Help Desk Management System
System Administrator's Guide

User Profiles and Network Access

DRAFT VERSION

This section of the Administrator's Guide describes the procedures necessary to configure a user for access to HDMS (or any other Accell application); for the benfit of those who are doing the installation, we have included a sample .login. In addition, this section of the User's Guide will briefly describe the strategy used at Delmarva Power to allow users to physically access the application over the network.

Setting Up Users

Rather than tiring the reader with useless details about how to set up users for HDMS, a sample `.login` for an HDMS user will now be listed. [Note that the `.login` is a UNIX file for the C Shell, if you use other shells at your site (such as Bourne Shell or K Shell), you should make the appropriate changes to the syntax in the following listing.]

```
#
# .login    -- Commands executed only by a login C-shell
#

setenv SHELL /bin/csh
setenv SPOOLER "lp -s -d ncclaser -o b"
setenv DBPATH /usr/db/ACL4/db
setenv UNIFY /usr/db/ACL4/lib
setenv TERMCAP /usr/db/ACL4/lib/termcap
setenv AMGRINIT 'heapsize=65536 init_mem_size=92160'

set path = ($path $home/bin /usr/db/ACL4/bin .) # execution search path

set noglob
set term = (`tset -m ansi:ansi -m :\?vt220 -r -S -Q`)
if ( $status == 0 ) then
      setenv TERM "$term"
endif
unset term noglob
stty ek
clear
echo Current System Activity:
w
echo ""
cat .todo
echo ""
```

Below is a description of what actions the example `.login` performs.

(1) After the comment block, the first group of six statements sets "environment variables" that are used by ACCELL and associated programs to access both the data and the application. The SHELL variable specifies what shell should be used when the user "shells" to the operating system from within another program (*e.g.* `vi` or the ACCELL/Environment). The `SPOOLER` variable specifies which printer the user's HDMS reports should be sent to as well as what print options the user wants for those reports. The `DBPATH` variable specifies the directory path where the data dictionary, application data, B-Tree

indeces and other important information is located. The UNIFY variable specifies the path where the executables and data files for the ACCELL Integrated Development System application are located. The TERMCAP variable specifies the path where UNIFY keeps its own files for terminal functions such as line draw, screen clear, and reverse video. The AMGRINIT variable is used by the ACCELL/Manager to set the value of certain variables.

(2) The next statement sets the user's search path. This is a utility of the C Shell where a certain variable (path) holds a set of paths that the shell should search in for files. Notice that the . (which specifies the current directory) is the last entry in the path variable; the reasoning behind this is that files in the user's current directory should not conflict with user commands. In order for the user to be able to execute an ACCELL application, `/usr/db/ACL4/bin` should be listed in path before the ., as shown in the listing.

(3) The purpose of the lines starting at `set noglob` and ending at `unset term noglob` is to inform the operating system of the terminal type that is being used to log on to the machine. This is done so that the user will be able to clear the screen, get bold and/or underlined characters, and other terminal capabilities. Note that when the user is logged on, but not in an ACCELL application, he/she is using the operating system's TERMCAP definitions. Once the user enters an ACCELL application, he/she is using UNIFY TERMCAP definitions.

(4) The purpose of the line reading `stty ek` is to assign keystrokes to the erase and kill functions, namely BACKSPACE and <CTRL-U>.

(5) The remainder of the file was added by the user. Any shell command or executable can be called in the `.login`. In this example, the user clears the screen, prints the line, "Current System Activity" to the screen, executes the `w` program (`w` is short for what), prints a blank line, prints the contents of the file `.todo` (a To Do List), and then prints a blank line. After this line, the `.login` will be finished executing, and the shell will present the user with its prompt (`%`, for example). Other ideas for a `.login` would be assigning aliases, printing a fortune or joke, running the mail program, etc.

As a sidenote to those doing the installation of HDMS, if people unfamiliar with UNIX will be using an application, you may choose to have their `.login` start a menu with a few options. An example menu (once again from Delmarva Power) is shown below:

```
                UNIX SystemPro Application Menu

1.  HDMS          Execute the HelpDesk Management System.
2.  Tutorial      Execute ACCELL Tutorial (CTRL-C to exit).
3.  Passwd        Change your password. (Can only be done
                  once every 14 days.)
4.  Mail          Read your mail.
5.  Exit          Logout of UNIX.

If you experience problems or have questions regarding:
    UNIX or this menu, contact Blaine Boyles x5224
      HDMS or ACCELL contact John Stewart x5319
```

You may limit these users even more by having their `.login` actually start the application. Note that you cannot simply specify the application as their shell in `/etc/passwd` because the environment variables must be set.

Network Access

(The following discussion pertains specifically to data communications, though Delmarva Power does own and operate a corporate voice network.) HDMS was originally developed in the Network Operations Department of Delmarva Power, the regional electric and gas utility for the Delmarva Peninsula. Utilities, in general, can more economically build wide area networks because of our advantage when it comes to such things as right of way; the bottom line is that, for utilities, bandwidth is cheap. As a result, Delmarva began developing its own internetwork very early. The people present during the beginning and early stages of the internetwork had the foresight to see that the management of such a network (both technical and administrative) would require massive amounts of time. In order to reduce this time as much as possible, an organizational tool was necessary. This tool should include Trouble Ticket Printing, Project/Task Organization, communication gateways for electronic mail, real-time statistics monitoring, and much more. This, in a nut shell, was the origin of HDMS. (As an aside, HDMS is the first stage of a larger project called Delmarva Integrated Alarm System, or DIALS for short. DIALS will assist in the management of the local and wide area networks by providing a consistent display of network alarm conditions and after-hours notification of critical alarms via voice synthesis. Work on the later stages of DIALS is going on as of this writing.)

Now that the reader understands what prompted the development of HDMS, we will go on to discuss the strategy used for allowing users to reach the machine where HDMS is located over the network. First of all, the machine that HDMS is currently being run on at Delmarva Power is a Compaq SystemPro running UNIX System V from the Santa Cruz Operation (SCO). The SystemPro itself is configured with two 80486 33MHz CPUs, 28 Megs of RAM and 1.2 Gigabytes of total disk space. We recently purchased a Dell 433TE as a backup for the SystemPro; it runs the same operating system, but is configured with one 80486 33 MHz CPU, less disk space and less RAM. The Dell machine will be installed at the alternate Network Operations Center (ANOC). In the middle of the night, the Dell machine will automatically connect to the SystemPro and transfer the HDMS database via File Transfer Program (FTP). The implementation of this will be much easier (and will eat up much less bandwidth) with the addition of the Print Journal utility (prjrn) in UNIFY 2000. The primary NOC is located at the Corporate Data Center where there are five ethernet sub-networks as well as a section of the multi-protocol (though predominantly IP) internetwork. The SystemPro sits on an Ethernet segment along with two LAN file servers running Banyan VINES, and many workstations.

All Network Operations personnel use 386 PC's with VINES and Microsoft Windows 3.0. We use FTP Software Inc.'s PC/TCP product that utilizes concurrent protocol stacks of TCP/IP and VINES IP (they have generic implementations as well as implementations specific to many other Network Operating Systems). This way, any PC running PC/TCP (either from a local or network drive), can use the Telnet protocol to access HDMS.

We also allow users sitting at a VT220 type terminal to access the SystemPro through our Equinox (a port selector) and a Corollary 8x4 Terminal Concentrator. This is provided for the convenience of dial-up users.