example will present an application server between an HP 9000 and an HP e3000. The session will cover mostly the HP e3000 portion of the application.

Biographical Sketch:

Alan Heter, after a brief stint with an IBM 360 (six months), has spent his entire career programming and performing system maintenance on HP 3000 equipment. He has been involved in education, manufacturing, and distribution.

Alan Wyman has been working on the HP 3000 since the Series II in the middle '70s and has been involved in health care, manufacturing, publishing, and distribution.

How did we get there?

- Corporate required connection to Dot Com initiative
 - HTTP Post
 - Java and Unix
- Developed in PerCOBOL
 - Write in COBOL
 - Compile into Java

HP e3000 Solutions Symposium
April 2002

Corporate had already made decisions to base the interface on HTTP Post and we would use Java to create these Posts.

We did not have Java expertise on-site. We are a HP e3000 site running the usual COBOL, TurboImage, and VPlus.

Problems brewing

- Performance was not up to necessary level using Java
- New machine was needed
 - Even before this project
 - Capital requests were frozen
- Left us looking for alternatives

HP e3000 Solutions Symposium
April 2002

Currently running on a 987/150

By the time that the project was up our company was being purchased and there was a freeze on all capital requests.

Went back to basics

- FTP existed on both platforms
- PerCOBOL was development language
 - Threw out the Java
 - Recompiled in HP COBOL
- File formats were defined to mimic the HTTP Post transactions

FTP

- Filename had to be determined
 - Corporate logon to use the root part of the filename as the session name

R0079418,FTP.PRD

- Makes the filename identifiable

/PRD/PUB/dotcom/R0079418_in

HP e3000 Solutions Symposium
April 2002

The root part of the filename has to follow the MPE rules for a filename.

- Start with a letter
- No special characters
- All uppercase
- No more than 8 characters in length

This allows the root to be used as a session identifier

FTP

- Had to process the file
 - Use the Site command to Stream a job.

```
quote site stream aljftp01
```

- This initiates a job on the HP e3000 to process the inbound file and create the outbound file.

HP e3000 Solutions Symposium
April 2002

Here is an example using a PC in place of the corporate Unix server.

```
C:\WINDOWS>ftp
ftp>OPEN SEVENOF9
Connected to SEVENOF9.
220 HP ARPA FTP Server [A0010H08] (C) Hewlett-Packard Co. 2000 [PASV SUPPORT]
User (SEVENOF9:(none)): T0028524,DOTCOM/*****.PRD
230 User logged on

ftp> put T0028524 /PRD/DOTCOM/alink/in/T0028524
200 PORT command ok.
150 File: /PRD/DOTCOM/alink/in/T0028524 opened; data connection will be
opened
226 Transfer complete. Some records were truncated during transfer.
ftp: 91 bytes sent in 0.00Seconds 91000.00Kbytes/sec.

ftp> quote site stream aljftp01
200 STREAM command ok.
```

FTP

- Had to process the file
 - The initiating session name is passed as the first part of the HPSTREAMEDBY variable.

```
:SETVAR fileroot
  LFT('!HPSTREAMEDBY',POS(',','!HPSTREAMEDBY')-1)
:SETVAR filerootx '!fileroot'+'.tmp'
:SHOWVAR fileroot
FILEROOT = R0079418
```

HP e3000 Solutions Symposium
April 2002

This is the first part of the jobstream executed by the remote process.

```
:SETVAR fileroot LFT('!HPSTREAMEDBY',POS(',','!HPSTREAMEDBY')-1)
:SETVAR filerootx '!fileroot'+'.tmp'
:SHOWVAR fileroot
FILEROOT = R0079418
:FILE aldcin= /PRD/DOTCOM/alink/in/!fileroot
:FILE aldcout= /PRD/DOTCOM/alink/out/!filerootx
:IF FINFO('/PRD/DOTCOM/alink/out/!fileroot','EXISTS') THEN
  *** EXPRESSION FALSE: COMMANDS IGNORED UNTIL MATCHING ELSEIF/ELSE/ENDIF
:  PURGE /PRD/DOTCOM/alink/out/!fileroot
:ENDIF
  *** RESUME EXECUTION OF COMMANDS
:RUN ALDC01OB.CODE.PRD
*****
Begin ALSJ01 read WEB on 03/08/02 at 06:40:47
Record size read = +000000063
```

Needed to hide the file

- While the file was processing we needed to hide it from the requestor
- Created the output file with a .tmp extension
- Renamed after the processing was completed.

```
:RENAME /PRD/DOTCOM/alink/out/!filerootx,  
/PRD/DOTCOM/alink/out/!fileroot
```

HP e3000 Solutions Symposium
April 2002

The end of the jobstream handles the 'hide the file' routine

```
End ALSJ01 for LOGIN-ID 2004      on 03/08/02 at 06:40:48  
*****  
END OF PROGRAM  
:RENAME /PRD/DOTCOM/alink/out/!filerootx, /PRD/DOTCOM/alink/out/!fileroot  
:EOJ
```

The Unix server attempts gets until successful with a pause before and between to allow the job to run on the HP e3000.

```
ftp> get /PRD/DOTCOM/alink/out/R0079418  
200 PORT command ok.  
150 File: /PRD/DOTCOM/alink/out/R0079418 opened; data connection will be  
opened  
226 Transfer complete.  
ftp: 56475 bytes received in 0.27Seconds 209.17Kbytes/sec.  
  
ftp> quit  
221 Server is closing command connection
```


Opening the bytestream file

- The file was sent from the Unix machine as bytestream
- The file is opened using HPFOPEN rather than a standard COBOL Open statement
- A large buffer is set aside for the inbound transaction

HP e3000 Solutions Symposium
April 2002

```
01  SERVLET-RECORD                                VALUE SPACES.
    05  SERVLET-RECORD-A                          PIC X(18824).
01  WS-FILE-NAME-IN.
    05  FILLER                                    PIC X      VALUE "".
    05  WS-FILE-NAME-ACT                          PIC X(06)
           VALUE "ALDCIN".
    05  FILLER                                    PIC X      VALUE "".
01  WS-FILE-NAME-OUT.
    05  FILLER                                    PIC X      VALUE "".
    05  WS-FILE-NAME-ACT-OUT                      PIC X(07)
           VALUE "ALDCOUT".
    05  FILLER                                    PIC X      VALUE "".
```

Opening the bytestream file for input

- The HPFOPEN parameters are defined in Working Storage
- Include the expected length and status return identifiers

HP e3000 Solutions Symposium
April 2002

```
01  HWB-INTERNAL.
    05  HWB-INTERNAL-ITEMS.
        10  HWB-INT-CONST-0      PIC S9(09) COMP SYNC VALUE 0.
        10  HWB-INT-CONST-1      PIC S9(09) COMP SYNC VALUE 1.
        10  HWB-INT-CONST-2      PIC S9(09) COMP SYNC VALUE 2.
        10  HWB-INT-CONST-3      PIC S9(09) COMP SYNC VALUE 3.
        10  HWB-INT-CONST-4      PIC S9(09) COMP SYNC VALUE 4.
        10  HWB-INT-CONST-9      PIC S9(09) COMP SYNC VALUE 9.
        10  HWB-INT-CONST-10     PIC S9(09) COMP SYNC VALUE 10.
        10  HWB-INT-FILE-NAME    PIC X(256) .
        10  HWB-INT-FNUM-D       PIC S9(09) COMP SYNC .
        10  FILLER                REDEFINES HWB-INT-FNUM-D.
            15  FILLER              PIC X(02) .
            15  HWB-INT-FNUM        PIC S9(04) COMP .
        10  HWB-INT-PTR          PIC S9(09) COMP SYNC .
        10  HWB-INT-STATUS       PIC S9(09) COMP SYNC .
        10  FILLER                REDEFINES HWB-INT-STATUS.
            15  HWB-INT-STATUS-INFO  PIC S9(04) COMP .
            15  HWB-INT-STATUS-SUBSYS PIC S9(04) COMP .
        10  HWB-INT-DISP         PIC S9(09) COMP SYNC VALUE 0.
    05  HWB-REC-SIZE             PIC S9(09) COMP VALUE -18824.
    05  HWB-ACT-SIZE             PIC S9(09) COMP VALUE 0.
    05  HWB-ERROR                PIC S9(04) COMP VALUE 0.
```

Opening the bytestream file for input

- Include the file number and status return identifiers and a list of other parameters
 - 2 – Filename
 - 3 – Domain – value of 3 = Old perm or temp
 - 6 – Record format – value of 9 = Bytestream
 - 11 – Access – value of 0 = Read
 - 77 – Data format – value of 2 = Access file as native bytestream

HP e3000 Solutions Symposium
April 2002

```
0100-OPEN-INPUT SECTION.
```

```
CALL INTRINSIC "HPFOPEN" USING HWB-INT-FNUM-D,  
                                HWB-INT-STATUS,  
                                2, HWB-INT-FILE-NAME,  
                                3, HWB-INT-CONST-3,  
                                6, HWB-INT-CONST-9,  
                                11, HWB-INT-CONST-0,  
                                77, HWB-INT-CONST-2,  
                                0.
```

```
IF HWB-INT-STATUS <> 0
```

```
    DISPLAY 'HPFOPEN FAILED' UPON SYSOUT
```

```
* Mode 2 call will display error msg on $STDLIST
```

```
CALL INTRINSIC "HPERRMSG" USING 2,  
                                \\  
                                \\  
                                HWB-INT-STATUS
```

```
    STOP RUN
```

```
END-IF.
```

Reading the bytestream file

- The read is a standard FREAD intrinsic.
- The expected length is sent and the actual length is returned

HP e3000 Solutions Symposium
April 2002

```
0200-READ-INPUT SECTION.
```

```
CALL INTRINSIC "FREAD" USING HWB-INT-FNUM,  
                             SERVLET-RECORD,  
                             HWB-REC-SIZE  
                             GIVING HWB-ACT-SIZE.  
DISPLAY "Record size read = " HWB-ACT-SIZE.  
CALL INTRINSIC "FCHECK" USING HWB-INT-FNUM  
                             HWB-ERROR.  
  
IF HWB-ERROR = 0  
    CONTINUE  
ELSE  
    DISPLAY "File read error = " HWB-ERROR UPON SYSOUT  
    STOP RUN  
END-IF.
```

Closing the bytestream file

- The close is a standard FCLOSE intrinsic.

HP e3000 Solutions Symposium
April 2002

```
0300-CLOSE-INPUT SECTION.
```

```
CALL INTRINSIC "FCLOSE" USING HWB-INT-FNUM,  
                                HWB-INT-DISP,  
                                0.
```

Opening the bytestream file for output

- Parameter list is longer
 - 2 – Filename
 - 3 – Domain – value of 4 = New perm
 - 5 – Designator – 0 = Other options specify filename
 - 6 – Record format – value of 9 = Bytestream
 - 7 – CCTL – 0 = No carriage-control
 - 10 – File Type – 0 - Standard
 - 11 – Access – value of 1 = Write
 - 13 – Exclusive – value of 1 = Exclusive access
 - 50 – Final disposition – value of 0 = No change

HP e3000 Solutions Symposium
April 2002

```
0400-OPEN-OUTPUT SECTION.
```

```
CALL INTRINSIC "HPFOPEN" USING HWB-INT-FNUM-D,  
                                HWB-INT-STATUS,  
                                2, HWB-INT-FILE-NAME,  
                                3, HWB-INT-CONST-4,  
                                5, HWB-INT-CONST-0,  
                                6, HWB-INT-CONST-9,  
                                7, HWB-INT-CONST-0,  
                                10, HWB-INT-CONST-0,  
                                11, HWB-INT-CONST-1,  
                                13, HWB-INT-CONST-1,  
                                50, HWB-INT-CONST-0,  
                                0.  
  
IF HWB-INT-STATUS <> 0  
    CALL INTRINSIC "HPERRMSG" USING 2,  
                                    \\  
                                    \\  
                                    HWB-INT-STATUS  
  
    STOP RUN  
END-IF.
```

Writing to the bytestream file

- The write is a standard FWRITE intrinsic.
- The actual length is sent
- The close is also standard FCLOSE

HP e3000 Solutions Symposium
April 2002

```
0500-WRITE-OUTPUT SECTION.
```

```
CALL INTRINSIC "FWRITE" USING HWB-INT-FNUM,  
                               SERVLET-RECORD,  
                               HWB-REC-SIZE,  
                               HWB-INT-STATUS.
```

```
IF HWB-INT-STATUS <> 0
```

```
    CALL INTRINSIC "HPERRMSG" USING  2,  
                                     \\  
                                     \\  
                                     HWB-INT-STATUS
```

```
    STOP RUN
```

```
END-IF.
```

```
0600-CLOSE-OUTPUT SECTION.
```

```
CALL INTRINSIC "FCLOSE" USING HWB-INT-FNUM,  
                               HWB-INT-DISP,  
                               0.
```

Data extracts were also needed

- Corporate wanted to keep a customer and product list on their system as well as sales history
- Originally thought they needed real-time access
- Set up a database to duplicate our data in their format

Real-time was dropped

- Already having performance issues on our box and on theirs
- Most of the data was actually not as dynamic as corporate thought
 - Sales history is once a day
 - Customer and product updates can be daily
 - Pricing is regenerated weekly only

Intermediate database existed by this time

- Extracts and reformatting to the corporate format were done
 - Corporate format was stored in a TurboImage database
- ALMOST!
- TurboImage data formats are not Oracle formats
 - Date format was thought to be a problem

HP e3000 Solutions Symposium
April 2002

Corporate was specifying a date format of ddMMMyyyy

For example 04APR2002

They already had an intermediate database on their side that they would translate this format to Oracle date format.

Extract process

- Easy to extract using Suprtool
- Had to change some fields to odd-number of bytes

HP e3000 Solutions Symposium
April 2002

```
!RUN SUPRTOOL.PUB.ROBELLE
BASE ALDC,5,CREATOR
GET CUST-MASTER
DEFINE CUST_NBR,CUST-NBR,8,DISPLAY
DEFINE CUST_NAME,CUST-NAME,25,BYTE
DEFINE OG_PRT_PRC_IND,PRINT-SW,1,BYTE
DEFINE STATE,ST-NAME,2,BYTE
DEFINE ZIP,ZIP-CODE,9,DISPLAY
DEFINE LAST_ORDD_DATE,ORDR-LST-DATE,10,BYTE
EXTRACT CUST_NBR
EXTRACT CUST_NAME
EXTRACT OG_PRT_PRC_IND
EXTRACT STATE
EXTRACT ZIP
EXTRACT LAST_ORDD_DATE
OUTPUT EXTRFILE,LINK
EXIT
```

Needed method to import into the Oracle database

- Suprtool/UX was the answer
- Need to convert the self-describing file to 2 files for Unix
 - SDUNIX utility available from Robelle
 - Files must reside in the same directory
- FTP files to corporate
- Only 4 Suprtool commands needed to load the data

HP e3000 Solutions Symposium
April 2002

Must convert the self-describing file to 2 files and FTP them to the Unix machine. The .sd file must reside in the same directory as the data file.

```
!  
!SDUNIX;INFO="extrfile exsdfile NOLF"  
!  
!ECHO OPEN hp821npc > alfstcst  
!ECHO USER !extruser !extrpass >> alfstcst  
!ECHO BINARY >> alfstcst  
!ECHO PUT extrfile /var/opt/load/extrfile >> alfstcst  
!ECHO PUT exsdfile /var/opt/load/extrfile.sd >> alfstcst  
!ECHO EXIT >> alfstcst  
!FTP < alfstcst  
!  
!EOJ
```

Suprtool/UX commands needed to load the oracle database

```
OPEN oracle_database  
INPUT extrfile  
ADD table_name  
EXIT
```

Corporate now sending us data using a Suprtool extract

- A related project needs daily data from corporate
- The HPe3000 tool worked so well for them that they extract our data using it.

HP e3000 Solutions Symposium
April 2002

Not a plug for Suprtool, just a satisfied customer.