



AT&T

999-300-534
(For use with UNIX®
System V/386 Release 3.2)

Simul-Task 386

Version 2.0

User's Guide

©1988 AT&T
©1984, 1985, 1986, 1987, 1988 Phoenix Technologies Limited and
Interactive Systems Corporation
All Rights Reserved
Printed in USA

NOTICE

The information in this document is subject to change without notice. AT&T assumes no responsibility for any errors that may appear in this document.

ASHTON-TATE and dBASE III are registered trademarks of Ashton-Tate, Inc.
IBM is a registered trademark of International Business Machines Corporation.
Intel is a registered trademark of Intel Corporation.
LOTUS and LOTUS 1-2-3 are registered trademarks of Lotus
Development Corporation.
LOGITECH is a registered trademark of Logitech Inc.
Microsoft and MS-DOS are registered trademarks of Microsoft Corporation.
OPEN LOOK is a trademark of AT&T.
UNIX is a registered trademark of AT&T.
VP/ix is a registered trademark of Phoenix Technologies Ltd. and Interactive
Systems Corporation.
WORDSTAR is a registered trademark of MicroPro International Corporation.

Table of Contents

Chapter 1 :	Introduction to Simul-Task 386	1-1
Chapter 2 :	Installing Simul-Task 386	2-1
Chapter 3 :	Configuring Simul-Task 386	3-1
Chapter 4 :	Simul-Task 386 Fundamentals	4-1
Chapter 5 :	Running MS-DOS and UNIX System Applications	5-1
Chapter 6 :	Using Simul-Task 386	6-1
Chapter 7 :	Utilities	7-1
Chapter 8 :	Using Simul-Task 386 with the XWIN System	8-1
Chapter 9 :	Using Serial Terminals	9-1

Table of Contents

Appendix A:	Terminal Descriptions	A-1
Appendix B:	Error Messages	B-1
Appendix C:	Glossary	C-1
Appendix D:	Corresponding Commands: UNIX System and MS-DOS	D-1
Index		I-1

Chapter 1: Introduction to Simul-Task 386

What Is Simul-Task 386?	1-1
Features of Simul-Task 386	1-1
Getting Started	1-2
About This Book	1-2
Notational Conventions	1-4

What Is Simul-Task 386?

Simul-Task 386 is a UNIX® System utility that provides a multitasking, multiuser MS-DOS® environment. For example, with Simul-Task 386, you can read your electronic mail, answer it, and write a memo—all while you are recalculating a Lotus 1-2-3® spreadsheet in the background.

Features of Simul-Task 386

Simul-Task 386 offers a number of attractive features that make it easy for you to install and use MS-DOS applications as well as UNIX System programs.

- Fully automated installation and removal via **installpkg** and **removepkg**, including automatic installation of the MS-DOS operating system (MS-DOS must be purchased separately.)
- Support for invocation of MS-DOS commands from the UNIX System command line and support for invocation of UNIX System commands from the MS-DOS command line
- Full multitasking and multiuser operation, including support for execution of text-based DOS applications on remote ASCII terminals such as the AT&T 600 series
- Highly integrated MS-DOS and UNIX System file sharing
- Support for popular MS-DOS application programs such as Lotus 1-2-3 and dBASE III and accommodation of popular "ill-behaved" MS-DOS applications

In addition, Version 2.0 has added the following:

- Full UNIX System V/386 Release 3.2 functionality
- Support for the AT&T XWIN Graphical Windowing System (XWIN) environment featuring the OPEN LOOK™ window manager under UNIX System V/386, Release 3.2, allowing you to open multiple windows that can contain both MS-DOS and UNIX System applications
- A new Direct Device Access (DDA) facility that supports a variety of AT-class hardware by allowing the associated DOS device driver direct access to the hardware

- Provision for a 3.5-inch floppy drive as drive 0 or drive 1
- Automated installation of MS-DOS and MS-DOS applications through the Framed Access Command Environment (FACE)
- Support for a wider variety of serial terminals
- A flexible operating environment through the use of Simul-Task 386 menu items as command line executables
- An Installable Emulation Module (IEM) toolkit as part of the standard product, allowing you to install new Simul-Task 386 hardware emulation modules to support AT hardware that is to be shared among users

Getting Started

No matter how you intend to use Simul-Task 386, you should start with a clear understanding of the interface software (if any) installed on your system. This manual explains how to use Simul-Task 386 without user interface software, and with user interface software (FACE and/or XWIN). You will find detailed information in the *AT&T FACE User's Guide* and the *AT&T XWIN Graphical Windowing System User's Guide*.

There is also a Glossary to help new users with unfamiliar terms.

About This Book

There are essentially two parts to this manual:

- Chapters 2 and 3 which contain installation and configuration information. Included in the installation chapter is information on giving users access to Simul-Task 386 and installing MS-DOS. If your users will all use the default configuration file, you should read Chapter 3 for information about this master file, but you do not have to create configuration files.
- Chapters 4 through 9 which are devoted to using Simul-Task 386 in a variety of situations.

Chapter 4 describes Simul-Task 386 simply and succinctly and provides information necessary to get a new user started with the system. Users who are unfamiliar with the UNIX System should not skip this chapter.

Chapters 5 through 9 contain information on using Simul-Task 386:

- Chapter 5 tells you how to run MS-DOS applications and is intended as an overview for users whose system activity will be confined to running these applications.
- Chapter 6 goes into more detail and includes information on installing MS-DOS applications.
- Chapter 7 describes Simul-Task 386 utilities.
- Chapter 8 is devoted to using Simul-Task 386 with XWIN.
- Chapter 9 contains information on serial terminals.

In addition, the appendixes provide the following:

- Appendix A: Terminal Descriptions
- Appendix B: Error Messages
- Appendix C: Glossary
- Appendix D: MS-DOS Commands and Corresponding UNIX System Commands

If Simul-Task 386 is new to you, you may want to read Chapter 4, "Fundamentals of Simul-Task 386," even before you install Simul-Task 386, and then proceed with Chapters 2 and 3.

If you are more familiar with MS-DOS than with the UNIX System, Appendix D, "MS-DOS Commands and Corresponding UNIX System Commands," should help to get you started.

If your system includes a serial terminal, you will need to read Chapter 9 (and possibly Appendix A) to make sure that the terminal is correctly identified to Simul-Task 386 before you run MS-DOS applications.

If your main activity is running MS-DOS applications, read Chapters 4 and 5.

Notational Conventions

The following conventions are used throughout the *Simul-Task 386 User's Guide*.

Keys: Keys or key sequences are shown enclosed in angle brackets (greater than and less than symbols), as <ALT> or <SYS RQ>. Where two or more keys are shown joined with a hyphen (<CTRL> - <ALT> -), you hold down the first key, and while pressing it, press the second (and third) key. This is sometimes referred to as pressing the keys simultaneously.

Command Syntax: Examples of commands appear throughout this book. Bold and italic type are used to indicate what you should enter.

- Words or symbols in **bold type** are to be entered literally, exactly as shown
- Words in *italics* stand for variables for which you should make the appropriate substitution (usually a filename).
- Square [brackets] indicate that the enclosed word (which can be a variable or the actual word to enter) is optional. If you use an option, *do not* enter the brackets.

Examples: `ls -l [filename]`

- `comp file1 file2`
- `cmp [options] file1 file2`
- `diskcopy [source] [target]`

UNIX System File or Command Names: When the text refers to a UNIX System file by name, it is written in bold type to make the name absolutely clear (e.g., **.profile**). When an entire path is given (e.g., /usr/dosapps) and the file or directory name is clear from the context, the name is in ordinary type.

Upper- and Lowercase: The UNIX System is case sensitive; that is, it distinguishes between upper- and lowercase letters. MS-DOS is not case sensitive. To avoid problems, enter commands in lowercase.

Slash and Backslash: Both MS-DOS and the UNIX System use slashes (/) and backslashes (\) though they use them differently. MS-DOS uses forward slashes (here referred to as slashes) as switch characters and back slashes as path delimiters. The UNIX System uses slashes as path delimiters and

backslashes as escape characters. Be sure you are entering the right kind of slash when you use a command from this book.

Chapter 2: Installing Simul-Task 386

Installing Simul-Task 386	2-1
Before You Begin	2-2
Overview of Steps in the Installation Procedure	2-3
Without FACE	2-3
With FACE	2-3
Upgrading Simul-Task 386 from Version 1.0	2-4
Step-by-Step Installation Procedure	2-5
Giving Users Access to Simul-Task 386	2-8
If MS-DOS Has Been Installed	2-10
Removing Access to Simul-Task 386	2-11
Installing and Upgrading the MS-DOS Operating System	2-11
Install/Update Procedure	2-12
Removing Simul-Task 386	2-18
Installing the IEM Tools Package	2-21
What Is Included in the IEM Tools Package	2-22
Installing IEM	2-22
Removing IEM	2-26
Troubleshooting	2-31

Installing Simul-Task 386

Simul-Task 386, Version 2.0, is designed to work in concert with user friendly interfaces such as FACE and XWIN, under UNIX System V/386 Release 3.2. This chapter describes the installation procedure in detail. In addition to this manual, there are other sources of information available to you:

- *AT&T UNIX System V/386 Release 3.2 User's Reference Manual*
- *AT&T Framed Access Control Environment (FACE) User's Guide*
- *AT&T XWIN Graphical Windowing System User's Guide*
- Your MS-DOS user's guide

To perform the installation and setup tasks described in this chapter and the next chapter, you must either be **root** or be a privileged user and you should be at the console.

If you are using FACE, you do not need to be **root**, but you do need to be a privileged user.

If you are not using FACE, you must be **root**.

This chapter also tells you how to give users access to Simul-Task 386. This is not the same as adding users to the system. There is a separate procedure for providing Simul-Task 386 access to users who already have system logins. This procedure, like the installation of Simul-Task 386, can be done through FACE or from the UNIX System prompt. And, as when you install Simul-Task 386, you must be **root** if you are working from the UNIX System prompt. If you are using FACE, you need only be a privileged user.

As your system grows, you may have to add files to **terminfo**. (For users of older UNIX Systems, **terminfo** replaces **termcap**. It is larger and better organized, and can be found in `/usr/lib/terminfo`. The **terminfo** directory contains a subdirectory for each letter of the alphabet; these subdirectories contain the information files.)

You will also be maintaining the Simul-Task 386 master configuration file, `/usr/vpixmap/defaults/vpixmap.cnf`. Alternatively, users can choose to have their own configuration files.

Installing Simul-Task 386 software is simple and quick. The installation process also includes giving users access to Simul-Task 386 and installing MS-DOS, so allow yourself enough time.

Before You Begin

1. Check your Simul-Task 386 package.

Your Simul-Task 386 package should include the following:

- *AT&T Simul-Task 386 User's Guide* (this document)
- Two diskettes, one containing Simul-Task 386 and the other containing the Installable Emulation Module (IEM) Tools Package
- *Simul-Task 386 Release Notes*

2. Check hardware requirements.

Simul-Task 386 requires the following hardware:

- An AT&T 6386 WGS computer or a compatible Intel 80386-based computer
- A minimum of 2 Mb of memory, *preferably 4 Mb*
- A 40 Mb fixed disk, *preferably 68 Mb*
- One high-density 5.25-inch or 3.5-inch diskette drive
- An EGA, CGA, or Monochrome Display Adapter (MDA) as the console
- A *minimum* of 2 MB of disk space for Simul-Task 386 files.

In addition, the following optional hardware is supported:

- Serial terminal
- AT&T 3-button mouse.

3. Check software requirements.

The AT&T UNIX System V/386 Release 3.2 Operating System must be installed on your system before attempting to install Simul-Task 386. Also, successful operation of Simul-Task 386 will require that you install AT&T's MS-DOS Version 3.2 or greater.

4. If you plan to use AT&T FACE, it should be installed before you install Simul-Task 386.
5. If you plan to install the AT&T XWIN Graphical Windowing System, check the installation requirements for the XWIN system, either in the XWIN User's Guide or in Chapter 8 of this manual.

Overview of Steps in the Installation Procedure

The following is a brief summary of the installation procedure. Detailed instructions can be found in the "Step-by-Step Installation Procedure" section.

Without FACE

1. Log in at the console as **root**.
2. Install Simul-Task 386 by entering **installpkg** and following the prompts.
3. If MS-DOS has not been installed or is not at least Version 3.2, install or upgrade MS-DOS by entering **installdos** and following the prompts.
4. Install MS-DOS applications packages. (This procedure is performed frequently, so it is covered in "Using Simul-Task 386.")
5. If you are going to use DDA or if you are making use of third party device support under Simul-Task 386, you will need to install the IEM toolkit.

With FACE

1. Log in at the console as a privileged user.
2. From the FACE Software Setup Menu, select *Install UNIX System Application* and follow the prompts
3. From the *Software Setup* menu, select *Install MS-DOS System* or *Install MS-DOS Software* (depending on whether you are installing or upgrading MS-DOS) and follow the prompts.
4. Install MS-DOS applications packages. (This procedure is performed frequently, so it is covered in "Using Simul-Task 386.")
5. If you are going to use DDA or if you are making use of third party device support under Simul-Task 386, you will need to install the IEM toolkit.

Upgrading Simul-Task 386 from Version 1.0

If you are upgrading your system from Simul-Task 386 Version 1.0 to Version 2.0, there are some preliminary steps:

1. *Do not remove Version 1.0.* You will be installing Version 2.0 on top of it.
2. Back up any software on your C: drive so you can restore it later after the upgrade is done.
3. If you are not **root**, become **root** and invoke MS-DOS with the **dos** command.
4. Modify **config.sys** by adding the line: **fcbs=40,40**.
5. Modify **autoexec.bat** by changing the command **netup** to its new name, **dosmount**. Do not change the rest of that line.
6. Exit from MS-DOS and continue with Step 1 of the installation procedure.
7. Don't forget to reinstall MS-DOS.

NOTE

If you don't update **config.sys**, your Simul-Task installation will still be successful; however, you will see a warning message when you invoke MS-DOS: Warning: please increase the number of FCBs in CONFIG.SYS. At this point, you must update **config.sys**.

If you have already removed Version 1.0, perform the following steps before beginning installation of Version 2.0:

1. If you are not **root**, become **root**.
2. Edit the `/etc/conf/sdevice.d/vx` file, changing the "N" to a "Y."
3. Save the file.
4. Execute `/etc/conf/bin/idbuild`. (All you have to do is enter the full path name of the **idbuild** command as given.) You will see a message, telling you that **idbuild** takes a few minutes.
5. When **idbuild** is finished, you will have a shell prompt. Shut down your system by running `/etc/shutdown`. (Just enter **/etc/shutdown**.)

6. Reboot the system and proceed with Step 1 of the installation procedure instructions.

Step-by-Step Installation Procedure

To install Simul-Task 386, do the following:

1. Enter **installpkg** at the UNIX System prompt or select *Install UNIX System* from the *FACE Software Setup* menu.

After the system checks that the install program is being run from the console and you have permission to execute privileged operations, the following message appears:

```
Please insert the floppy disk.
```

```
If the program installation requires more than  
one floppy disk, be sure to insert the disks in  
the proper order, starting with disk number 1.
```

```
After the first floppy disk, instructions will  
be provided for inserting the remaining floppy  
disks.
```

```
Strike ENTER when ready  
or ESC to stop.
```

2. Insert the diskette labeled Simultask 386 into the drive and strike <Enter>. You see the following message:

```
Installation is in progress -- do not remove the floppy disk.
```

The system executes the install script provided on the Simul-Task 386 diskette, and the following message is displayed:

Successful installation of the AT&T Simul-Task 386 Package Version 2.0 will require that you install AT&T's MS-DOS operating system either version 3.2 or 3.3.

If you do not have either of these MS-DOS versions, you may cancel the installation at this time by striking ESC.

Strike ENTER when ready
or ESC to stop.

3. If the IEM Tools have already been installed, you will see the following message.

Warning: The IEM Tools package is installed on the system. If you continue reinstalling Simul-Task 386 Version 2.0, any driver support that may have been added to Simul-Task will have to be re-added through the IEM mechanism.

Strike ENTER when ready
or ESC to stop.

4. Strike <Enter> and the following message is displayed:

Installing AT&T Simul-Task 386 Package Version 2.0 ...

Installation continues with the install script replacing certain UNIX System files with new Simul-Task 386 versions of the files. You see the following message:

```
You will now be asked to select from those user
logins, currently on your system, the set of users
who will be allowed access to the Simul-Task 386
environment.
```

```
Strike ENTER when ready
or ESC to stop.
```

5. If <Esc> is selected, installation continues with Step 6. If <Enter> is selected, the following message is displayed, and you are prompted to allow users access to the Simul-Task 386 environment.

```
Currently Installed Users Who Are Not
Currently Simul-Task 386 Users:
```

```
install
root
shb
```

At this point, the installation script executes INSTALL.USR, described in the next section, "Giving Users Access to Simul-Task 386." After execution of INSTALL.USR, installation continues with the following screen.

Installation of Simul-Task 386 Version 2.0
is now complete.

Remember that when adding new user logins to the UNIX system, they will need to be explicitly provided access to the Simul-Task 386 environment. To do this, use the super-user command, "INSTALL.USR" from the UNIX System prompt.

You may now proceed to install
the MS-DOS operating system.

Strike ENTER to continue
or ESC to stop.

6. After the above screen is displayed, strike <Enter> to install the MS-DOS operating system, and read the "Installing and Upgrading the MS-DOS Operating System" section.

Giving Users Access to Simul-Task 386

The following procedure allows additional users access to the Simul-Task 386 environment.

NOTE

You must be **root** to run **INSTALL.USR**. Only users that have been added using **INSTALL.USR**, either during installation of Simul-Task 386 or from the shell prompt, will have access to the Simul-Task 386 environment.

1. At the shell prompt, enter **INSTALL.USR**. The following message is displayed:

Confirm

This procedure sets up a user's environment to allow the use of the Simul-Task 386 Environment. If you wish to set up a new Simul-Task 386 user, strike ENTER. Otherwise, strike ESC to cancel.

Strike ENTER when ready
or ESC to stop.

2. Strike <Enter> and the following message is displayed:

Enter the User's login ID:

3. Enter an existing login (for example, **bob**) and strike <Enter>. The procedure validates the login to make sure that it exists and that a home directory for that login exists. It also makes sure that the user is not already a Simul-Task 386 user.

If the login validation is successful, the following confirmation screen is displayed:

Confirm

If you wish to add <login> as a Simul-Task 386 user, strike ENTER. Otherwise strike ESC to cancel.
Strike ENTER when ready
or ESC to stop.

4. Strike <Enter> and <login> is given access to Simul-Task 386. If MS-DOS has not been installed, the default configuration is given to <login> and the command displays the following confirmation message:

User <login> can now use the Simul-Task Environment

The procedure then returns to Step 1 giving the user an opportunity to set up another login or to stop.

If MS-DOS Has Been Installed

If MS-DOS has been installed, the following message is displayed giving the user an opportunity to create a private copy of the C: drive instead of getting the system default configuration.



Most of the installation screens prompt you to press <Enter> for the most probable action, and <Esc> for the less likely action. The following screen prompts you to press <Enter> to make a private copy of the C: drive. This uses up disk space and in most cases you would not want to do it.

Unless you want a private copy of the C: drive, press <Esc> at the next prompt.

Refer to the Simul-Task 386 User's Guide for details on the C: drive. If you would like to make a private copy of the C: drive, strike ENTER. NOTE: The C: drive currently occupies <variable> bytes on the hard disk. If you would prefer to save space and use the default C: drive, you may do so by striking ESC.

Strike ENTER when ready
or ESC to stop.

If you strike <Enter>, a private copy of the C: drive is created. If you strike <Esc>, the defaults are used. In either case, the following confirmation message will be displayed to indicate that the login now has access to Simul-Task 386:

User <login> can now use the Simul-Task Environment

The procedure then returns to Step 1 giving you an opportunity to set up another login or to stop.

Removing Access to Simul-Task 386

The following procedure removes a user's access to the Simul-Task 386 environment.

NOTE

You must be **root** to run **REMOVE.USR**.

To remove a user from the Simul-Task 386 environment, type **REMOVE.USR login**. The following message is displayed:

User <login> can no longer use the Simul-Task 386 Environment.

Note that this does not remove the user's files, only his or her access to Simul-Task 386.

Installing and Upgrading the MS-DOS Operating System

This section provides procedures for installing or upgrading MS-DOS, version 3.2 and above, in the Simul-Task 386 environment.

The installation creates three virtual hard disks to which MS-DOS will have access and which are recommended for use in the following ways:

- C: This is the boot drive and is intended to contain only the MS-DOS system files, COMMAND.COM, AUTOEXEC.BAT, and CONFIG.SYS. Users should not install applications software on this drive. Drives D: and E: are provided for this purpose.
- D: A pseudo-volume used only for installing MS-DOS applications which are copy-protected and "expect" to be installed on a true MS-DOS format hard disk (i.e., the applications write "fingerprint" information to areas of the disk which are not accessible through normal MS-DOS commands).
- E: A regular UNIX System directory, /usr/dosapps, used for installing all other MS-DOS applications which are to be made available to Simul-Task 386 users.

Users will also be able to reference the U: drive which points to the entire UNIX System file system, but whose current directory is the same as when MS-DOS was invoked. Output data from applications programs should be stored on the H: drive which points to the user's home directory.

Users can install applications in their home directories or private C: drives; however, this should be discouraged because it wastes disk space if several users install the same application.

Install/Update Procedure

1. After installing Simul-Task 386, you are prompted to install the MS-DOS operating system. If you do not want to install MS-DOS at this time, it can be installed later, using one of the following methods:
 1. Type **installdos** from the UNIX System command line
 2. If AT&T FACE is installed, go to the *Software Setup* menu and select *Install MS-DOS System* to install MS-DOS for the first time, or select *Install MS-DOS Software* to upgrade your version of MS-DOS.

If you are installing MS-DOS for the first time, the following message is displayed:

The MS-DOS system will now be loaded onto your hard disk.
When MS-DOS boots, you will need to carry out the following steps:

1. Strike ENTER at the date prompt.
2. Strike ENTER at the time prompt.
3. Type d:dosadd at the A> prompt and strike ENTER.

You may strike Esc to cancel this procedure now if you do not have the MS-DOS system disk available.

Strike ENTER when ready
or Esc to Stop

If you are upgrading to a more current version of MS-DOS, the following message is displayed:

The MS-DOS system is already installed.

This procedure will replace your existing MS-DOS system with a new version. Please note that this will involve the removal and re-creation of the "public" C: drive. Be sure to back up any files that you may have stored on this drive before proceeding further.

The upgrade will not affect currently installed applications or any programs and data files stored in your home directory. Any private C: volumes in your home directory may be incompatible with the new MS-DOS version.

You will need to have the MS-DOS System/Startup and Supplemental Programs disks available in order to complete this upgrade.

Strike ENTER when ready
or Esc to stop.

NOTE

The bootable MS-DOS diskette must *not* have an AUTOEXEC.BAT file on it. If you can, use the original distribution diskette for the installation procedure.

2. If you are upgrading an earlier version of MS-DOS, you may see an additional screen at this point:

Saving autoexec.bat and config.sys files

3. Installation continues with the following message:

Please insert the MS-DOS System Disk (Version 3.2) or Startup Disk (Version 3.3) in the diskette drive and strike ENTER to continue

4. Insert the appropriate diskette (System/Startup) in drive A: and strike <Enter>.

After MS-DOS has booted,

- You are prompted to enter the new date. Strike <Enter> and you are prompted to enter the new time.
- Strike <Enter> and you are left at the A> prompt.

NOTE

If the version of MS-DOS that you are using does not take you to the A> prompt, follow the instructions on the screen until the A> prompt appears.

5. Enter **d:dosadd** and the following message is displayed:

Transfer in progress do not remove the floppy disk.

When the file transfer is complete, the following message is displayed:

Please remove the MS-DOS System/Startup Disk from the diskette drive and strike ENTER.

6. Remove the diskette, strike <Enter>, and the following message is displayed:

Installing MS-DOS commands - Please Wait ..

NOTE

If MS-DOS version 3.3 is being installed and if you are using 5.25-inch diskettes, the following message is displayed after the Startup commands have been copied to the hard disk.

Please insert the Operating Disk
and strike ENTER to continue

Insert the diskette, strike <Enter>, and the following message is displayed:

Transfer in progress do not remove the floppy disk.

When the file transfer is complete, the following message is displayed:

The MS-DOS system files have been transferred
to the hard disk. The next step will copy the
MS-DOS Supplemental Programs to the hard disk.

Strike ENTER when ready
or Esc to Stop.

7. If <Esc> is entered, installation continues with Step 8. If you strike <Enter>, the following message is displayed:

Please insert the Supplemental disk into the Diskette Drive
Strike ENTER to continue

8. Insert the diskette, strike <Enter>, and the following message is displayed:

Transfer in progress do not remove the floppy disk.

9. When the file transfer is complete or if <Esc> was entered in Step 6, the following message is displayed:

If you wish to install the MS-DOS Help function, please remove the MS-DOS Supplemental Programs from drive A: (if present) and make sure the diskette drive is empty. If you do not wish to install the MS-DOS Help function, press Esc to skip this step.

10. If you do not wish to install MS-DOS Help at this time, strike <Esc> and proceed with Step 11. If installing MS-DOS Help, the following message is displayed:

Please insert the Help disk into the Diskette Drive
Strike ENTER to continue

11. Insert the diskette, strike <Enter>, and the following message is displayed:

Transfer in progress do not remove the floppy disk.

When the file transfer is complete, the following message is displayed:

It is safe to remove the floppy disk.

Please wait while the system files are being set up.

When the setup is complete, the following message is displayed:

The MS-DOS system has now been successfully installed on your system. To run it, either select "MS-DOS System" from your "Main" menu or type "dos" at the command line.

12. The installation of MS-DOS is now complete.

Removing Simul-Task 386

If your hard disk is low on space, you may wish to remove some software packages which are not commonly used. You can also use the **removepkg** command to remove Simul-Task 386. You can also use FACE to remove software, including Simul-Task 386.

NOTE

If installed, the IEM Tools Package must be removed from the system before Simul-Task 386 can be removed.

1. To remove Simul-Task 386,
 - If you are not using FACE, you must be **root**. Type **removepkg** without arguments and strike <Enter>.
 - If you are using FACE, you must be a privileged user. Select *Remove Installed Software* from the *Software Setup* menu,

The system checks to see if you have permission to execute privileged operations. If you don't, you will see the following message:

```
You (<logname>) do not have permission to
perform software removal.
Please consult your Operations/System
Administration Guide for more
information on assigning permissions
to privileged operations.
```

Refer to the Operations/System Administration Guide.

If you are a privileged user, you then see a sorted listing of installed packages with the following message:

```
Select a number (1-n) from this list to remove:
```

NOTE

n is the number of packages currently installed.

2. To remove Simul-Task 386 (or any other software installed on the system), type the number corresponding to the package, strike <Enter>, and you see a confirmation screen, asking whether you really want to remove the software you have selected. For example, if you select AT&T Simul-Task Version 2.0, the following message is displayed:

```
Do you really want to remove
AT&T Simul-Task 386 Package Version 2.0?
```

```
Strike ENTER when ready
or ESC to stop.
```

3. Strike <Enter> to remove the package.

If you decide not to remove the package or if you specified the wrong one, strike <Esc> and you see the following message:

```
AT&T Simul-Task 386 Package Version 2.0 not removed.
```

After striking <Enter>, in step 3, the following message is displayed.

```
WARNING: If you still have MS-DOS software applications
installed on this system and you continue with the
removal of Simul-Task 386, you will be unable to run or
remove these applications and will have to use UNIX System
tools to remove them later.
```

```
Also, as part of the removal of the Simul-Task 386 package,
the MS-DOS operating system will automatically be removed.
Do you wish to proceed with the removal?
```

```
Strike ENTER when ready
or ESC to stop.
```

4. Strike <Enter>, and the following message is displayed.

```
Removing AT&T Simul-Task 386 Package Version 2.0 ...
```

After the removal of Simul-Task 386 files, the following message is displayed. This completes the removal process.

```
The AT&T Simul-Task 386 Package Version 2.0
files have now been removed.  If DOS applications
are still installed on your system or individual
users have private Simul-Task 386 related files,
they should remove them as soon as possible in order
to provide more free space on the hard disk.
```

For additional information on **removepkg**, see the Operations/System Administration Guide.

Installing the IEM Tools Package

The installation of this diskette is optional since the Simul-Task 386 program diskette contains all the files required for the operation of Simul-Task 386. However, the IEM tools diskette must be installed if you are going to use the Direct Device Attachment facility (described in Chapter 3) or make use of Simul-Task 386 driver software supplied by third party AT board manufacturers (described in the Integrated Software Developer's Guide).

This section is a step-by-step guide for installing and removing the AT&T Simul-Task 386 Installable Emulation Module (IEM) Tools Package. Before proceeding with installation, you must first have installed AT&T Simul-Task 386 Version 2.0. Version 2.0 is the first version of Simul-Task 386 that supports extension of the Emulation Control Task (ECT)

NOTE

This tools package is not supported and cannot be installed if you have an earlier version of Simul-Task 386 installed on your system.

What Is Included in the IEM Tools Package

The IEM tools package includes:

- The direct device access (DDA) software and command set
- The IEM command set used to effect changes to your Simul-Task 386 configuration
- A Simul-Task 386-/MS-DOS emulation module for the AT&T Bus Mouse
- A loadable Simul-Task 386 binary, including the ECT extensions for the virtual device interface (VDI)
- A UNIX System driver for the VDI

Installing IEM

1. Log in at the console as **root**.

As with other AT&T software packages, the **installpkg** command can be used to install the IEM package.

You can install the IEM Tools package using FACE, but you must still log in as **root** at the console in order to perform the system reboot required in Step 8.

If you are not logged in as **root**, **installpkg** displays the message, "You must be root or super-user to install software."

2. If you are logged in as **root**, installation proceeds with the following message:

Confirm

Please insert the floppy disk.

If the program installation requires more than one floppy disk, be sure to insert the disks in the proper order, starting with Disk number 1.

After the first floppy disk, instructions will be provided for inserting the remaining floppy disks.

Strike ENTER when ready
or ESC to stop.

NOTE

If you have more than one floppy drive (e.g. a 3.5-inch and a 5.25-inch drive), **installpkg** asks you which drive you are using.

3. If you wish to proceed with the installation, insert the floppy disk and strike <Enter>. The following message is displayed:

Installation is in progress -- do not remove the floppy disk.

4. This message is followed by:

Searching for the Size file

5. This message is followed by:

Install in progress

If the IEM tools package has already been installed, the following message is displayed:

Confirm

The Simul-Task 386 Installable Emulation Module Tools Version 2.0 has already been installed. The new installation will now replace the original Simul-Task 386 Installable Emulation Module Tools Version 2.0 files.

Strike ENTER when ready
or ESC to stop.

You can proceed with the reinstallation by striking <Enter> or you can stop by striking <ESC>. Reinstalling will reinstall the IEM tools package. Reinstallation will recover files that may have been accidentally removed. It is not necessary to remove the IEM tool package prior to reinstallation.

6. At this point, the package is read into a temporary location on the hard disk. Once this process is complete, the IEM installation script is executed.

The installation script first verifies that you have the correct version of Simul-Task 386 installed. Then the following message is displayed:

```
Installing Simul-Task 386 Installable Emulation Module  
Tools Version 2.0
```

7. The IEM Tools package includes a UNIX System driver for the VDI. If the VDI driver is not installed on your system, the following message is displayed on the screen:

```
Installing Simul-Task 386 Installable Emulation Module  
Tools Version 2.0
```

```
The UNIX Operating System will now be rebuilt.  
This will take approximately 2 minutes. Please wait.
```

```
The UNIX Kernel has been rebuilt.
```

8. Assuming a new kernel has just been relinked, the following message is displayed:

To complete the install/remove process a system shutdown is necessary. You must be logged onto the console (/dev/console) and be using virtual terminal 0 in order to shut the system down.

You should now exit, log in at the console, and manually reboot as soon as possible.

Make sure your floppy drive is empty. If you are installing or removing controller boards, you may power down the system after the shutdown has completed.

Strike any key to continue.

9. If you intend to install other packages on your system that also install UNIX System drivers, you can press <ESC> to suspend the shutdown.

NOTE

You must perform a shutdown before you attempt to reconfigure your Simul-Task 386 environment. Not doing so may yield unpredictable results.

If you don't need a new kernel or if you opted not to reboot the system at this time, the following message is displayed, followed by the shell prompt:

The installation of the Simul-Task 386 Installable Emulation Module Tools Version 2.0 is now complete.

Removing IEM

1. Log in at the console as **root**.

As with other AT&T software packages, the **removepkg** command can be used to remove the IEM package, or you can use **FACE**. In either case, you must be logged in as **root** at the console.

If you are not logged in as **root**, **removepkg** displays a message (as it does with the **installpkg** command) and exits.

To verify that the IEM tools package has been installed on your system, use the **displaypkg** command.

When you execute **removepkg**, you see a menu, listing all the packages currently installed on your system. To remove the IEM Tools, type the number that corresponds to this package.

2. The following screen is similar to the one you see when you execute **removepkg**:

```
1      2 Kilobyte File System Utility Package Version 2.0
2      Bus Mouse Driver Package - Version 1.0
3      C Software Development Set      4.1.5
4      Editing Package Version 2.0
5      Extended Terminal Interface Package Version 2.0
6      FACE HELP Version-1.2
7      FMLI Version 1.2
8      Security Administration Package Version 2.0
9      Simul-Task 386 Package Version 2.0
10     Simul-Task 386 Installable Emulation Module Tools Version 2.0
11     Remote Terminal Package Version 2.0
```

Select a number (1 - 11) from this list to remove:

3. Next, a confirmation message is displayed that allows you to abort package removal if you have made an incorrect selection:

Confirm

Do you really want to remove Simul-Task 386 Installable
Emulation Module Tools Version 2.0?

Strike ENTER when ready
or ESC to stop.

4. If you have ECT or DDA (Direct Driver Access) devices configured into your Simul-Task 386 system, the following message is displayed when you strike <Enter>:

Confirm

WARNING: You still have Simul-Task 386 device emulation modules installed on your system. If you proceed with the removal of this package, these modules will remain and will continue to occupy storage on the hard disk. You will be unable to remove them unless you re-install this package.

Strike ESC now if you would prefer to cancel the removal of this package or ENTER to continue.

Strike ENTER when ready
or ESC to stop.

Note that you are permitted to remove the IEM package under these conditions. As the message implies, changes to your Simul-Task 386 configuration will no longer be possible unless you reinstall the package. The space used by the VDI driver and the the emulation modules will not be recovered. They are not removed. If your intent is to recover the largest possible amount of space, you should first remove any IEM and DDA entries configured into Simul-Task 386 with the **IEM ectremove** and/or the **ddaremove** command. An exception to this rule is the AT&T Bus Mouse (IEM name **bmouse** and logical name

MOUSE). Removal of the AT&T Bus Mouse IEM is not required.

NOTE

If you install a mouse on your system other than the AT&T Bus Mouse (3-button), you should select an IEM name and a logical name that do *not* correspond to those used by the AT&T Bus Mouse (listed above).

5. If you remove the IEM package with IEM/DDA entries configured in, subsequent reinstallation of the IEM Tools package will not require reinstallation of the IEM/DDA entries; they are preserved.

If you have not removed all IEM/DDA entries, the VDI driver is not configured out of the UNIX kernel, and the following message is displayed:

```
Removing Simul-Task 386 Installable Emulation Module Tools  
Version 2.0 ...
```

6. This message is followed by:

```
Package removal complete. Please note that the  
emulation modules have been preserved and may  
only be removed by re-installing this package  
and running ddaremove/ectremove.
```

7. Next you see this message and, finally, the shell prompt:

The Simul-Task 386 Installable Emulation Module Tools
Version 2.0 is now removed.

8. If you have removed all IEM/DDA entries, the VDI driver is configured out of the UNIX kernel, and this message is displayed instead:

```
Removing Simul-Task 386 Installable Emulation Module Tools  
Version 2.0 ...
```

```
The UNIX Operating System will now be rebuilt.  
This will take approximately 2 minutes. Please wait.
```

```
The UNIX Kernel has been rebuilt.
```

```
Simul-Task 386 Installable Emulation Module Tools Version  
2.0 has been removed.
```

9. If a new kernel was relinked, the following message is displayed:

To complete the install/remove process a system shutdown is necessary. You must be logged onto the console (/dev/console) and be using virtual terminal 0 in order to shut the system down.

You should now exit, log in at the console, and manually reboot as soon as possible.

Make sure your floppy drive is empty. If you are installing or removing controller boards, you may power down the system after the shutdown has completed.

Strike any key to continue.

As with **installpkg**, you can forgo a system reboot if you intend to remove other packages that remove UNIX System drivers. In other words, if you are going to remove three packages that all involve UNIX System drivers, you can omit the first two reboots by pressing <ESC>. Press <Enter> the third time you see the above prompt (after removal of the third and last package).

NOTE

Always remember to reboot your system if you've relinked a new kernel.

If you plan to shut down your system, ensure that other users will not be inconvenienced by this action. There is no grace period for users during this process.

Troubleshooting

The following table is provided as a quick checklist in case you experience difficulties when installing or removing the IEM package.

Question	Explanation
Are you logged in as root ?	Package installation and removal can be performed only by root .
If rebooting your machine, are you logged in on the system console?	You cannot reboot unless you are logged in at the system console. This is done for your protection.
Have you installed Simul-Task 386 Version 2.0?	The IEM Tools will not install if you have the AT&T Simul-Task 386 Version 1.0 software installed. The older version does not have ECT extension capability.
Are you experiencing difficulty when rebuilding the UNIX Operating System?	You may examine the contents of an error file (the name of the file is part of the error message) and attempt to reconstruct the events leading up to the error. You may also find it helpful to consult your user documentation under the ID/TP Tools section. If you are unsuccessful, please contact your AT&T Service Representative or your AT&T Personal Computer Dealer.
Was an error encountered during the automatic installation of the VDI driver?	

Question	Explanation
<p>Did you get a WARNING when removing the IEM package?</p> <p>Do you need to recover as much disk space as possible?</p>	<p>You may continue with the removal despite this warning. You are being reminded that you are not removing the package in its entirety. To remove the entire package, you must remove all IEM and DDA entries (except for the AT&T Bus Mouse) before you remove the IEM Tools package.</p>
<p>Are you logged in as root when you attempt to access the IEM tools after installation?</p>	<p>Execution of the ectinstall, ectremove, ectconfig, ddainstall, ddaremove, and ddachange tools is limited to the root login. The only time this is not true is when you have FACE installed. If FACE is installed, you must be a privileged user in order to have access to these tools.</p>

Chapter 3: Configuring Simul-Task 386

Configuring Simul-Task 386	3-1
The Master vpix.cnf File	3-1
Contents of the vpix.cnf File	3-2
Example vpix.cnf File	3-2
The DOSPATH Variable	3-7
DOSPATH Example	3-8
The VPIXCNF Variable	3-9
Search Precedence	3-9
Example VPIXCNF	3-9
The VPIXTERMDIR Variable	3-10
Override Order	3-10
Enabling Peripheral Devices	3-11
Printers	3-11
Mouse	3-11
Diskettes	3-12
Pseudo-Volume Fixed Disks	3-12
Serial Ports	3-13
Direct Device Access	3-14
DDA Installation and Removal	3-17

Configuring Simul-Task 386

This chapter provides information about Simul-Task 386 configuration files. Simul-Task 386 allows each user to have his or her own configuration file; however, a default configuration file is provided when Simul-Task 386 is installed. All users can use this default configuration file, and if they do, you do not have to perform the procedures described in this chapter.

Simul-Task 386 provides for a configuration file for each user and recognizes certain shell environment variables:

vpix.cnf	is the configuration file. It specifies access to resources and maps them between MS-DOS and the UNIX System.
-----------------	---

The shell environment variables recognized by Simul-Task 386 are as follows:

DOSPATH	determines which directories in the user's search path (PATH) are checked for MS-DOS applications.
VPIXCNF	specifies the path name of the user's Simul-Task 386 configuration file.
VPIXTERMDIR	specifies the path name for the directory that contains a terminal specification file that overrides the default.

These variables can be set within a **\$HOME/.profile** file.

This chapter also provides information for enabling peripheral devices.

The Master **vpix.cnf** File

The installation procedure places a default configuration file in the `/usr/vpix/defaults` directory. Unless a user requests a private copy of the C: drive, the user always shares this master **vpix.cnf** file. This guarantees that the user has access to the most recent version of **vpix.cnf**.

If the user does request a private copy of the C: drive, the `INSTALL.USR` program copies the master default `vpix.cnf` to `$HOME/vpix/vpix.cnf`.

You can update or change the master `vpix.cnf` in the `/usr/vpix/defaults` directory. All Simul-Task 386 users share the master file unless they are using private copies of the C: drive.

You can also change an individual `vpix.cnf` file to minimize resource conflicts. You can assign the devices for use during an individual Simul-Task 386 session.

Contents of the `vpix.cnf` File

The `vpix.cnf` file maps MS-DOS device names (such as `A:`, `C:`, and `COM1:`) to devices in the UNIX System environment.

NOTE

During the `INSTALL.USR` procedure, if a private C: drive is selected, the `vpix.cnf`, `C:`, and `unixbin` files are created.

For example, Drive `A:`, which is normally the first diskette drive in MS-DOS, can be mapped to `/dev/rdisk/f0t`. When a Simul-Task 386 user requests a directory listing of Drive `A:` (with `DIR`) the request goes to the UNIX System environment, which reads and displays the contents of the diskette. Conversely, if there is no entry for Drive `A:` in the user's `vpix.cnf` file, then the same request listing results in an error message:

```
Drive not ready reading Drive A
Abort, Retry, or Ignore?
```

Mapping within the `vpix.cnf` file lets you customize the virtual PC environment. You can edit this file with any UNIX System text editor.

In addition to familiar MS-DOS device names, the `vpix.cnf` file contains entries that are specific to the Simul-Task 386 environment.

Example `vpix.cnf` File

The format of each line of the `vpix.cnf` file consists of entries containing the MS-DOS device name and the UNIX System device name separated by either spaces or tabs.

You can include comments by placing a semicolon (;) as the first character on the comment line.

The following is a sample configuration file. Each entry is discussed in the paragraphs that follow.

```
; Simul-Task 386 Configuration File
;
;
; DOS DEVICE          UNIX DEVICE PATH NAME

C                    /usr/vpix/defaults/C:
ROM                  /usr/vpix/defaults/rom
EGAROM               /usr/vpix/defaults/romega
VGAROM               /usr/vpix/defaults/romvga
CMOS                  /usr/vpix/defaults/cmos
BOOTIMAGE            $HOME/vpix/vpix.img

; The following entries are provide as examples
; for customization of the Simul-Task 386 environment

; To access the floppy as drive A: uncomment the following.
;
A                    /dev/rdsk/f0t

; To add an additional Pseudo-Floppy (accessible as Drive B:);
;B                    $HOME/B:

; To add an additional floppy which is attached to the second floppy drive
; (accessible as B:) on your system:
;
;B                    /dev/rdsk/f1t

; To access a serial communications port:
;COM1                  /dev/tty00

; To access a serial communications port for use with a serial mouse:
;COM1MOUSE            /dev/tty00

; If you have a Mouse:
;MOUSE                 /dev/mouse
```


; If you have a line printer and are using the standard UNIX printer spooler:
LPT1 /usr/bin/lp

1. The first entry in the configuration file is for the MS-DOS C Drive.

```
;
; Simul-Task 386 Configuration File
;
; DOS DEVICE            UNIX DEVICE PATH NAME
;
C                        /usr/vpix/defaults/C:
```

Drive **C:** under standard MS-DOS contains system boot programs such as `COMMAND.COM`, `CONFIG.SYS`, and `AUTOEXEC.BAT`. MS-DOS uses these programs to boot Simul-Task 386 and set up the operating environment. The `/usr/vpix/defaults/C:` file, created when MS-DOS is installed, provides a bootable image that all users can share to get into MS-DOS.

NOTE

Only **root** can modify the default **C:** Drive. This restriction prevents accidental corruption of the boot device.

2. The next lines contain Simul-Task 386-specific configuration information.

```
ROM                    /usr/vpix/defaults/rom
EGAROM                /usr/vpix/defaults/romega
CMOS                   /usr/vpix/defaults/cmos
```

The first line entry defines a file image of the standard MS-DOS BIOS which is normally stored in the computer's ROM. The ROM entry lists the path and file from which to load a virtual image of the ROM BIOS. The **EGAROM** entry is for those who have an EGA adapter and display. This entry should be unchanged even if you use serial terminals or monochrome displays, since Simul-Task 386 will

automatically determine not to load an EGA ROM image for non-EGA users. If you have a non-standard EGA adapter card, you can use the BIOS located on the adapter card by removing the **EGAROM** line from the configuration file.



Not all EGA video adapter ROMs are Simul-Task 386-compatible.

The **CMOS** entry defines a file image of the CMOS information normally stored in the battery backup CMOS RAM. This **cmos** file is for the setup configuration information.

3. The next entry defines the Simul-Task 386 quickboot image file.

```
BOOTIMAGE    $HOME/vpix/vpix.img
```

The MS-DOS image is loaded from the specified file. The Simul-Task 386 image file contains a fast loading image of a user's Simul-Task 386 environment. This quickboot image is used:

- to enter Simul-Task 386 quickly without going through the MS-DOS boot sequence (with the command **dos -r**) or **dos -c**.
- to run an MS-DOS application program directly from the UNIX System.

NOTE

With the quickboot image, Simul-Task 386 is entered quickly from a known saved state. If the quickboot image does not exist, the Simul-Task 386 system automatically creates it the first time you fast load Simul-Task 386. The quickboot image is recreated each time that Simul-Task 386 sees that the master configuration file or your personal configuration file has been changed. If you are running from a different display device (console instead of terminal or vice versa), or Simul-Task 386 has been reconfigured with devices using the IEM tools, the quickboot image is also recreated.

- The next section of the configuration file contains entries for diskette drives.

```
;
; The following entries are provided as examples
; for customization of the Simul-Task 386 environment
;
; To access the floppy as Drive A: uncomment the following.
;
A /dev/rdsk/f0t
;
; To add an additional Pseudo-Floppy (accessible as Drive B):
; B $HOME/B:
;
```

The string `$HOME` represents a shell variable that is equivalent to your home directory path.

NOTE

The **vpix.cnf** file comes automatically configured with the A: drive assigned to the first floppy disk drive. In addition, if there is a B: drive on the system, the installation procedure will detect this and assign the B: drive to the second floppy disk drive.

```
; To add an additional floppy which is attached to the second floppy drive
; (accessible as B:) on your system:
;
; B /dev/rdsk/f1t
;
; To access a serial communications port:
;COM1 /dev/tty00
```

To use an application program that requires serial communication, remove the semicolon in front of the **COM1** line entry. Change `/tty00` to the UNIX System device name for the desired serial port.

A **COM2** entry can be added for a second serial communications port.

5. The next line of the configuration file identifies the bus mouse by providing the mouse device path name.

```
MOUSE /dev/mouse
```

This entry is specific to the AT&T Bus Mouse (3-button) pointing device. It does not apply to serial pointing devices that use COM1 or COM2 .

The **COM1MOUSE** entry is used with a serial mouse. ; To access a serial communications port for use with a serial mouse:
;COM1MOUSE /dev/tty00 ; See the "Serial Ports" section for more information on connecting a serial mouse.

6. The last lines of the configuration file identify printer redirection. The UNIX System parameter attaches the output stream of an MS-DOS printer to a UNIX System printer spooler, as follows:

```
; Printer redirection  
;  
LPT1 /usr/bin/lp
```

In this entry, **lp** spools the output to the default line printer. If you also need a second or third printer, you could add entries such as:

```
LPT2 /usr/bin/lp -Plj1  
LPT3 /usr/bin/lp -Php
```

The **DOSPATH** Variable

The **DOSPATH** variable augments the standard shell variable **PATH**. **DOSPATH** specifies the directories in **PATH** to be searched for MS-DOS applications. With **DOSPATH** set, the shell searches for UNIX System applications before MS-DOS applications.

NOTE

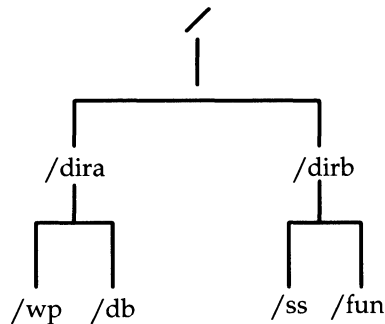
You must include any directory paths in **PATH** that you include in **DOSPATH**.

When the UNIX System shell looks for programs in a directory that is in the **DOSPATH**, it searches for files ending with a **.bat**, **.com**, or **.exe** extension. If these files are found, they are executed as programs in the MS-DOS environment. Programs with the same name will be searched in the following order:

- **.bat**
- **.com**
- **.exe**

DOSPATH Example

The following example of a directory hierarchy shows how to use **DOSPATH**.



The root directory has two subdirectories named **dira** and **dirb**. Here, **dira** contains an MS-DOS word processing program in its **wp** subdirectory and a UNIX System database application in its **db** subdirectory. Also, **dirb** contains two subdirectories. The **ss** directory holds a spreadsheet application and **fun** holds a game.

To run the MS-DOS word processing application, set the **PATH** and **DOSPATH** variables to:

```
PATH=./dira/wp
DOSPATH=/dira/wp
```

The **PATH** variable gives the search order, which, in this case, is first, the current directory (.) and then **/dira/wp**. Here, Simul-Task 386 only searches for MS-DOS applications within the **/dira/wp** directory, because that is the only directory value of **DOSPATH**. For more information on running MS-DOS application programs see "Running MS-DOS Programs"

The **VPIXCNF** Variable

VPIXCNF specifies the path name for the Simul-Task 386 configuration file. This variable lets you

- have multiple Simul-Task 386 configuration files stored in different directories
- provide an alternative name for the default **vpix.cnf**

Search Precedence

Simul-Task 386 searches for the configuration file in the following sequence:

1. For a **VPIXCNF** environment variable. If an environment variable is set, but Simul-Task 386 cannot find or open the file, Simul-Task 386 issues a warning message.
2. Next, the **\$HOME/vpix/vpix.cnf** file.
3. Finally, for **/usr/vpix/defaults/vpix.cnf**. If not found, Simul-Task 386 generates a fatal error message - **error opening configuration file**, and terminates.

Example **VPIXCNF**

Simul-Task 386 first looks at the value of the **VPIXCNF** shell environment variable. For example,

```
VPIXCNF =/usr/vpix/myconfig
```

Here, the file **myconfig** is the Simul-Task 386 configuration file. This file overrides any configuration information within a **\$HOME/vpix/vpix.cnf** file.

The VPIXTERMDIR Variable

VPIXTERMDIR specifies the path name of the directory that contains a terminal specification file that overrides the default. This variable lets a user redefine terminal characteristics when activating Simul-Task 386.

NOTE

The corresponding /usr/lib/terminfo files *must* be on the system or Simul-Task 386 won't work properly on that terminal even though the /usr/vpix/term files (described in the "Override Order" section that follows) are present. **terminfo** files are on the Remote Terminal Package for UNIX System V/386 Release 3.2.

Override Order

If the environment variable **TERM** is set to **att605**, and the environment variable **VPIXTERMDIR** is set to **/usr/you/vpixterm**, then the files that Simul-Task 386 would scan, in order, would be:

File	File Description
/usr/lib/terminfo/a/att605	standard UNIX System terminfo database for an att605.
/usr/vpix/term/sys.default	Simul-Task 386 terminal-independent specification file.
/usr/vpix/term/att605	Simul-Task 386 terminal specific file with the name set in the environment variable TERM.
/usr/you/vpixterm/att605	optional Simul-Task 386 terminal specification override file with the file name of the value of TERM and a directory path name of VPIXTERMDIR.

Note that whether or not Simul-Task 386 locates a file, it still searches for the other files. A definition in a subsequent file overrides a previous file's definition.

So if the standard att605 definition does not define keys necessary for an MS-DOS application, the sys.default in the /usr/vpix/term directory might suffice. If not, there may be a specific set of modifications in /usr/vpix/term for

att605. In any event, VPIXTERMDIR gives the path name of an additional terminal specification file.

Enabling Peripheral Devices

This section describes entries within the **vpix.cnf** file for the following devices:

- printers
- mice
- diskettes
- fixed disks
- serial ports

within the **vpix.cnf** file.

Printers

MS-DOS devices **LPT1**, **LPT2**, and **LPT3** are available MS-DOS printers. They must be assigned some UNIX System command. To do so, add or uncomment an entry in the **vpix.cnf** file. For example,

```
LPT1  pr | lp
LPT2  /usr/bin/lp -Plj1
LPT3  /usr/bin/lp -Plj2
```

NOTE

The printer flush option on the Simul-Task 386 menu or **doscmd flush** sends any buffered data to the printer. (MS-DOS does not have a printer close operation.)

Mouse

Simul-Task 386 supports the AT&T Bus Mouse (3-button) which can be shared among virtual terminals. (Virtual terminals are described in Chapter 5.)

If you are using the AT&T Bus Mouse (3-button) under Simul-Task 386, the following entry should be added or uncommented in the **vpix.cnf** file:

```
MOUSE      /dev/mouse
```

NOTE

Since Simul-Task 386 always emulates a Microsoft Bus Mouse, MS-DOS applications must be configured for the Microsoft Bus Mouse and not for the physical AT&T mouse.

Diskettes

Simul-Task 386 supports the following types of diskettes:

- 5¼" 360KB double density
- 5¼" 1.2 MB high density
- 3½" 360 KB double density, single-sided
- 3½" 720 KB double density, double-sided
- 3½" 1.44 MB high density

Simul-Task 386 also supports pseudo-volume diskettes, which are really a UNIX System files. MS-DOS drives **A:** and **B:** could be assigned in the **vpix.cnf** file. For example,

```
A: /dev/rdisk/f0t    (using a physical diskette drive)
B: /usr/vpix/defaults/B: (a diskette image assigned to Drive B:)
```

NOTE

When specifying a physical diskette drive, you must use the raw device, **/dev/rdisk**, or some copy-protection mechanisms may cause a read failure.

Pseudo-Volume Fixed Disks

The most common type of pseudo-volume is an exact sector-by-sector image of a PC-type fixed disk stored as a single UNIX System file. The **C:** drive is an example of a pseudo-volume. The pseudo-volume can be any UNIX System file or block device.

MS-DOS allocates available disk sectors sequentially. The UNIX System file will grow as the pseudo-volume stores more information.



When files are removed from pseudo-volumes, the size of the pseudo-volume is unchanged, and free space is not reused during the current session.

By placing the appropriate entries in the **vpix.cnf** file, you can make the pseudo-volume a private virtual disk (exactly like using any stand-alone PC's Drive C:).

The only drives that can be designated as pseudo-volumes are A:, B:, C:, and D:. Normally, A: is configured as a physical device; B: is not configured; and C: and D: are configured as pseudo-volumes. If all four of these drives were to be configured as pseudo-volumes, A: and B: would be pseudo-diskettes, and C: and D: would be pseudo-fixed-disks.

Serial Ports

COM1 and **COM2** are MS-DOS names for serial ports. You must assign these devices to UNIX System serial devices by adding or uncommenting an entry in the **vpix.cnf** file. For example,

```
COM1 /dev/tty00
```

or

```
COM2 /dev/tty01
```

During a Simul-Task 386 process, Simul-Task 386 owns these devices unless you release them by selecting "close serial" option from the Simul-Task 386 menu.

An alternative method of releasing or assigning serial ports is to use the **doscmd** command, described in "Using Simul-Task 386."



The UNIX System cannot use the serial port at the same time as MS-DOS.

In order for Simul-Task 386 to access serial ports, the following conditions must be met:

- No other process can be using the device.
- Users must have read/write permission on the device.

If these conditions are not met, the message, **Error attempting to open serial device**, is displayed on the screen. To ensure access, the port must be configured through FACE as type **NONE**. The procedure is as follows:

1. If **getty** or **uugetty** is using the port, stop these processes, using FACE:
 - a. From the *System Administration* menu, select *Peripherals Setup*.
 - b. From *Peripherals Setup*, select the *Serial Ports Setup* menu.
 - c. From *Serial Ports Setup* window, select *Device Type NONE*.
2. Exit from FACE.
3. The permission for `/dev/ttyxx` is changed by this procedure to 666 (read-write), thus allowing Simul-Task 386 to use the port.

To restore **getty** or **uugetty** when you are finished with Simul-Task 386, do the following:

1. Invoke FACE again, and go to the *Device Type* menu (as described above). Choose **MODEM** to restore **uugetty**. Choose **TERMINAL** to restore **getty**.
2. Exit from FACE. The appropriate process (**getty** or **uugetty**) is respawned, and the permission on `/dev/ttyxx` is changed to 622.

Direct Device Access

The DDA mechanism, in cooperation with the Virtual Device Interface (VDI) driver, allows MS-DOS applications/device drivers to directly access a hardware device within the Simul-Task 386 environment without a specific UNIX system device driver and an IEM.

Most hardware devices designed to run under MS-DOS on AT-class systems are sold in conjunction with MS-DOS device drivers or special MS-DOS programs that interface directly with the hardware. It is possible to tailor the

Simul-Task 386 Environment so that the MS-DOS driver or program can directly access the hardware, without requiring any emulation support from the ECT or UNIX system drivers.

In general, an AT-class hardware device can be uniquely described by specifying:

- the I/O registers specific to the device (if any)
- the controller memory locations specific to the device (if any)
- the interrupt line (IRQ number) used by the device to interrupt the CPU (if any).
- the DMA channel used by the device (if any).

In order to define a device to the system using DDA, these device characteristics must be known. This information can usually be found in documentation provided with the hardware, or can be obtained by consulting the hardware manufacturer. Most AT-class hardware devices utilize interrupts to communicate with the CPU, and contain I/O registers for transferring information to and from the device. Devices that contain controller memory are generally limited to display devices and network cards.

NOTE

The specification of DMA channels is not currently supported by DDA, and consequently devices that utilize DMA are supported under Simul-Task 386 only if Emulation Modules for them are available.

The DDA facility supports a variety of AT-class hardware on a "drop-in" basis. Its main advantage is that it requires no software modification and comes with a simple front end for defining devices. Its drawback is that Simul-Task 386 must own the device exclusively when it is being used; consequently devices cannot be shared by multiple users.

NOTE

DDA may not lend itself well to time critical devices because it does not guarantee timely servicing of interrupts.

To define a new direct-access device the system administrator must define the device's characteristics.

Once a device has been defined to the system using the DDA facility, its access is controlled by the presence of its logical device name in a Simul-Task 386 configuration file (**vpix.cnf**), just as with standard emulated devices. If a DDA device is encountered in this file during startup of a Simul-Task 386 process, it is looked up in **/usr/vpix/etc/vpixdevs**. The device information contained in this file is used by the ECT to call the (VDI) special driver in the UNIX system kernel. The VDI is responsible for providing the ECT with direct access to the device, and for guaranteeing that the device is not being used by another user. The ECT, in turn, allows the Virtual 8086 to access the device directly.

The DDA mechanism provides a means by which a MS-DOS device driver or program can receive interrupts from a device. It is important to understand that even though the MS-DOS device driver or program is given direct access to the non-standard hardware device, it still cannot directly access the 8259A interrupt controllers on the motherboard. As noted earlier, emulation of standard devices such as the 8259 is not configurable, and cannot be removed. This means that when the UNIX system kernel receives a real interrupt from the physical 8259 interrupt controllers, this must be transmitted to the MS-DOS program and be made to appear as a real interrupt to the Virtual 8086 process. With the DDA mechanism, this task is accomplished in several steps as follows:

1. The device causes a real interrupt on the 8259 Interrupt Controller.
2. The VDI's interrupt routine is automatically invoked by the UNIX system kernel.
3. The interrupt routine sends a pseudo-interrupt to the Simul-Task 386 process that opened the device.
4. The ECT receives the pseudo-interrupt, and generates a virtual interrupt via the built-in 8259 emulation in Simul-Task 386.
5. The MS-DOS driver or program receives what appears to be a real interrupt, and services the device. Because the MS-DOS program has complete access to the device, it can service the interrupt correctly without need for any emulation. On completion, the MS-DOS driver issues an EOI instruction.

6. The ECT detects the EOI instruction issued by the Virtual 8086 task and turns off the virtual interrupt on the emulated 8259.

DDA Installation and Removal

The Simul-Task 386 Installable Emulation Module Tools Package contains the following items necessary for support of DDA:

- VDI UNIX System device driver
- DDA support tools

VDI UNIX System Device Driver

DDA device support requires the installation of the VDI device driver into the base system. The VDI device driver allows direct access of I/O ports, and controller memory of the DDA devices by MS-DOS programs.

DDA Support Tools

To aid the installation, removal, and modification of DDA devices, three programs are provided: **ddainstall**, **ddachange**, and **ddaremove**. These DDA Support Tools are installed in `/usr/vpix/etc` and have links into the `/usr/bin` directory.

ddainstall

The **ddainstall** program interactively allows you to enter necessary information. The necessary information consists of the logical name and device parameters. Whereas the logical name must be given by you, at a minimum, only one of the device parameters must be supplied.

The **ddainstall** program will print out error messages for input that is invalid, or when in conflict with a built-in device.

The following scenario and accompanying messages and queries specify the **ddainstall** program's functionality.

When the you type **ddainstall** on the console, the following introduction is displayed.

ddainstall allows your DOS device drivers and application programs to use an add-on board without the support of an emulation module.

You may install DDA devices with conflicting device parameters, and in order to use the installed DDA device, the VPIX.CNF file must be updated.

Please consult your Simul-Task 386 User's Guide for more information.

Strike ENTER when ready
or ESC to stop.

When you strike the key, the introduction screen clears and you are queried for the logical name of the DDA device.

Please enter the logical name of the DDA device you want to install. Lower case letters will be converted to upper case letters.

Blank input or ESC terminates this program.

The logical name entered by you will be used to check for conflicts with existing devices (Built-in, IEM, DDA). If a conflict is found, you will be presented with the following screen, otherwise, the installation shall continue with the next step. The screen clears and you see:

The logical name <logical name>
is already installed.

If you would like to enter another
device name, then continue.

If you would like to change parameters for
<logical name>, then stop, and run
the "ddachange" program.

If you would like to remove the <logical name>,
then stop, and run the "ddaremove" program.

Strike Enter when ready
or ESC to stop

If you strike the **Enter** key, then you are prompted for another name, otherwise the screen is cleared, the following message is displayed, and you exit.

`ddainstall command canceled`

Using the logical name entered, the following screen will displayed:

DDA Device Installation for <logical name>

SELECT ONE OF THE FOLLOWING:

1. I/O Range
2. Cntrl Mem Range
3. Interrupt Request Number
4. Exit (Install the DDA device)
5. Cancel (Cancel the installation and exit)

Enter Selection:

When 1 is chosen, you will see the CHANGE PARAMETER MENU as follows:

```

                                CHANGE PARAMETER MENU
Logical   IEM      Node      IRQ      DMA      Cnfg
Name     Name     Name      val      val      Flag
=====  =====  =====  =====  =====  =====
TAROUND  taround  /dev/ta  0xffff   0xffff   Y

                Entry      I/O ranges      Cntrl mem ranges
                =====  =====
                1          0x24c           0x24e
    
```

SELECT ONE OF THE FOLLOWING:

- | | |
|-----------------------------|-------------------------------|
| 1. Modify I/O range. | 4. Modify DMA channel. |
| 2. Modify Cntrl Mem range. | 5. Toggle configuration flag. |
| 3. Modify interrupt vector. | 6. Exit this menu. |

Enter Selection:

>From here, you see the I/O ranges and choose to modify them if desired.

When 2 is chosen from the Main Menu, the CHANGE PARAMETER MENU is displayed. >From here, you see the Controller Memory ranges and choose to modify them if desired.

When 3 is chosen from the Main Menu, the CHANGE PARAMETER MENU is displayed. >From here, you see the Interrupt Request Number and choose to modify it if desired.

When 4 is chosen from the Main Menu, the DDA will be installed. The **vpixdevs** file will be updated to include the new DDA device. In addition, a special device file will be created in the **/dev** directory with the same name as the logical name of the DDA except in lowercase. The default **vpix.cnf** file is also modified to include the DDA logical name.



If you have a private **vpix.cnf** file you will have to manually add the logical name to this file.

When 5 is chosen from the Main Menu, the installation will be canceled and no changes will be made. nothing and exit as if the ddainstall has never been executed.

ddachange

The **ddachange** command is a link to the **ectconfig** command. The **ectconfig** command is an interactive command that allows the privileged user to modify the configuration of IEM devices. After entering **ddachange**, the following menu will appear:

```

                                     Installed IEM/DDA Devices
=====
MOUSE          TAROUND

SELECT ONE OF THE FOLLOWING:

    1. Display next page
    2. Change an entry
    3. Exit (Update the vpixdevs and re-link dos)
    4. Cancel (Exit without update).

Enter Selection:
```

When 1 is chosen, the upper display shows another screenful of IEM/DDA device logical names that could not be displayed, if there are any. If not, the display stays the same. The logical names are displayed in the order they appear in the **vpixdevs** file.

When 2 is chosen, you are prompted for a logical name for the device.

When 3 is chosen, and the information has been changed, then the program exits, the results would be written out to the **vpixdevs** file, and **ectbuild** will be invoked.

When 4 is chosen, then the program exits without changing the **vpixdevs** file.

ddaremove

The **ddaremove** command prompts for the DDA device's logical name and deletes it from the **vpixdevs** file. It removes the entry from the default **vpix.cnf** file and removes the special file device node.

When you type **ddaremove** on the console, the following screen is displayed.

```
Please enter the logical name of
the DDA device you want to remove.
Lower case letters will be converted
to upper case letters.

Blank input or ESC terminates this program.
```

If the DDA name is not installed, then the screen clears and the following is displayed, where <DDA Name> is what you entered:

```
The <DDA Name> is not installed.
```

```
To specify another DDA device for  
removal, continue. Otherwise, stop.
```

```
Strike Enter to continue  
or ESC to stop
```

When you strike the key, the prompting for the DDA device is repeated as above.

If the DDA name given is not a DDA device, then the screen clears and the following is displayed, where <DDA Name> is what you entered:

```
The <DDA Name> is not a DDA device.
```

```
To specify another DDA device for  
removal, continue. Otherwise, stop.
```

```
Strike Enter to continue  
or ESC to stop
```

When you strike the key, the prompting for the DDA device is repeated as above.

Chapter 4: Simul-Task 386 Fundamentals

Simul-Task 386 Fundamentals	4-1
Before You Begin	4-1
Logging In	4-2
Logging Out	4-2
Changing Your Password	4-2
System Prompts	4-3
Privileged Users	4-3
Basic Concepts	4-4
What Applications Can You Run?	4-4
The Importance of Your .profile:	4-5

Simul-Task 386 Fundamentals

This chapter is intended for a user who is new to the UNIX System and to Simul-Task 386. If you are a new or casual user and you will mainly be running MS-DOS applications, you can probably get most of the information you'll need from this chapter and the following chapter.

This chapter tells you:

- How to log in
- How to log out
- How (and why) to change your password
- What to expect from Simul-Task 386
- How to see whether a particular software package has been installed
- How to run a software application

Before You Begin

The way you go about the tasks listed above depends on the interface software on your system. This book describes each procedure for systems with no interface software, for systems using FACE, and for systems using the XWIN System.

The first time you log in, the system will be new to you. Before you start:

- Do you know your login name?
- Do you know your password?
- Do you know whether FACE has been installed?
- Do you know whether the XWIN System has been installed?
- If you aren't at the console, do you know what kind of terminal you are using and what the UNIX System calls that terminal?
- Have you read "Notational Conventions" in Chapter 1 of this book?

Remember, if you get stuck on an unfamiliar word or concept or on a technical term, Appendix C is a glossary.

Logging In

Logging in requires the same information from you whether you have interface software or not, whether you are at the console or at your terminal. The only difference is that you will see the **Console login:** prompt if you're at the console. You will see the **Login:** prompt if you are at a terminal—if it's turned on; if not, turn it on.

At the **Login:** prompt, enter your login name. (In this book, "enter" means to type something and then press <Enter> which is your Enter or Return key.)

Now you see the **Password:** prompt. Enter your password. You will not see your password on the screen. This is a security measure to ensure that your password remains private, known *only* to you. It also means that you have to type carefully because you won't see a typing error. If you should make a mistake, the system will repond **Login incorrect** and again prompt you for your login name.

That's all there is to it. At this point, you are on the system. What you see depends on your system interface software. The *AT&T FACE User's Guide* and the *AT&T XWIN Graphical Windowing System User's Guide* contain screen illustrations.

Logging Out

Logging out is extremely simple. Depending on your system interface, you select "Exit" from a menu or you press <CTRL>-<D>. (Entering the word "exit" is equivalent to <CTRL>-<D> in most situations.) If you select Exit from the FACE menu, you will be asked to confirm the fact that you really want to log out.

Changing Your Password

In the interest of system security, you should change your password from time to time. You can do this from the UNIX System shell prompt (\$) by entering **passwd**. You are then prompted to enter your new password. (Remember, you won't see it, so type carefully.) You are asked to enter the password twice, to ensure against inadvertent typing errors. If the two entries don't match, you are prompted to start over.

Depending on your interface software, you may be able to select "Administration" or "Change Password" to change your password. Whichever method you use, you will be asked to enter the new password twice.

System Prompts

As an MS-DOS user, you are probably accustomed to seeing the A>, B>, and C> prompts on your screen, indicating that MS-DOS is ready for your next command. UNIX System users expect the dollar prompt (\$) or, for **root**, the pound sign (#). Simul-Task 386 has several system prompts, depending on where you are and what you are trying to do. Don't let this confuse you. If you understand what each prompt means, the prompts are actually helpful because you can tell at a glance whether you're in the right place.

The following are prompts you may see in the course of a Simul-Task 386 session:

NOTE

Prompts can be changed. If your system prompts seem to be completely different from the ones on this list, check with other users to see whether your system prompts have been changed.

\$	UNIX System
#	root
U:	UNIX System, accessed through Simul-Task 386
H:	Your home directory tree, accessed through Simul-Task 386

Privileged Users

A privileged user can read, write, and execute any file on the system. The person who installed Simul-Task 386 and gave you access to Simul-Task 386 is a privileged user. If you're a general user, you may not have these privileges.

Super user privileges are often referred to as **root** privileges because the system administrator's login name on a UNIX System is always **root**. This is not the same as the top directory of the UNIX System file system.

These two distinct meanings of the word "root" are not meant to confuse you—although that sometimes happens. When you see the word "root," remember that it can mean:

- the login of a privileged user, and thus refer to a person
- the top of the file system (UNIX operating system or MS-DOS), and thus refer to an all-encompassing directory

Note that / (slash) and \ (backslash) always refer to the file system.

Basic Concepts

Simul-Task 386 acts within the UNIX operating system as a kind of arbiter between the UNIX System and MS-DOS, enabling the UNIX operating system to interpret and execute MS-DOS commands and to run MS-DOS applications. Because the UNIX System is multiuser and multitasking, you and the other users on your 6386 WGS can share resources.

Since each user will not have his or her own floppy disk drives, printer, and communication ports, Simul-Task 386 allows sharing of these physical devices among MS-DOS sessions. For example, the files on the UNIX System are shared among many Simul-Task 386 and UNIX System users. UNIX System files are accessible to Simul-Task 386 users through the H: and U: drives.

In addition, Simul-Task 386 users can utilize "pseudo-volume" drives for copy-protected MS-DOS programs that require file systems that mirror the physical PC/AT-type fixed disks.

Pseudo-volumes are described in more detail in "Using Simul-Task 386."

What Applications Can You Run?

If you're a UNIX System pro, you can simply select the UNIX System and check the contents of /usr/dosapps. You will see a U:> prompt, and you can then list MS-DOS applications with the UNIX System command **ls** or the MS-DOS command **dir**. (Appendix D is a table of MS-DOS commands and the corresponding UNIX System commands.)

An easier method, especially if the UNIX System is new to you, is to select *Programs* from the *FACE Office of* menu. You then see a list of installed software.

The Importance of Your .profile:

Every user on your system has a **.profile** (pronounced "dot profile") file in his or her home directory. The **.profile** is automatically executed by the system every time you log in. It establishes information that the system needs, and it allows you to express certain preferences. Specifically, the **.profile** tells the system what kind of terminal you are using. Without this information, your screen may look jumbled.

If you ever log in at a terminal other than the console, you will be prompted to tell the system about your terminal. (This means you have to know what the system calls the terminal.)

If you should change terminal types permanently, you should change your **.profile**.

You can change your terminal type for one session by entering the following two lines:

```
TERM=termname  
export TERM
```

(**TERM** is the name by which the UNIX System knows your terminal; **export** means "not just for the next command, but for my whole session.") The uppercase letters are important. The UNIX System sees no similarity whatsoever between **TERM** and **term**, so enter those lines exactly as shown, substituting the name of your terminal for *termname*.

Your **.profile** has another couple of lines that are crucial for running MS-DOS applications: the line that sets your **PATH** variable, the line that sets your **DOSPATH** variable, and the line that **exports** them. These are automatically set during the installation process. These can be changed at any time by editing your **.profile**.

There is just one more thing you may have to know before you start. Some MS-DOS applications require certain files to be present in the directory from which you are running the application. Nothing will happen if you try

Simul-Task 386 Fundamentals

to run such applications from the wrong directory. You won't break anything, but you won't run anything, either. You'll see an error message if this happens. Check with an experienced user before proceeding.

Chapter 5: Running MS-DOS and UNIX System Applications

Running MS-DOS and UNIX System Applications	5-1
Running MS-DOS Programs with Simul-Task 386	5-1
Running MS-DOS Programs from the UNIX System	5-1
Prerequisites	5-2
Example	5-2
Operational Explanation	5-2
Avoiding Problems	5-4
Piping and Redirection	5-5
Running More Than One Task	5-6
VTM On the Console Display	5-6
Creating Virtual Terminals	5-7
Background Console Tasks	5-7
VTM On a Serial Terminal	5-8
Running UNIX System Commands From MS-DOS	5-8
Example	5-9

Running MS-DOS and UNIX System Applications

As soon as the Simul-Task 386 installation process is complete, you can start using Simul-Task 386. You can work at the console or at a terminal. You can use any of the UNIX System facilities (commands, programs, utilities), any of the MS-DOS facilities, and all applications software that has been installed on your system. This chapter is organized to get you up and running as quickly and easily as possible. Once you are comfortable with the basic operations of Simul-Task 386, you may want to explore further—by reading and by trying things out.

Running MS-DOS Programs with Simul-Task 386

One of the easiest ways of running an MS-DOS applications is to use the FACE menu system, and simply select the application you want to run.

- From the FACE *Office of* menu, select *Programs*.
- Select the application.

If you are not using FACE:

- Change directories to the appropriate directory.
- Enter the name of the application.

Running MS-DOS Programs from the UNIX System

You can also run an MS-DOS application from the shell prompt. The shell environment variables set in your **.profile** establish the environment so you can run an MS-DOS application in your current directory directly from the UNIX System. All that is required is to have the MS-DOS program's path in your **DOSPATH** as well as in your **PATH** variable. Then you type the application name at the UNIX System shell prompt. MS-DOS program names are those files with the extensions **.exe**, **.com**, or **.bat**.

Prerequisites

The MS-DOS application is run by executing it from **RUNDOS.COM**, a special MS-DOS loader program. This must be in the MS-DOS **autoexec.bat** file, automatically setup when Simul-Task 386 is installed.

Many MS-DOS applications require that several other support programs be accessible to MS-DOS in the current directory. Therefore, change to the appropriate directory before invoking the application.

Example

Assume that the MS-DOS program Lotus 1-2-3 resides in the UNIX System directory `/usr/dosapps/123` as **lotus.com**. Let us also assume that both **DOSPATH** and **PATH** contain `/usr/dosapps/123`.

The Simul-Task 386 installation added the shell environment variable **DOSPATH** to your **.profile** file and exported it.

To run this application from the UNIX System, type the following at the UNIX System shell prompt:

```
cd /usr/dosapps/123
lotus
```

The first line entry changes your directory to the one in which Lotus 1-2-3 files reside. The second line executes the MS-DOS program **lotus.com** within that directory.

NOTE

You can alter the **PATH** and **DOSPATH** variables to run MS-DOS programs from other directories.

Operational Explanation

There is an underlying mechanism that enables the retrieval of an MS-DOS program, as in the above example. This mechanism preserves the environment for an application.

A look at the environment's underlying operation will help explain what actually occurs when you run Lotus 1-2-3 from the UNIX System.

DOSPATH specifies the directories that the UNIX System shell has to search for an MS-DOS executable file (if the search does not find the named file under the UNIX System). The directories specified in the shell variable **DOSPATH** must be a subset of the directories specified in the shell variable **PATH**.

For example, if you try to execute the command **dosprog** from the UNIX System environment, the shell first searches for the file **dosprog** in each directory specified in the shell variable **PATH**. If the shell finds a UNIX System file by that name, it runs that file as a UNIX System executable program. If it does not find a file in a directory specified in the shell variable **PATH**, it sees if that directory is also specified in the shell variable **DOSPATH**. If that directory is not specified in the shell variable **DOSPATH**, the shell continues this search process with the next directory specified in the shell variable **PATH**.

However, if the same directory is also specified in the shell variable **DOSPATH**, then the shell searches for equivalent MS-DOS executable file names in that directory.

In our example, the shell searches for the files **dosprog.bat**, **dosprog.com**, and **dosprog.exe** (in that order) in the directory specified in the shell variable **DOSPATH**. If the shell finds a file that matches the specification in the **DOSPATH** directory, it then runs the command **dos -c** and passes to it as parameters whatever you typed on the command line. (The **dos** command and its options are described in Chapter 6, "Using Simul-Task 386.")

The file **AUTOEXEC.BAT** is executed as part of booting MS-DOS. This is similar to what happens when MS-DOS is booted on any machine. The last line executed in **AUTOEXEC.BAT** is **rundos.com**. (This is all set up automatically when Simul-Task 386 is installed.) When **rundos.com** executes, it determines that this was a fast load with the intention to execute an MS-DOS command. It runs the MS-DOS command processor with the **-c** option, gets the command line that you typed, and then passes it as parameters to the command processor. The MS-DOS command that you typed now executes in the MS-DOS environment. When the command execution is complete, the entire sequence returns to the UNIX System shell prompt. Note that all of this happens without your having to do anything except invoke the MS-DOS program.

Avoiding Problems

In the Lotus 1-2-3 example above, you type the MS-DOS command (lotus) at the UNIX System shell prompt. It is important to understand that the UNIX System shell prompt interprets commands and performs expansion differently from the MS-DOS command processor.

For example, the UNIX System directory separator / is the MS-DOS switch character. Therefore, if you type:

```
chkdsk/f a:
```

to execute the MS-DOS command **CHKDSK.COM** from the UNIX System, the UNIX System shell will search the file **f** in the directory **chkdsk**. This is not what you intended. To eliminate this problem, add a space before the UNIX System directory separator and type:

```
chkdsk /f a:
```

The UNIX System shell expands wildcard characters, such as the asterisk (*) and the question mark (?) and uses the backslash (\) as an escape character to process a command line before executing the command. However, the MS-DOS command processor uses the backslash as a directory separator and leaves the expansion of the wildcard characters to MS-DOS.

This difference between UNIX System and MS-DOS operation can cause problems if you try to execute the MS-DOS command **REN**, for example, from the UNIX System to rename files specified with wildcard characters.

For example, if you type:

```
ren \src\*.c \src\*.b
```

the UNIX System shell expands the string ***.c** into all the files that match that description in the current directory. It then passes the file names on as parameters to the MS-DOS command processor through the mechanism described in the preceding subsection.

If more than one file is specified by ***.c**, the MS-DOS rename fails. Furthermore, the shell expunges the backslash (\) from the directory path with the result that your target directory becomes **src*.c**. However, **src*.c** is not the intended target name.

To prevent the UNIX System shell from expanding wildcard characters and interpreting the backslash as an escape character, you have to use single quotation marks:

```
ren '\src\*.c' '\src\*.b'
```

This command works because the UNIX System shell passes on the wildcard characters and the slash to the MS-DOS command processor.

Piping and Redirection

From either the UNIX System or MS-DOS, you can pipe and redirect UNIX System and MS-DOS processes within the same command line.

When combining these processes, UNIX System commands can be first and last in the pipeline. **A UNIX System command can have a pipe (|) on either side of it; the UNIX System command cannot have pipes on both sides of it.** For example, the following command line, which uses the MS-DOS sort utility, is valid:

```
ls -la /usr/ron/vpix | sort | pg > LPT1:
```

The following command line is invalid:

```
ls -la /usr/ron/vpix | grep -i "*.H" | pg > LPT1:
```

The first command line uses an MS-DOS sort utility; the second has the UNIX System **grep** utility enclosed by pipes.

Unless you are using absolute (full) path names:

- All UNIX System executable files must reside in directories specified within the PATH environment variable.
- All MS-DOS executable files must reside in directories specified within the DOSPATH environment variable as well as the PATH variable.

When using MS-DOS and UNIX System commands in the same pipeline, the **ADDCR** and **RMCR** filters may be necessary (refer to the "Utilities" section for more information on **ADDCR** and **RMCR**). When piping from within MS-DOS, you must have permission to create files in the root directory of your current drive in order to pipe. Thus, your current drive cannot be U: unless you are **root**.

Running More Than One Task

If the XWIN System is installed on your system, you can use it to execute simultaneous tasks; that is, you can start a task (for example, you can calculate a Lotus 1-2-3 spreadsheet), and then start a second task, a third task, and so on. Running simultaneous tasks in XWIN windows is described in Chapter 8, "Using Simul-Task 386 with XWIN."

You can also run multiple simultaneous tasks with the Simul-Task 386 Virtual Terminal Manager (VTM). The VTM creates a new virtual screen for each task that is run. One of these tasks runs in the foreground; the others run in the background.

You can use the VTM from the console or from a serial terminal; however, it works slightly differently, depending on your type of display device:

On the console display:

You can have eight simultaneous tasks: one runs visibly (in the foreground); the others run as background tasks. You can bring a background task to the foreground.

On a serial terminal:

You can stack as many UNIX System or MS-DOS tasks as memory allows. However, you cannot switch between or among them. A task returns to the foreground only when the immediately preceding foreground task stops.

NOTE

The AT&T 3-button mouse can be used between virtual terminals.

VTM On the Console Display

Each task uses a different virtual screen. Only one task can be the current or foreground session.

The console screen displays the virtual screen belonging to the current task. Keyboard activity applies to this task, which can be a UNIX System or MS-DOS task.

Key sequences switch between tasks. These are:

key sequence	meaning
<ALT> - <SysRQ> <N>	Next Virtual Session
<ALT> - <SysRQ> <P>	Previous Virtual Session
<ALT> - <SysRQ> <F1>	VT01
<ALT> - <SysRQ> <F2>	VT02
<ALT> - <SysRQ> <F3>	VT03
<ALT> - <SysRQ> <F4>	VT04
<ALT> - <SysRQ> <F5>	VT05
<ALT> - <SysRQ> <F6>	VT06
<ALT> - <SysRQ> <F7>	VT07
<ALT> - <SysRQ> <H>	Console

Creating Virtual Terminals

You create virtual tasks (sessions) with the Simul-Task 386 Menu. (The Simul-Task 386 Menu is fully described in Chapter 6, "Using Simul-Task 386.")

From a Simul-Task 386 session, press <ALT>-<SYS RQ> <M> <U> (for run (U)NIX System shell). This spawns a new shell and creates a virtual terminal for it. The new shell can invoke another MS-DOS session (type **dos** again) or a UNIX System task.

You can return to the initial MS-DOS session by pressing a key sequences, such as

<ALT> - <SYSRQ> <P>

or

<ALT> - <SYSRQ> (<F1> to <F7> and <H>).

Background Console Tasks

Background tasks (those that do not currently own the screen) are given an area of virtual memory that looks to the background task like the actual display buffer. Background programs continue to execute even if they require direct I/O to the video device. Background programs can also perform:

- disk I/O
- printer output
- serial I/O

NOTE

Background sessions cannot receive mouse or keyboard input until moved to the foreground. Background sessions executing EGA graphics applications are also suspended.

VTM On a Serial Terminal

On a serial terminal, the menu option **(U)NIX System shell** lets you place your initial MS-DOS task in the background and spawn a shell that runs Simul-Task 386 or a UNIX System command. The MS-DOS task is suspended, not terminated, while the UNIX System task is running.

You can spawn as many shells as you want. However, you cannot switch between background and foreground tasks. A new task is pushed onto a stack of tasks.

A background task returns to the foreground when the immediately previous task stops. If you do not want the foreground task to run to completion, you can stop it by typing **exit** or **<CTRL> - <D>**

Running UNIX System Commands From MS-DOS

Simul-Task 386 lets you execute any UNIX System command from the `/bin` and `/usr/bin` directories. (The exceptions are the commands with names that are longer than 8 characters. There are 8 such commands: **captoinfo**, **displayenv**, **displaypkg**, **INSTALL.USR**, **installdos**, **installpkg**, **REMOVE.USR**, and **removepkg**. The **rununix.exe** program within `/usr/vpik/unixbin` enables this feature.

During installation, the names of all UNIX System programs in `/bin` and `/usr/bin` are mapped to equivalent MS-DOS executable names. These are then linked to **rununix.exe**.

For example, the name of the UNIX System program **ls** creates the equivalent MS-DOS executable name **ls.com**. **ls.com** is then linked to the MS-DOS program **rununix.exe**. You can mimic the Simul-Task 386 installation from the UNIX System if you type:

```
cd /usr/vpix/unixbin
ln rununix.exe ls.com
```

These entries create the file **ls.com** in the directory `/usr/vpix/unixbin`. When you execute Simul-Task 386 from the installed setting, your MS-DOS path contains the directory `U:\usr\vpix\unixbin`. Therefore, to list the directory from where Simul-Task 386 was executed, type the following command at the MS-DOS prompt:

```
ls -al
```

NOTE

The switch parameters must be specified in the same way as under the UNIX System.

When you type **ls**, the MS-DOS program **rununix.exe** is executed under the name **ls.com** because of the UNIX System link. The program *ls* executes in the same manner that it would execute if you had typed **ls** at the UNIX System prompt.

Example

The user Bob wants to run his UNIX System program **myprog** from Simul-Task 386. This program resides in Bob's home directory `/usr/bob`. As part of the user-specific Simul-Task 386 installation for Bob, the directories **vpix** and **vpix/unixbin** are created in Bob's home directory. The MS-DOS file **rununix.exe** is copied into the **vpix/unixbin** directory. Furthermore, each time Bob executes Simul-Task 386, the directory `H:\vpix\unixbin` is automatically set in his MS-DOS path.

Thus, all Bob needs to do to run his UNIX System program from Simul-Task 386 is to link an equivalent MS-DOS executable name to the MS-DOS program **rununix.exe**. This linking can be done either from the UNIX System or Simul-Task 386.

To link from either the UNIX System or Simul-Task 386, Bob would type:

```
cd /usr/bob/vpix/unixbin
ln rununix.exe myprog.exe
```

Notice that you must use the UNIX System directory separator and switch character when running UNIX System commands from the Simul-Task 386 environment. The UNIX System command **ln** must have its source and target in the same file system. In this example, it must be the /usr file system.

The derived MS-DOS executable name for the particular UNIX System program must be in your MS-DOS path so that you can execute that UNIX System command from Simul-Task 386.

In the example above, if you want the MS-DOS executable name **ls.com** to reside on Drive **A:**, then type the following command from Simul-Task 386:

```
copy U:\usr\vpix\unixbin\rununix.exe a:\ls.com
```

If you want the MS-DOS executable name for the UNIX System program to reside on a different file system, then copy the MS-DOS program **rununix.exe** to the target name rather than linking it (see the **ln** command in the *User/System Administrator's Manual*).

Chapter 6: Using Simul-Task 386

Using Simul-Task 386	6-1
Basic Considerations	6-1
Installing MS-DOS Applications	6-2
The Simul-Task 386 Command Line	6-3
Example	6-4
The Simul-Task 386 Menu	6-4
Menu Options	6-6
Command Line Options: doscmd	6-7
System File Access From Within MS-DOS	6-8
The UNIX System Virtual Disk (U:)	6-9
The \$HOME Virtual Disk (H:)	6-9
Differences Between Drives H: and U:	6-10
An Example of H: and U: Usage	6-10
Troubleshooting	6-11
File Name Mapping	6-12
Mapping Algorithm	6-12
Examples	6-13

Using Simul-Task 386

This chapter explains

- concepts
- installing MS-DOS applications
- the Simul-Task 386 command line
- the Simul-Task 386 menu (and *doscmd*)
- access to UNIX System files from MS-DOS, including
 - virtual disks
 - file name mapping

Basic Considerations

- The Simul-Task 386 cross-execution environment allows you to run MS-DOS programs from within the UNIX System and UNIX System programs from MS-DOS.
- Simul-Task 386 can store MS-DOS files inside a single contiguous UNIX System file. This UNIX System file is called a **pseudo-volume** and looks to MS-DOS like a disk drive. These pseudo-volumes contain files that the UNIX System cannot directly manipulate. *Use caution when executing UNIX System commands that manipulate UNIX System files.*
- When you run a UNIX System program from Simul-Task 386, the UNIX System program inherits the Simul-Task 386 environment, which includes the current directory. So if you execute Simul-Task 386 from your home directory and change your current MS-DOS directory, a forked UNIX System process inherits your home directory as its current directory.
- Changing the directory setting under MS-DOS within Simul-Task 386 does not filter its way to the UNIX System environment, unless you change directories on **U:** or **H:** virtual disk.

Installing MS-DOS Applications

MS-DOS applications ordinarily come on MS-DOS formatted diskettes. You can use FACE to install these applications, or you can invoke Simul-Task 386 and install them as you would under MS-DOS. If the application has an INSTALL utility, you can use that. You must be able to write to the C: drive and to the drive on which you are installing the application. This means that you may have to ask your users to release the D: drive while you install the application.

If you use FACE, you must be a privileged user. FACE has automated procedures for installing, administering, and removing MS-DOS applications.

From the *Software Setup* menu, select *Install MS-DOS Software* and follow the prompts. For more detailed information, consult the AT&T FACE User's Guide.

If you don't use FACE, you must be **root**. To copy the application from diskette to fixed disk, use the MS-DOS **copy** command (Simul-Task 386 must be loaded) or follow the application manufacturer's instructions for using an INSTALL utility.

Note that some INSTALL utilities modify, or instruct you to modify, the AUTOEXEC.BAT and CONFIG.SYS files. In cases where AUTOEXEC.BAT and CONFIG.SYS do not exist in the root directory of the current drive (e.g., D:, E:, U:), they may be created.

NOTE

All changes to AUTOEXEC.BAT and CONFIG.SYS must be made in the root directory of the C: drive (i.e., C:AUTOEXEC.BAT and C:CONFIG.SYS).

The following guidelines may help you to decide where to install an application:

- C: should contain only MS-DOS system files (COMMAND.COM, AUTOEXEC.BAT, and CONFIG.SYS). Applications software should not be installed on this drive.
- D: is a pseudo-volume used only for installing MS-DOS applications which are copy protected and "expect" to be installed on a true MS-DOS format hard disk (i.e., the applications write

"fingerprint" information to areas of the disk which are not accessible through normal MS-DOS commands).

- E: is a regular UNIX System directory, /usr/dosapps, used for installing all other MS-DOS applications which are to be made available to all Simul-Task 386 users.

NOTE

If you begin installation and then discover that you don't have write permission on C: or D:, abort the installation and ask the other users to release D:.

The Simul-Task 386 Command Line

You can select Simul-Task 386 options from the command line by specifying switches (flags). These switches override **vpix.cnf** (the configuration file) and shell environment variables. The **dos** command is used in the following manner to start an MS-DOS session:

```
dos [ options ]
```

where *options* are:

- b** Boot from the drive A: diskette. If you do not specify **-b**, the system will boot from C:, and if there is no diskette in the A: drive, the system will attempt to boot from C:.
- c command [command_args]** Executes *command* after loading Simul-Task 386. This option provides a quick way to boot by telling Simul-Task 386 to fast load from a saved bootimage (i.e., the **-c** option uses a saved bootimage from a previous **-s** option if there is one). Single quotes must enclose any *command_args* with wildcard operators. When **-c** is specified, it must be the last option specified.
- f** Starts in full screen mode from an XWIN window when MS-DOS is invoked from XWIN.

- h** Help. Describes the optional switches.
- n** No prompting (you don't have to press the Enter key to continue). Appropriate for batch files.
- r [bootimage]** Restores the *bootimage* file previously saved with the **-s** switch. This option provides faster loading of MS-DOS. If *bootimage* is not specified, it uses the entry in **vpix.cnf** (\$HOME/vpix/vpix.img).
- s [bootimage]** Saves a file (*bootimage*) containing a Simul-Task 386 image for subsequent fast loading. If *bootimage* is not specified, it uses the entry in **vpix.cnf** (\$HOME/vpix/vpix.img).

Example

The command:

```
dos -c wp.com '/R textfile'
```

uses the saved Simul-Task 386 image and loads the wp application.

NOTE

To avoid shell file name expansion (special characters for regular expressions) use single quotation marks, rather than double quotation marks.

The Simul-Task 386 Menu

Simul-Task 386 has a control menu which can be invoked during the Simul-Task 386 session by pressing the **<ALT> - <SYSRQ> <M>** key sequence. (Press **<ALT>** and **<SYSRQ>** simultaneously, release, and then press **<M>**.)

NOTE

Certain other computer keyboards use the **<SYS Req>** key in place of the **<ALT> - <SYSRQ> <M>** sequence.

Refer to the keyboard mapping section to determine which key will invoke the Simul-Task 386 Menu on a serial terminal which does not have a <SYS Req> key.

The following is a sample Simul-Task 386 Menu. Each menu option is explained in the following subsection.

NOTE

When the screen is in graphics mode, the menu is active but not displayed.

Simul-Task 386

Simul-Task 386

(E)scape menu
(F)loppy release
(P)rinter flush
(Q)uit DOS
(R)eset DOS
(S)ound is ON
(C)lose serial
(U)NIX System
(Z)oom OUT

Note that the (C)lose serial option appears only if COM1 or COM2 is defined in the **vpix.cnf** configuration file. The (Z)oom option appears only if you are running in an XWIN window. Simul-Task 386 also displays a device list; for example,

Device	Assignment
A:	dev/rdisk/f0t
C:	/usr/vpix/defaults/C:
D:	/usr/dosapps/D:
COM1:	/dev/tty00
LPT1:	pr usr/bin/lp
LPT2:	pr sort /usr/bin/lp

Terminal att605	Assignment /dev/tty01
--------------------	--------------------------

Menu Options

- | | |
|------------------|--|
| (Esc)ape menu | Clears the menu and returns you to the current MS-DOS session. This is the default. |
| (F)loppy release | In a multiuser system, tasks sharing diskette drives require a locking mechanism. A Simul-Task 386 task automatically acquires the diskette drive when an MS-DOS application reads or writes to it. When one application has acquired the diskette drive, subsequent requests result in a drive not ready message. (For a complete list of error messages, see Appendix B.) To allow other Simul-Task 386 sessions to access the device, the owning session must use the (F)loppy release function. |
| (P)rinter flush | Sends any buffered printer data to the printer. (MS-DOS has no close printer operation which the UNIX System requires.) |
| (Q)uit DOS | Terminates the MS-DOS session. |
| (R)eset DOS | Reboots Simul-Task 386 to the virtual PC environment by performing a "warm boot." It functions like the <CTRL> - <ALT> - keyboard sequence on a stand-alone PC. This restarts the MS-DOS session only and does not reboot the UNIX System. |
| (S)ound is ON | Sound emulation on the serial terminal is translated into a series of beeps. These could become distracting during some applications or games. You can toggle this option on or off. |

- (C)lose serial In a multi-user system, tasks sharing serial devices require a locking mechanism. A Simul-Task 386 task automatically acquires the COM device when an MS-DOS application reads or writes to it. When one application has acquired the COM device, subsequent requests result in a message indicating that the device is unavailable. To allow other Simul-Task 386 sessions to access the COM device, the owning session must use the Close serial function. This function closes COM1 and COM2 at the same time.
- (U)NIX System Allows you to switch to a new UNIX System shell without terminating the MS-DOS program.
- (Z)oom OUT When your MS-DOS session is invoked through the XWIN System, this option toggles between running on full screen (OUT) and in a window (IN).

NOTE

To select an option, enter the first letter of the option. Options may also be selected by positioning the cursor on the option with the arrow keys and striking <Enter>.

Command Line Options: doscmd

You can select Simul-Task 386 menu options from the MS-DOS command line with the syntax:

```
doscmd option [option_argument]
```

where **doscmd help** displays the syntax message:

```
{C}ommunications [Off | Release | On | Acquire]
    Acquires/Releases serial ports (default Release)
{Flo}ppy [Off | Release | On | Acquire]
    Acquires/Releases Floppy Drives (default Release)
```

{Flu}sh	Flushes the Simul-Task 386 Printer Buffer
{P}rinter	Flushes the Simul-Task 386 Printer Buffer
{Q}uit	Terminate the Current Simul-Task 386 Session
{Reb}oot	Reboot the Current Simul-Task 386 Session
{Res}et	Reboot the Current Simul-Task 386 Session
{Se}rial [Off Release On Acquire]	Acquires/Releases serial ports (default Release)
{Sh}ell	Fork a New Command Shell
{So}und [Off Release On Acquire]	Toggles/Sets Sound Capability (default Toggle)
{Z}oom [In Out]	Zooms in to XWIN window or out to full screen

The characters within braces indicate the minimal (unique) name of the *option* specification.

System File Access From Within MS-DOS

Simul-Task 386 can map UNIX System files to MS-DOS files on a one-to-one basis. This permits transparent access from any MS-DOS application without modifications to the MS-DOS program. To map files, use the Redirector utilities

- DOSMOUNT.EXE
- REDIR.EXE

The Utilities section describes these programs. This section explains necessary concepts, which are

- the UNIX System virtual disk (U:)
- the \$HOME virtual disk (H:)
- file name mapping
- troubleshooting

The \$HOME and UNIX System disks can be any letter otherwise unused for disk drive names. **H:** and **U:** are conventions in this manual. By default, Simul-Task 386 understands H: or Y: for the \$HOME disk and U: or Z: for the UNIX System disk. Drive names A: through D: are reserved for MS-DOS disk drives.

You can change these disk letter designations within the AUTOEXEC.BAT file by using the **DOSMOUNT** utility.

The UNIX System Virtual Disk (U:)

You can access all UNIX System files through a virtual disk called the **U:** disk. The U: disk drive enables access to the entire UNIX System directory hierarchy. Drive U: is available when you activate the Redirector (with the REDIR and DOSMOUNT commands).

- The root of the U: directory hierarchy is the UNIX System root (/) directory.
- The current working directory of U: is set to what the current working directory was immediately before U: became active.
- UNIX System access privileges apply to drive U: files.

The \$HOME Virtual Disk (H:)

You can access all UNIX System files **in and below the home directory** through a virtual disk called the **H:** disk drive. Drive H: is available when you start a Simul-Task 386 session.

- The UNIX System root of the H: directory hierarchy is your home directory (the value of the \$HOME variable).
- The UNIX System current working directory of H: is your home directory; the MS-DOS current working directory is \ (the MS-DOS backslash root).
- UNIX System access privileges apply to drive H: files.

The Simul-Task 386 user installation script adds

```
H:\unixbin
```

to the MS-DOS search path. This lets you automatically execute UNIX System programs from MS-DOS. (The MS-DOS search path is in your AUTOEXEC.BAT file, the location of which is set in your vpix.cnf file).

The H: drive lets you store MS-DOS files in your \$HOME directory hierarchy and lets you set the MS-DOS search path to H:*directory*.

Differences Between Drives H: and U:

You set the U: virtual disk to the current UNIX System directory with an MS-DOS **cd** command. You can change to any accessible directory. The H: virtual disk only permits changing to a directory below \$HOME.

The root of H: is set permanently to your \$HOME directory. The root of U: is set to the root (/) of the UNIX file system.

Virtual disk drives U: and H: are set up by the execution of **DOSMOUNT.EXE** in the **AUTOEXEC.BAT** file.

Do not remove this program from the **AUTOEXEC.BAT** file unless you do not want virtual drives.

To summarize the differences between U: and H:

Virtual Drive	UNIX System root Directory	Current Directory of Drive at Startup
U: (Z:)	/ (slash)	UNIX System current directory
H: (Y:)	\$HOME	\
E:	/usr/dosapps	\usr\dosapps

NOTE	Drive U: is mapped to Drive Z: and Drive H: is mapped to Drive Y: by Simul-Task 386. You can use DOSMOUNT to perform this type of mapping.
-------------	---

An Example of H: and U: Usage

Given that the Simul-Task 386 user **joe** is currently running at the UNIX System shell prompt. His current directory is **/usr/joe/work**. His home directory, **\$HOME**, is **/usr/joe**.

If Joe enters a Simul-Task 386 command line, he gets into the MS-DOS environment on drive U:. If his prompt includes the current directory, it is

U:\usr\joe\work

Joe's \$HOME directory is `/usr/joe`. Drive **H:** would be set to `\usr\joe`. If Joe performed a **DIR** command on drive H: by typing:

```
U:\usr\joe\work > DIR H:
```

he would see a listing of the files in the `/usr/joe` directory.

If Joe were to change his drive to the drive **H:** directory by issuing the MS-DOS command:

```
U:\usr\joe\work > H:
```

the prompt, which displays the current directory, would be **H:>**.

Notice that the prompt does not display as **H:\usr\joe >**, because Drive **H:** has been given a different path assignment (using **DOSMOUNT**). You can perform other useful drive substitutions using **DOSMOUNT**, which the Utilities chapter describes.

Troubleshooting

Sometimes an application program cannot find files on the U: or H: drive.

The U: Drive is a valid drive under MS-DOS. However, some applications use only the most frequently used disk drives, such as drives C: through H:. If a problem occurs when running an application on drive U: or H:, use **DOSMOUNT** to change the drive designation.

NOTE

Pick a drive that is not used by any MS-DOS devices.

Loaded RAM disks occupy MS-DOS drives.

If you have an available disk drive at Drive **F:**, issue the command:

```
DOSMOUNT F:
```

This command directs all file requests from Drive **F:** to the UNIX System files normally accessible from Drive **U:**. You change the current drive by typing **F**. Then make the directory containing the application the current directory.

File Name Mapping

Simul-Task 386 maps all UNIX System file names to MS-DOS file names.

MS-DOS file names have the format:

name.*[ext]*

where *name* is 1 to 8 characters long. The extension (*ext*) is optional, and if present, from 1 to 3 characters long. The dot (.) is always implied at the end of the name and *must* be explicit if an extension is specified.

MS-DOS has problems with UNIX System file names because:

- UNIX System file names can be longer than MS-DOS file names
- the dot (.) character can appear anywhere in a UNIX System file name, multiple times
- UNIX System file names can mix uppercase and lowercase characters
- UNIX System file names can have illegal MS-DOS characters

To overcome these restrictions, Simul-Task 386 maps all illegal characters and UNIX System names to acceptable MS-DOS names.

Mapping Algorithm

- File name mapping uses the first 5 characters of the original file name, if possible.
- All letters are forced to uppercase.
- Any illegal MS-DOS characters are replaced by a tilde (~) character.
- Any legal MS-DOS extension is appended.
- When a file name undergoes a mapping other than lower to upper case, Simul-Task 386 assigns a 2-character index immediately before the explicit or implied dot character.
- When a file name is shortened, Simul-Task 386 precedes the index characters with a tilde.

NOTE

This algorithm always yields a name with a tilde (~) character replacing any illegal character. This lets you do wildcard searches.

File name mapping is performed dynamically. It generates a unique name

based on the order in which mapped names are requested. Consequently, the same UNIX System file name can map to different MS-DOS file names in different situations.

Do not build mapped names into MS-DOS applications. The first five characters remain the same; however, the sixth, seventh, and eighth characters of the mapped file names will vary.

Examples

The following table shows examples of name mapping. In the following examples, *XX* can represent any 2 characters generated by the file mapper.

UNIX System file name	is mapped to ...	explanation
abc123.def	ABC123.DEF	Lower to upper case.
a	A	Lower to upper case.
A	A~~~~XX	Upper case requires mapping. Notice the 2-character index at the end of the mapped name.
a_long_name	A_LON~XX	Mapped due to length. Simul-Task 386 inserts the tilde before the 2-character index to indicate the shortened name.
AB.c	AB~~~~XX.C	Mapped using available legal characters.
Ab.C	AB~~~~XX.C	The value of the 2-character index will differ from the immediately preceding example. Simul-Task 386 avoids overwriting files by changing the index value during mapping.
a.b.c	A~B~~~XX.C	Too many dots. Notice that the suffix maps to the .C extension.
abcd.efgh	ABCD~~XX	Requires mapping because the suffix is greater than 3 characters.
.login	~LOGI~XX	A leading period (.) always makes a name illegal. This file would map to an MS-DOS "hidden" file.

Chapter 7: Utilities

Utilities	7-1
Converting Text Files: ADDCR, RMCR	7-1
Syntax	7-2
Example	7-2
Displaying UNIX System Directories: XDIR	7-3
Syntax	7-3
Example	7-3
Enabling Remote Disks: REDIR	7-4
The Redirector	7-4
Assigning Remote Disks: DOSMOUNT	7-5
Syntax	7-5
Getting at MS-DOS Partitions from Simul-Task 386: DOSSLICE	7-7
DOSPATH and UNIXPATH Delimiters	7-8
The EMS Driver	7-8
QUIT	7-9

Utilities

This chapter describes:

- converting text files with **ADDCR** and **RMCR** (and using the equivalent commands, **unix2dos** and **dos2unix**)
- displaying UNIX System directories with **XDIR**
- enabling remote disks with **REDIR**
- assigning names to remote disks with **DOSMOUNT**
- defining the pathname delimiter with **DOSPATH** and **UNIXPATH**
- exiting Simul-Task 386 with the **QUIT** utility

NOTE

The two programs **REDIR** and **DOSMOUNT** make up the Redirector. Both are necessary to access the UNIX file system from within MS-DOS.

NOTE

The statement **lastdrive = Z** must be in the **config.sys** file to access UNIX System files from within MS-DOS.

Converting Text Files: **ADDCR**, **RMCR**

- ADDCR** converts from the UNIX System to MS-DOS (ends every line in a UNIX System text file with a carriage return/line feed combination).
- unix2dos** works exactly like **ADDCR**. (This command is been provided because it may be easier to remember that you are converting from the UNIX System to MS-DOS than to remember that you are adding carriage returns.)
- RMCR** converts from MS-DOS to the UNIX System (removes the carriage return from the end of every line within an MS-DOS text file leaving only the linefeed or newline character).

Utilities

dos2unix works exactly like **RMCR**. (This command is been provided because it may be easier to remember that you are converting from MS-DOS to the UNIX System than to remember that you are deleting carriage returns.)

You run these utilities from the UNIX System (as UNIX System processes).

Syntax

The syntax of the **ADDCR**, **RMCR**, **unix2dos**, and **dos2unix** commands are identical:

```
addcr old_filename new_filename
```

```
rmcr old_filename new_filename
```

where

old_filename is the name of the file to be converted.

new_filename is the resulting file.

NOTE

These utilities can use the standard input and output streams. (You can redirect input or output and use these utilities in a pipeline.)

NOTE

new_filename should not be the same as *old_filename*.

Example

The command:

```
grep "COST" account.dat | addcr > doscost.lst
```

searches the UNIX System file **account.dat** for lines with the string "COST." After adding carriage returns to each line it finds, the set of lines is placed in the MS-DOS **doscost.lst** file.

Displaying UNIX System Directories: XDIR

XDIR displays UNIX System directories. XDIR is useful for displaying UNIX System pathnames after file name mapping.

XDIR displays

- the file name
- the file size in bytes
- the creation date of the file
- the owner's user and group numbers
- access privileges.

Syntax

XDIR [*options*] [*pathname*]

where *options* are:

/D includes only directory entries.

/P displays one screen at a time.

/H includes hidden files.

/? displays this list of options.

and where *pathname* indicates a directory to be listed. The default is the current directory.

Example

The command:

```
XDIR /D
```

displays only directories. Its output will include read/write information and ownership information.

Enabling Remote Disks: REDIR

To initialize the Redirector (to enable remote disks):

- Execute the MS-DOS utility **REDIR.EXE** while in the MS-DOS environment. There are two ways of doing this. Choose whichever seems easier to you:
 - Enter the line **REDIR** in an **autoexec.bat** file. (This is the default.)
 - Enter the **REDIR** command at any MS-DOS prompt.
- Now edit the MS-DOS **config.sys** file to include:

BUFFERS = 99

FILES = 50

FCBS = 25, 25

LASTDRIVE = Z

For MS-DOS version 3.3, also include

STACKS = 0,0

The standard Simul-Task 386 program already has these entries.

NOTE

See the *MS-DOS User Guide* for a detailed discussion of the use of the **autoexec.bat** and the **config.sys** files.

After initializing the Redirector, use the **DOSMOUNT** utility to assign MS-DOS drive letters to the UNIX System directory hierarchy.

The Redirector

The Redirector lets you access MS-DOS files within the UNIX file system from MS-DOS. Once MS-DOS files are stored, you can access them from either MS-DOS or the UNIX System.

From within Simul-Task 386, Drive **U:** is set to the UNIX System file system directory. Writing an MS-DOS file to Drive **U:** creates a file in your current UNIX System directory. You do not have to specify any entries in **vpix.cnf** to use the Redirector.

Simul-Task 386 installation creates an AUTOEXEC.BAT file with these entries. If, however, you are creating your own AUTOEXEC.BAT file, before accessing Drive **H:** (which is set to your \$HOME directory), you must run the programs **REDIR** and **DOSMOUNT**. You can either add an entry to the AUTOEXEC.BAT file or run these programs from the prompt.

Assigning Remote Disks: DOSMOUNT

DOSMOUNT controls the Redirector, which you enable with **REDIR**. **DOSMOUNT** provides access to the UNIX file system from MS-DOS. This utility enables or disables specific drive letters as remote (UNIX System) drives. **DOSMOUNT** also lets you set virtual drives at specific points in the UNIX file system.

You can use **DOSMOUNT** any time the MS-DOS prompt is present. Before executing the **DOSMOUNT** command, the Redirector will not have assigned any MS-DOS Volumes to the UNIX file system.

Syntax

Entering **DOSMOUNT HELP** displays the message:

usage: dosmount d: [-] [path-prefix]

Use **dosmount** to select a DOS drive letter through which you can access the Unix file system.

dosmount d:	will make drive d: a networked drive
dosmount d:\usr\bin	will make d: a network drive and will set the path-prefix for d: to '\usr\bin'
dosmount d:\	will make d: a network drive and will eliminate the path-prefix for d:
dosmount d:-	will restore drive d: as a normal drive
dosmount +r	will enable the redirector (default)
dosmount -r	will disable the redirector
dosmount	will show the redirector status and will list all networked drives

Utilities

`dosmount -hpc` will set the hostname to 'pc'
`dosmount -h` will clear the hostname
`dosmount d:=$ENV` will make drive d: a networked drive and set the current directory to the value of the \$ENV Unix environment variable

Note that you can use shell environment variables as values for paths. So

`dosmount g:$HOME`

sets mount points so that **g:** becomes the home directory. Likewise,

`dosmount g:\=$HOME`

sets the current MS-DOS directory to the value of the \$HOME shell environment variable.

NOTE

You *do not* have to execute the - option before reassigning a drive letter to another path.

Also note that entering **dosmount** without parameters displays a listing (with the characters ==> indicating mount points). For example,

```
Hostname is 'hostname'  
Redirected drives:  
  Drive D: \  
  Drive F: \ ==> '\usr\dice\ect'  
  Drive H: \VPIX ==> '\usr\dice'  
  Drive U: \USR\DICE  
  Drive Z: \USR\DICE\REDIR  
For help on DOSMOUNT, type DOSMOUNT HELP
```

Getting at MS-DOS Partitions from Simul-Task 386: DOSSLICE

It is possible to use physical MS-DOS partitions on your fixed disk from Simul-Task 386, thus allowing files to be shared between native MS-DOS and Simul-Task 386. Basically, this allows the C: or D: drive to be assigned (by means of the **vpix.cnf** file) a raw disk slice which points to a real MS-DOS partition. This partition can then be used equivalently by native MS-DOS and Simul-Task 386.

Once you have established an MS-DOS partition with **fdisk** (either from MS-DOS or from the UNIX System), you must run the **dosslice** command. At the shell prompt, enter

```
dosslice
```

You see the following message:

```
For which disk drive would you like to set up an MS-DOS slice?  
(0 or 1)?
```

Select the desired disk drive.

The **dosslice** command will make sure that everything is set up to get at the MS-DOS partition from the UNIX System, including a slice definition in the **vtoc** and a node file for the slice (i.e., `/dev/rdisk/?S?`).

You will then be asked whether the MS-DOS partition should be public or private. If you choose public, the node will be made readable/writable by everybody and owned by **root**. If you choose private, you will be asked for a user login name; the node file will be made readable/writable by owner only and owned by the specified user.

Finally, **dosslice** will tell you the name of the node file so you can use it in a **vpix.cnf** file.

Decide whether you want to use the partition as the C: drive or the D: drive from Simul-Task 386. Then edit your **vpix.cnf** file(s) to replace the C or D entry with the file name specified by **dosslice**.

The **dosslice** command may tell you that you need to reboot your system in order for the changes to take effect. Otherwise, you're ready to go.

DOSPATH and UNIXPATH Delimiters

The MS-DOS utilities DOSPATH and UNIXPATH let you select either the slash (/) or the backslash (\) to delimit file names within a pathname.

DOSPATH sets the backslash (\) as the pathname delimiter.

UNIXPATH sets the slash (/) as the pathname delimiter.

NOTE

The slash (/) is the default delimiter.

Many MS-DOS applications include batch files that require a backslash as the path delimiter. These applications will not run correctly after you run UNIXPATH.

The EMS Driver

The **EMS** (Expanded Memory Specification) driver is a software interface between Simul-Task 386 and programs such as Lotus 1-2-3. You must make an entry in the CONFIG.SYS file to use the EMS driver (and take advantage of up to 2 MB of expanded memory).

To install the driver, insert the line

device=ems.sys [Sx]

where:

Sx is the size of expanded memory allocated on startup. The *x* is the number of 16K pages.

To configure a 64K space, insert:

device=ems.sys S4

into your **config.sys** file.

NOTE

Simul-Task 386 has a maximum of 128 of these 16K pages.

NOTE

EMS driver installation increases Simul-Task 386 memory requirements. Install only when necessary.

QUIT

QUIT lets you stop a Simul-Task 386 session and return to the UNIX System shell. You can use **QUIT** in an MS-DOS batch file to terminate Simul-Task 386 without manual intervention.

Chapter 8: Using Simul-Task 386 with the XWIN System

Using Simul-Task 386 with the XWIN System	8-1
Starting Simul-Task 386 in an XWIN window	8-1
Zooming In and Out	8-2
Terminating Simul-Task 386 Under XWIN	8-3
Exceptions and Limitations	8-3
Configuring Simul-Task 386 Window Attributes	8-4

Using Simul-Task 386 with the XWIN System

The AT&T XWIN Graphical Windowing System allows you to run several MS-DOS or UNIX System applications at the same time and display them in separate windows on your screen. It provides you with the tools to create the windows you need and to manipulate them by moving them around on the screen, changing their size, and temporarily converting them to icons.

It is possible, for example, to run an MS-DOS based spreadsheet program in one XWIN window, a database program in another, and a business graphics program in yet another XWIN window. At the same time, you could have other XWIN windows running UNIX System applications.

This chapter describes how to use Simul-Task 386 in the XWIN environment. It assumes that you already know how to start XWIN and use the window manager. To learn more about the XWIN System, refer to the *AT&T XWIN Graphical Windowing System User's Guide*.

Starting Simul-Task 386 in an XWIN window

From inside an XWIN window running *xterm*, you start Simul-Task 386 the same way you would under a non-XWIN environment. A new XWIN window will be created for the MS-DOS session to run in. You can then use Simul-Task 386 inside this window in the same way you would use it directly on the console. The "Exceptions and Limitations" section below describes differences in the behavior of Simul-Task 386 when it is run in an XWIN window instead of directly on the console.

There are several ways of starting MS-DOS or an MS-DOS application in an XWIN window: From an *xterm* window, you can enter any of the following:

- **dos**
- **dir**
- **123**

If you are running XWIN on a VDC 750 or other EGA-compatible display, when you start Simul-Task 386 you may see the screen clear. The Simul-Task 386 startup messages may appear on the full screen before they shrink down into an XWIN window. This behavior is described in detail in the following section.

Zooming In and Out

At any time while you are running Simul-Task 386 in an XWIN window, you can "zoom" out to a full-screen virtual terminal and continue execution there. Zooming out to the full screen is useful for running graphics-intensive MS-DOS programs that require very fast screen updates. There are also some operations which can only be performed when zoomed out (see "Exceptions and Limitations").

There are three ways to tell Simul-Task 386 to zoom out:

- Select "(Z)oom OUT" from the Simul-Task menu.
- Type <Alt>-<SYSRQ> <Z>.
- Enter the **doscmd zoom out** command.

Zooming is not allowed when you are running XWIN remotely over a network. If there are no more virtual terminals available, zoom out will fail. While the MS-DOS session is zoomed out, its XWIN window will contain the following message:

```
DOS is in full screen mode
on Virtual Terminal nn
```

where *nn* is the full-screen virtual terminal number that the MS-DOS session is now running in.

From a zoomed-out screen, you can return to the XWIN window by zooming in. There are three ways to tell Simul-Task 386 to zoom in:

- Select "(Z)oom IN" from the Simul-Task menu.
- Type <Alt>-<SYSRQ> <Z>.
- Enter the **doscmd zoom in** command.

You can only zoom in from an MS-DOS session which was started from XWIN. An MS-DOS session started in a full-screen virtual terminal cannot be made to run in an XWIN window. Simul-Task 386 may also zoom out automatically. This happens when EGA-specific operations are attempted. For example, if an application switches to an EGA graphics mode (using INT 10H), Simul-Task 386 will automatically zoom out. Booting or rebooting an MS-DOS session on an EGA display also requires full-screen mode and therefore causes Simul-Task 386 to zoom out automatically.

Terminating Simul-Task 386 Under XWIN

An MS-DOS session running in an XWIN window is terminated by the usual means. When you use the Simul-Task menu "(Q)uit" entry or the **quit** command, the MS-DOS session terminates and the XWIN window disappears. This is also true for full-screen MS-DOS sessions that were originally started from the XWIN environment.

An MS-DOS session in an XWIN window can also be terminated by selecting the "Quit" entry in the XWIN window menu.

Exceptions and Limitations

There are several cases where the behavior of Simul-Task 386 running in the XWIN window is different from its behavior when running in a full-screen virtual terminal. These are listed below:

- **Executing a UNIX System command:**
When you execute a UNIX System command from an MS-DOS session running in an XWIN window, the command does not run in the same window. Instead, a new XWIN window is created, and the UNIX System command is executed within this new window. When the command terminates, its window disappears.
- **Different font size used:**
The default character font size used for MS-DOS running in an XWIN window is not the same as the font size for MS-DOS running in a full screen. The default font size for an MS-DOS session in an XWIN window is 6 by 10 pixels. In an XWIN window, both 40-column and 80-column text modes use the same font size.
- **Color mapping:**
The colors displayed for MS-DOS programs in an XWIN window may not be the same as those which appear in the full-screen mode.
- **EGA graphics modes:**
EGA graphics modes are not supported in the XWIN window. When a mode transition to an EGA graphics mode is detected, automatic zooming-out occurs.

- **No mouse support:**

The mouse is not supported for MS-DOS running inside an XWIN window. Once you zoom out to full screen, you can use the mouse.

Configuring Simul-Task 386 Window Attributes

A number of XWIN window attributes for Simul-Task 386 can be changed from their defaults by editing your **.Xdefaults** configuration file. This is entirely optional. Simul-Task 386 will work fine with the default settings. Each line in the **.Xdefaults** file has an attribute name and its value, separated by a colon (:). The attributes supported by Simul-Task 386 are listed below. For detailed information on *.Xdefaults* see the *AT&T XWIN Graphical Windowing System User's Guide*.

- **dos.font:**

Simul-Task 386 normally uses an XWIN font called **pc6x10**. You can make it use a different font by setting this attribute. Most fonts will not have a full 256-character PC character set, so if you change the font you may notice missing characters.

- **dos.Border** and **dos.BorderWidth:**

These attributes control the window border color and width, respectively. They are *not* used when the window manager is running. The **dos.BorderWidth** attribute is in thousandths of an inch.

- **dos.Geometry:**

This attribute controls the size and/or position of the XWIN window. If this attribute is not set, the default window size will be the size of 25 by 80 characters in the current font. See the *AT&T XWIN Graphical Windowing System User's Guide* for the syntax of a geometry specification.

Chapter 9: Using Serial Terminals

Using Serial Terminals	9-1
Required Features	9-1
Optional Features	9-1
Option Descriptions	9-2
Overriding Terminal Definitions	9-2
Example	9-3
Troubleshooting	9-4

Using Serial Terminals

This chapter describes the list of features required in order for a serial terminal to be used with Simul-Task 386. Optional features are also described. In addition, there is an explanation of the precedence of terminal definitions intended to explain how to override terminal definitions in order to run certain "ill-behaved" applications.

Required Features

- Display of at least 24 rows by 80 columns
- Direct cursor addressing (an escape sequence that allows cursor movement to any row and column)
- An escape sequence for clearing the screen
- Escape sequences for line insertion and deletion, or setting the scrolling region and reverse scrolling (like an AT&T 605)

Optional Features

- A PC-compatible mode that can be entered and exited through escape sequences
- Keys outside the basic ASCII set, such as function and arrow keys
- A 25-line display (although a 24-line display is supported; for example, the AT&T 615 has a 24-line display which can be scrolled up or down to display the 25th line)
- Screen attributes:
 - bold intensity
 - blinking
 - reverse
 - underline
 - invisible

Option Descriptions

<i>PC-compatible mode</i>	A PC-compatible mode that can be entered and exited through escape sequences. This feature is now only available on a few terminals such as the AT&T 605. It is important because it allows the entire PC character set to be displayed, making the terminal as close as possible to a "real" PC console.
<i>Non-ASCII characters</i>	It is inconvenient to provide key mappings for all of the "extra" keys (for example, <SYSRQ>) on a PC keyboard if there are no single keys on the terminal's keyboard that send out sequences of ASCII characters.
<i>A 25-line display</i>	Because most standard terminals have only 24 lines, Simul-Task 386 provides support for "scrolling" the terminal screen up and down to show either the first or last 24 lines of the "true" 25-line screen. The specific keys used for scrolling are configured by the super-user by editing the specification file in /usr/vpix/term.
<i>Screen attributes</i>	This allows applications that use different screen attributes to display correctly. For example, games make extensive use of blinking and word processors rely heavily on bolding and underlining text.

(Appendix A contains more information).

Overriding Terminal Definitions

Simul-Task 386 can run "ill-behaved" MS-DOS applications on a serial terminal. To support this capability, Simul-Task 386 needs certain information about your terminal.

Simul-Task 386 uses the standard shell environment variable **TERM** to find out your terminal type. Simul-Task 386 then searches four different areas for specific information about how to run PC applications on that type of terminal. Simul-Task 386 overrides information found in an earlier source with that from a subsequent source. The sources are:

1. The **terminfo** database (included in the Foundation Set of UNIX System V/386 Release 3.2), where Simul-Task 386 gathers information on escape sequences sent to the terminal. The applicable **terminfo** file *must* be present.
2. The Simul-Task 386 terminal-independent specification file, which is always **/usr/vpixmap/term/sys.default**. This file must be present.

Simul-Task 386 uses this file to gather information on how to translate keyboard input into "scan codes." Scan codes are the values that a PC receives when a key is pressed on the keyboard.

3. The Simul-Task 386 terminal-specific specification file. The name for this file is the same name as that assigned to the **TERM** user environment variable. This file also must be located in the directory **/usr/vpixmap/term**. For example, if **TERM** is set to **att605**, then the terminal-specific file that Simul-Task 386 uses is **/usr/vpixmap/term/att605**. If this file is not present (i.e., the specific **att605** file), then Simul-Task 386 uses the generic specification file, **/usr/vpixmap/term/generic**.

Simul-Task 386 will use these files to see how the non-ASCII keys of the terminal are translated.

4. An optional override file. Like the Simul-Task 386 terminal-specific specification file, the override file must have the same name as that assigned to the **TERM** environment variable. The environment variable **VPIXTERMDIR** specifies in what directory this optional override file is located.

Example

If the environment variable **TERM** is set to **att605**, and the environment variable **VPIXTERMDIR** is set to **/usr/you/vpixmap**, then the files that Simul-Task 386 will scan would be, in order:

Using Serial Terminals

<code>/usr/lib/terminfo/a/att605</code>	Standard UNIX System terminfo data base for an att605 terminal
<code>/usr/vpixmap/term/sys.default</code>	Simul-Task 386 terminal-independent specification file.
<code>/usr/vpixmap/term/att605</code>	Simul-Task 386 terminal specific file with the name that was set in the environment variable TERM
<code>/usr/you/vpixmap/att605</code>	Optional Simul-Task 386 terminal specific override file with the name that was set in the environment variable TERM and in the directory with the name that was set in the environment variable VPIXTERMDIR

The last three files all have the same format, as described in Appendix B.

Troubleshooting

There are three reasons why Simul-Task 386 may not display correct information when you are using the serial terminal:

1. Your **TERM** variable is set incorrectly. The **terminfo** database is present but the Simul-Task 386 terminal directory is not set to the correct terminal type.
2. Your terminal definition is not contained in this database.
3. The **terminfo** database is not installed on the system.

(For a detailed listing of terminal capabilities for which Simul-Task 386 needs definitions, see Appendix B.)

Appendix A: Terminal Descriptions

Terminal Descriptions	A-1
Necessary Terminal Information	A-1
Example Syntax	A-1
Terminal Capabilities	A-2
Screen Attribute Definitions.	A-4
svpix and fvpix.	A-5
swrap and rwrap.	A-5
Special Keys	A-6
pccompat	A-6
ASCII Input to Scan Code Mappings	A-7
Output Character Mappings	A-9
Standard PC Key/Terminal Key Sequence Mappings	A-10

Terminal Descriptions

Simul-Task 386 requires a terminal definition in the **terminfo** database for any terminal type from which a Simul-Task 386 process can run. Simul-Task 386 automatically compiles the definition when activated. This appendix describes the terminal information required by Simul-Task 386 and includes a table showing PC key to terminal key equivalences for the AT&T 615, 620, and 630 and for the "generic" terminal defined in **terminfo**.

Necessary Terminal Information

Simul-Task 386 needs three types of information about a terminal:

- escape key sequences for terminal capabilities
- mappings from ASCII sequences to scan code sequences
- mappings from scan code sequences to ASCII sequences

You can specify all the information Simul-Task 386 needs in the Simul-Task 386 terminal configuration file format. Example configuration files are in the **/usr/vpix/term** directory.

Example Syntax

The following demonstrates syntax for a configuration file.

Terminal Descriptions

* Comments begin with an asterisk (*) and continue
* until the end of the line. Tokens are separated by
* white space. The following line defines a terminal
* capability. The line has the form
* "capname: definition"
* and the definition is terminfo-compatible.

clear: \E[H\$<10>
* ANSI clear screen with 10 millisecond delay
* The following line is an input ASCII to scan code
* mapping definition; it maps the VT100 up-arrow key
* to the PC keyboard up-arrow scan codes.

input: \033-[-A 48-C8
* The following line is a PC output to ASCII sequence
* mapping.

output: \001 0
* This process maps the PC character \001 to an ASCII 0.

Terminal Capabilities

You can specify any capability that the terminal physically has to increase the display efficiency or the accuracy of the PC emulation. The following are typical **terminfo** capabilities that are probably already defined for your terminal.

Capability Description

clear	clear screen and home cursor
ed	clear to end of screen
el	clear to end of line
csr	change scrolling region (vt100)
ri	scroll screen down one line
ind	scroll screen up one line
indn	scroll screen up <i>n</i> (variable number of) lines

ill	insert one line
il	insert lines
dll	delete one line
dl	delete lines
ich1	insert one character
ich	insert characters
dchl	delete one character
dch	delete characters
cup	position cursor
home	move cursor home
cubl	move cursor left
cubl	move cursor right
cuul	move cursor up
cudl	move cursor down

Simul-Task 386 requires additional capabilities as shown in the list below. These capabilities are described in detail in the sections that follow.

NOTE

The first six commands have the same names as corresponding terminfo capabilities; however, they work a little differently. (See "Screen Attribute Definitions.")

<i>capability</i>	<i>description</i>
sgr0	normal mode
blink	blink mode
bold	bold mode

Terminal Descriptions

rev	reverse video mode
invis	invisible mode
smul	underline mode
blbo	blink and bold mode
blr	blink and reverse mode
blu	blink and underline mode
bor	bold and reverse mode
bou	bold and underline mode
blbor	blink, bold and reverse mode
blbou	blink, bold and underline mode
svpix	Simul-Task 386 starting string
fvpix	Simul-Task 386 shutdown string
swrap	enable autowrap mode
rwrap	disable autowrap mode
k_up	up "rocker" sequence for 24-line terminals displaying 25-line screens
k_down	down "rocker" sequence for 24-line terminals displaying 25-line screens
k_refr	refresh sequence
pcompat	PC-compatible terminal

Screen Attribute Definitions.

The capabilities **sgr0**, **blink**, **bold**, **rev**, **invis**, **smul**, **blbo**, **blr**, **blu**, **bor**, **bou**, **blbor**, and **blbou** all define screen attributes that can be toggled off and on. While the first six of these have the same name as a generic **terminfo** capability, they work a little differently. The other capabilities are new to **terminfo** definitions. If the terminal can support these attributes, all of them should be defined in one of the Simul-Task 386 specific definition files.

For example, the differences between the ordinary **blink** (as understood by **terminfo**) and the Simul-Task 386 **blink** is that the former means "turn on blinking without changing any existing attributes." Thus if reverse video were already on, it would remain on. However, the Simul-Task 386 "blink" means "turn on blinking and turn off all other attributes." The same is true of the other twelve definitions: The **terminfo** capability turns on that capability without changing any existing attributes; the Simul-Task 386 capability turns on that capability and turns off all other attributes. On an ANSI terminal, the **terminfo** **blink** is defined as `\E[5m`, while the Simul-Task 386 "blink" is `\E[0;5m`.

svpix and fvpix.

When entering and exiting Simul-Task 386 (or when entering/leaving a spawned shell from Simul-Task 386), **svpix** (start Simul-Task 386) or **fvpix** (finish Simul-Task 386) sequences are sent to the terminal. This procedure lets you specify an arbitrary set of escape sequences to toggle any terminal modes that make sense to the UNIX System but not to Simul-Task 386. When entering or exiting Simul-Task 386, your terminal should:

- Make the cursor a blinking underscore (solid block, or whatever you prefer). This cursor is like the PC cursor.
- Toggle PC-compatible mode if running on a PC-compatible terminal.
- Reset **terminfo** attributes appropriate for the UNIX System.

For example, a configuration file may contain the lines:

```
svpix:  \E'7\E~5$<3000>\Ev!\E.3
fvpix:  \Ev4$<6>\Ed*\EO$<6>\Ee(\E'5)
```

The **svpix** string specifies "make default attribute dim, go into PC-compatible mode, wait 3 seconds, enhance mode on, make cursor a blinking line."

The **fvpix** string specifies "get out of PC-compatible mode, wait 6 milliseconds, set autpage off, set autoscroll off, wait 6 milliseconds, set screen to 24 lines, make cursor a blinking block."

swrap and rwrap.

Simul-Task 386 on a serial terminal works best if the terminal is not in autowrap mode. In autowrap mode, the terminal's cursor automatically moves to the beginning of the next line when a character is written to the last position of a line, and the screen is automatically scrolled up when a character is written to the bottom right corner of the screen.

Terminal Descriptions

If your terminal can turn autowrap mode on and off, you should define the **swrap** and **rwrap** sequences. The **rwrap** sequence, if present, will be sent (before the **svpix** sequence) when Simul-Task 386 starts up, in order to disable autowrap mode. If the **terminfo** capability, **am**, indicates that the terminal uses autowrap mode, the **swrap** sequence will be sent (after the **fvpix** sequence) when Simul-Task 386 exits, in order to re-enable autowrap mode.

If the **rwrap** sequence is not defined, Simul-Task 386 will check the **terminfo am** capability to know whether or not the terminal is in autowrap mode, and act accordingly.

Special Keys

You can define three special ASCII sequences for non-PC-compatible terminals. These sequences correspond to pressing a single key on an incompatible terminal.

The capabilities **k_up** and **k_down** support 24-line terminals. Defining either of these capabilities:

- tells Simul-Task 386 that the terminal has only 24 lines
- defines a sequence that "rocks" the screen up or down, showing either the first or last 24 lines of the true 25-line display.

You can use identical definitions for **k_up** and **k_down**. If you define only one of these, the same key rocks the screen alternatively up and down.

On a 24-line terminal within Simul-Task 386, you should scroll with the screen "rocked up" (with the "25th" line showing). If the screen is rocked down, scrolling will be much slower, and the "25th" line will be invisible.

You can clear messages from the screen with a refresh key. The **k_refr** capability does so.

pccompat

Simul-Task 386 must know whether your terminal is PC-compatible or not, so that it knows whether to expect scan codes or ASCII characters from the keyboard.

Give **pccompat** the value **y** on PC-compatible terminals. Assign the value **n** for normal ASCII terminals.

ASCII Input to Scan Code Mappings

PC-compatible terminals transmit numerical values called scan codes. These scan codes correspond to the position of the key on the keyboard. Separate scan codes indicate when a key is pressed and when it is released ("make" and "break" codes).

On incompatible terminals, key sequences result in ASCII values. There must be a mapping between ASCII input characters and the sequence of scan codes that a PC application expects to receive.

For example, the ASCII **a** key, when shifted, transmits (octal) 101, the ASCII code for **A**. However, a PC application expects to see the (hex) scan codes **2A**, **1E**, **9E**, and **AA** in that order, corresponding to the make code for the shift key, the make and break codes for the **a** key, and the break code for the shift key.

The Simul-Task 386 terminal-independent configuration file `/usr/vpixon/term/sys.default` contains the ASCII-to-scan-code mappings for all single ASCII characters. You do not need to define these mappings elsewhere.

NOTE

If your terminal is PC-compatible, these mapping specifications have no meaning. However, having them specified will not hurt anything.

There are a number of keys on a PC keyboard that do not correspond to any ASCII character:

- the 10 function keys,
- the 3 toggle keys: `<CAPS LOCK>` , `<NUM LOCK>` , `<SCROLL LOCK>` , and
- the nine-key numerical keypad, `<INS>` , `` , `<SYS REQ>` , `<PRT SC>` , - and + on the numerical keypad, `<ALT>` , `<CTRL>` , `<LEFT SHIFT>` , and `<RIGHT SHIFT>` .

These last three, while present on ASCII keyboards, do not transmit a character, but rather modify the character transmitted by other keys. To communicate with all PC applications, it must be possible to "tell" Simul-Task 386 that each of these keys has been pressed. In fact, in the case of `<ALT>` , `<CTRL>` , `<LEFT SHIFT>` , `<RIGHT SHIFT>` , it is necessary for some applications to know separately when they have been pressed (make code) and when they

Terminal Descriptions

have been released (break code).

The terminal configuration file **input:** has lines that allow for arbitrary mappings between sequences of input ASCII characters and sequences of scan codes to translate them to. For example, if your terminal is extremely simple and has no "extra" keys whatsoever, you may decide to map each of the above PC keys to ASCII sequences that start with a **<CTRL>** - **<Z>** and have a second character that is some mnemonic for the PC key.

Part of this mapping might look like:

```
input:  \032-1      3B-BB
input:  \032-2      3C-BC
input:  \032-3      3D-BD
input:  \032->      4B-CB
input:  \032-\032   1D-2C-AC-9D
```

The first of these lines says that when a user types a **<CTRL>** - **<Z>** (octal 032) followed by a **<1>** on an ASCII terminal, it will have precisely the same effect as if he had pressed and released the **<F1>** key on a PC keyboard, since 3B and BB are the make and break codes for the **<F1>** key.

Similarly, the next two lines allow a user to emulate **<F2>** and **<F3>** respectively. The fourth line maps the **<->** key on the PC keyboard (the **<6>** on the numeric keypad) to **<CTRL>** - **<Z>** followed by the greater than sign key on the ASCII keyboard.

Since the first character of any such multiple-character ASCII sequence is now special in that it no longer actually transmits its own, "natural" scan code sequence, we must also provide a sequence that does transmit its natural scan codes. That is what the fifth line above does. It says that typing **<CTRL>** - **<Z>** twice will send the scan codes for the PC **CTRL** make code, the **Z** make and break codes, and the **CTRL** break code.

NOTE

The above ASCII sequences are completely arbitrary, whereas the scan codes correspond specifically to some key that we wish to emulate on the PC keyboard. Therefore, if your ASCII terminal has a number of "extra" keys that naturally transmitted some sequence of ASCII characters (typically starting with **esc**, octal 033), you could conveniently map that ASCII sequence to a corresponding key on the PC keyboard. Again, remember that if you follow this procedure, you must also provide a sequence for actually typing an **esc** character (which is used by many PC applications).

The syntax of an input mapping line is:

input: *ASCII_sequence scan_code_sequence*

The ASCII sequence is a series of ASCII characters separated by dashes (-). An ASCII character is either a single printable character, or an octal representation of a character (a backslash followed by 3 octal digits). Since the actual characters dash, backslash and asterisk are special characters in a terminal configuration file, you must enter them using the octal representation just described if they are actual characters in an ASCII input sequence. The scan code sequence is a series of 2-digit hex scan codes separated by dashes.

A number of sample mappings can be found in the various distributed Simul-Task 386 terminal configuration files located in **/usr/vpix/term**.

Output Character Mappings

To support serial terminals, it is necessary to allow mappings from each of the 256 characters in the PC character set to other characters, since many of these are not present on ASCII terminals. Moreover, even PC-compatible terminals require that control characters, namely octal 000 through 037 and 177 through 377, be somehow escaped if you want to display the actual PC character for that code.

In a Simul-Task 386 terminal configuration file, use **output:** lines to perform this mapping. The default output mappings contained in **/usr/vpix/term/sys.default** simply map every character to itself, except the control characters mentioned above. It maps each of these to **esc** (octal 033), followed by itself (except for 000, which is mapped to a space, since 0 and

Terminal Descriptions

space are identical in the PC character set). This default mapping will work on most PC-compatible terminals.

For non-PC-compatible terminals, you must decide what to map to each of the characters 001 through 037 and 177 through 377, since they are typically not present in an ASCII terminal character set. As with **input:** lines, **output:** lines allow total flexibility. For example, you can map all of the above characters to a period (.), or you can find the visually closest ASCII character to the corresponding PC character (see a PC Reference Manual for pictures of the PC character set). Moreover, if the ASCII terminal has a graphics mode that supplies some of these characters, you can map them to the escape sequence to get into graphics mode, followed by the graphics character, followed by the escape sequence to get out of graphics mode.

While this procedure sounds costly, most of the characters output by a PC application will be regular ASCII characters and the overhead may be acceptable.

The syntax of an output mapping line is:

```
output: character character_sequence
```

The character **sequence** has the same syntax as an ASCII sequence in an input mapping line, as described above (except that the characters \200 through \377 make sense in this context, but not in the other context). The character is also the same, but is a single character rather than a sequence of characters.

Standard PC Key/Terminal Key Sequence Mappings

The following table shows the key sequences on three AT&T terminals and the PC keys to which the key sequences are mapped.

AT&T 615, 620, 630	
PC Key	Terminal Key Sequence
<ESC>	<ESC><ESC>
<Sys Rq>	<F14>
<Pg Up>	<ESC>u
<Pg Dn>	<ESC>d
<End>	<ESC>e
<Ins>	<ESC>i
<PrtSc>	<ESC>p
<Alt> press	<ESC>a
<Alt> release	<ESC>A
<Ctrl> press	<ESC>c
<Ctrl> release	<ESC>C
24-line rock	<F11>
refresh	<F12>

The following table shows the key sequences on the "generic" terminal and the PC keys to which the key sequences are mapped.

Terminal Descriptions

"Generic" Terminals	
PC Key	Terminal Key Sequence
<ESC>	<ESC><ESC>
<Sys Rq>	<ESC>s
<F1>	<ESC>1
<F2>	<ESC>2
<F3>	<ESC>3
<F4>	<ESC>4
<F5>	<ESC>5
<F6>	<ESC>6
<F7>	<ESC>7
<F8>	<ESC>8
<F9>	<ESC>9
<F10>	<ESC>0
<up arrow>	<ESC>^
<down arrow>	<ESC>v
<right arrow>	<ESC>>
<left arrow>	<ESC><
<Home>	<ESC>h
<Pg Up>	<ESC>u
<Pg Dn>	<ESC>n
<End>	<ESC>e
<Ins>	<ESC>i
	<ESC>d
<PrtSc>	<ESC>p
<Alt> press	<ESC>a
<Alt> release	<ESC>A
<Ctrl> press	<ESC>c
<Ctrl> release	<ESC>C
<L-Shift> press	<ESC>l
<L-Shift> release	<ESC>L
<R-Shift> press	<ESC>r
<R-Shift> release	<ESC>R
24-line rock	<ESC>/
refresh	<ESC>,

Appendix B: Error Messages

Error Messages

B-1

Error Messages

This appendix contains a list of Simul-Task 386 error messages, arranged in alphabetical order by first word. Each message is followed by an explanation.

Can't Allocate Memory for Screen Buffer

Unable to allocate memory required for the screen. The possible causes are in `brk(2)` of the UNIX System User Manual. Contact the system administrator. If the UNIX System error is "Not enough space" (12), the system administrator should increase the amount of swap space.

Can't Allocate Memory for Screen Save

Unable to allocate memory required to save the screen. The possible causes are in `brk(2)` of the UNIX System User Manual. Contact the system administrator. If the UNIX System error is "Not enough space" (12), the system administrator should increase the amount of swap space.

Can't execute 'progname'; 'UNIX System error message'

Failed attempt to execute either the file name "progname" given by the environment variable "SHELL" or the file name "/bin/sh". The value "UNIX System error message" indicates the UNIX System error number of the failed exec system call. The possible causes are in `exec(2)` of the UNIX System User Manual. Exit Simul-Task 386 and run the environment variable "SHELL". If "/bin/sh", the standard UNIX System shell, cannot be executed, contact your system administrator immediately. If the UNIX System error is "Not enough space" (12), your system administrator should increase the amount of swap space.

Can't get enough memory for pipe buffer

Unable to allocate memory required to create a pipe between Simul-Task 386 and the UNIX System. The possible causes are in `brk(2)` of the UNIX System User Manual. Contact the system administrator. If the UNIX System error is "Not enough space" (12), the system administrator should increase the amount of swap space.

Can't map memory

Error setting up access to the screen memory. Internal error; call your system administrator.

Can't Open Virtual Terminal; 'UNIX System error message'

Failed attempt to open an AT console virtual terminal. If the "UNIX System error message" is "No such file or directory" (2), then the virtual terminal does not exist. Make sure that all virtual terminals have been properly created. If the "UNIX System error message" is "Permission denied" (13), then the file permissions for this file do not allow it to be read or written. The file must be readable and writable.

Can't query VT manager; 'UNIX System error message'

Failed attempt to access the AT console terminal. The UNIX System kernel may not have the proper AT console driver support. In this case, it is not possible to use the console with this UNIX System kernel with Simul-Task 386.

Can't set virtual terminal mode; 'UNIX System error message'

Failed attempt to initialize the AT console terminal. The UNIX System kernel does not have the proper AT console driver support. In this case, it is not possible to use the console with this UNIX System kernel with Simul-Task 386.

Can't set virtual terminal mode; 'UNIX System error message'

Error using the virtual terminal. Internal error; call your system administrator.

Can't set virtual terminal text mode; 'UNIX System error message'

Failed attempt to change modes on the AT console terminal. The UNIX System kernel does not have the proper AT console driver support. In this case, you can't use the console with this UNIX System kernel with Simul-Task 386.

Can't track memory; 'UNIX System error message'

Error setting up access to the screen memory. Internal error; call your system administrator.

CMD_PARM_ERR

Invalid command line parameter at "hex parameter value".

The Simul-Task 386 program was invoked with an invalid argument "hex parameter value". Invoke **dos -h** to print a usage message for Simul-Task 386. See the Simul-Task 386 Manual page for the correct procedures for executing Simul-Task 386 .

CMD_USE1_ERR_SAVE_ERR_WRITE

Cannot write file for saving virtual PC State.

Write error writing out the Simul-Task 386 image to be saved on the file specified with the '-s' option. This file, used for saving a pre-booted Simul-Task 386 image for a later fast load, cannot be created. Make sure the file system this file is being written to contains enough space, because the Simul-Task 386 image contains over 100K. Also make sure that the ulimit value is large enough to handle a file of this size. See write(2) in the UNIX System User Manual for other causes.

COM_ALLOC_ERR

Error allocating space for serial.

Unable to allocate memory required by support of the COM emulation. The possible causes are in brk(2) of the UNIX System User Manual. Contact the system administrator. If the UNIX System error is "Not enough space" (12), the system administrator should increase the amount of swap space.

COM_CLOSE_ERR

Error attempting to close uart device.

Error closing the terminal representing the COM device. Internal error; call your system administrator. See close(2) in the UNIX System User Manual.

Error Messages

COM_DOS_ERR

Error attempting to set DOS mode.

Failed attempt to set the terminal representing the COM device into a special mode. The COM device is represented by the COM1 or COM2 entry in the Simul-Task 386 configuration file. Modifications necessary for Simul-Task 386 to run on this terminal have not been put into the UNIX System kernel. Cannot use this COM device in Simul-Task 386.

COM_INFO_ERR

Error attempting to send serial information.

Failed attempt to set the terminal representing the COM device into a special mode for Simul-Task 386 . The COM device is represented by the COM1 or COM2 entry in the Simul-Task 386 configuration file. Modifications necessary for Simul-Task 386 to run on the terminal have not been put into the UNIX System kernel. Cannot use this COM device in Simul-Task 386.

COM_INTT_ERR

Error sending serial INT type.

Failed attempt to set the terminal representing the COM device into a special mode for Simul-Task 386 . The COM device is represented by the COM1 or COM2 entry in the Simul-Task 386 configuration file. Modifications necessary for Simul-Task 386 to run on the terminal have not been put into the UNIX System kernel. Cannot use this COM device in Simul-Task 386.

COM_NDOS_ERR

Error attempting to set non-DOS mode.

Failed attempt to set the terminal representing the COM device into a special mode. The COM device is represented by the COM1 or COM2 entry in the Simul-Task 386 configuration file. Modifications necessary for Simul-Task 386 to run on the terminal have not been put into the UNIX System kernel. Cannot use this COM device in Simul-Task 386.

COM_OPEN_ERR

Error attempting to open serial device.

Failed attempt to open the serial device given by COM1 or COM2 in the Simul-Task 386 configuration file. This file contains the input and output to the COM ports. The COM1 or COM2 entry in the Simul-Task 386 configuration file may be set to a file that does not exist. Fix the Simul-Task 386 configuration file, or create the right device. If the "UNIX System error message" is "Permission denied" (13), then either:

- The COM1 or COM2 file "UNIX System file name" is already being used by another user. Only one user may use a COM1 or COM2 pseudo volume at a time.
- File permissions for file "UNIX System file name" do not allow it to be read or written. The file must be readable and writable.

Read or write the file at the shell, for example, with the `dd(1)` command. Other possible causes are in `open(2)` of the UNIX System User Manual.

COM_OREGS_ERR

Error attempting to send to serial port.

Failed attempt to emulate the serial communications port on the terminal represented by the COM device has failed. The COM device is represented by the COM1 or COM2 entry in the Simul-Task 386 configuration file. Modifications necessary for Simul-Task 386 to run on the terminal have not been put into the UNIX System kernel. Cannot use this COM device in Simul-Task 386. Possibly, an error emulating the COM device.

COM_READ_ERR

Error reading serial device.

Error while reading data from the serial device. Could be lost carrier. See `read(2)` of the UNIX System User Manual for a discussion of possible read errors. The application may have input from a serial port with a port number that is not

implemented. The invalid 8086 input instruction should be fixed, even though Simul-Task 386 should probably ignore input from invalid ports.

COM_SET_ERR

Error setting line for serial device.

Error while setting the terminal characteristics (such as the baud rate) from the serial device. Could be lost carrier. See `ioctl(2)` of the UNIX System User Manual for a discussion of possible read errors.

COM_WRITE_ERR

Error writing serial device.

Failed attempt to write data to the serial device. Could be lost carrier. See `write(2)` of the UNIX System User Manual for a discussion of possible write errors. The application may have output to a serial port with a port number that is not implemented. The invalid 8086 output instruction should be fixed, even though Simul-Task 386 should probably ignore output to invalid ports.

CONF_OPEN_ERR

Error "UNIX System error message" opening configuration file "filename".

The Simul-Task 386 configuration file cannot be opened. The Simul-Task 386 configuration file contains the value given by the environment variable `VPIXCNF`, or the string `vpix.cnf`. If the "UNIX System error message" is "No such file or directory" (2), then the configuration file "filename" does not exist. The file `vpix.cnf` must be created, or the environment variable `VPIXCNF` must be changed to refer to the Simul-Task 386 configuration file. If the "UNIX System error message" is "Permission denied" (13), then the permissions to the configuration file `UNIX System filename` does not allow it to be read. The file must be readable. Read the file at the shell, for example, with the `cat(1)` command. Other possible causes are in `open(2)` of the UNIX System User Manual.

Couldn't alloc memory for EGA BIOS; 'UNIX System error message'

Unable to allocate memory required to save the screen. The possible causes are in brk(2) of the UNIX System User Manual. Contact the system administrator. If the UNIX System error is "Not enough space" (12), the system administrator should increase the amount of swap space.

Couldn't get VT mode; 'UNIX System error message'

Error using the virtual terminal. Internal error; call your system administrator.

Couldn't map EGA ROM 'UNIX System error message'

Error setting up access to the EGA ROM. Internal error; call your system administrator.

Couldn't Re-track VRAM

Error setting up access to the screen memory. Internal error; call your system administrator.

Couldn't set VT mode; 'UNIX System error message'

Error using the virtual terminal. Internal error; call your system administrator.

Couldn't unmap EGA ROM; 'UNIX System error message'

Error releasing access from the EGA ROM. Internal error; call your system administrator.

ECT_MAIN_ERR

ECT_main error returned

Internal error; call your system administrator.

EXEC_ERR

Exec failed at "hex value".

Either the file name given by the environment variable 'SHELL' or the file name `/bin/sh` was executed. The value "UNIX System error message" indicates the error number of the failed exec system call. The possible causes are in `exec(2)` of the UNIX System User Manual. Exit Simul-Task 386 and run the environment variable 'SHELL' to make sure that it can be executed. If `/bin/sh`, the standard shell, cannot be executed, contact your system administrator immediately. If the error is "Not enough space" (12), your system administrator should increase the amount of swap space.

FLPY_CLOSE_ERR

Error "UNIX System error message" encountered closing floppy file on drive "drive letter".

Error closing the file representing the floppy A: or B: device. Internal error; call your system administrator. See `close(2)` of the UNIX System User Manual for possible reasons.

FLPY_OPEN_ERR

Error "UNIX System error message" opening file "UNIX System file name" as drive "drive letter".

The pseudo volume "drive letter" for the A: or B: diskette drive cannot be opened. This file is specified by the A: or B: entry in the Simul-Task 386 configuration file. If the "UNIX System error message" is "No such file or directory" (2), then the "A" or "B" file "UNIX System file name" does not exist. It may be that the A: or B: entry in the Simul-Task 386 configuration file is set to a file that does not exist. Fix the Simul-Task 386 configuration file, or create a copy of the A: or B: pseudo volume. If the "UNIX System error message" is "Permission denied" (13), then either:

- The "A" or "B" file "UNIX System file name" is already being used by another user. Only one user may use a A: or B: pseudo volume at a time unless the file is read-only.
- The file permissions of file "UNIX System file name" may not allow it to be read. The file must be readable. Read the file at the shell, for example, with the `od(1)` command.

Other possible causes are in open(2) of the UNIX System User Manual.

FLPY_PHASE_ERR

Error in floppy state transition code. Simul-Task 386 is unable to emulate the input/output commands sent to the floppy. This program cannot be run under Simul-Task 386.

FORK_ERR

Fork failed "UNIX System error message".

Error forking to execute a shell. The value "UNIX System error message" indicates the error number of the failed fork system call. The possible causes are in fork(2) of the UNIX System User Manual. Contact the system administrator. If the error is "No more processes" (11), or "Not enough space" (12), the system administrator should increase the amount of swap space.

General protection fault, cannot emulate instruction

An 8086 process caused a General Protection Fault at the offending instruction. Simul-Task 386 emulates some privileged instructions, but this is not one of them. Possible causes are:

- An instruction that fetched or stored a word at location 0xFFFF (65535) caused the next address to become too large.
- An instruction with an "addr32" (0x67) prefix caused an address that is bigger than 0xFFFF (65535).
- Attempt to execute privileged instructions.
- Attempt to execute an 8086 "POP CS" instruction. This instruction was removed from Intel processors after the 8086, and cannot be emulated in Simul-Task 386.
- Too many instruction prefixes in an instruction.
- Any other reason that might cause a General Protection Fault in a Virtual 8086 Mode process. See the 80386 Programmer's Reference Manual.

Error Messages

Make sure that the offending 8086 program works correctly when running MS-DOS without Simul-Task 386.

Got to signal 'signal number' Catcher, About to Exit Simul-Task 386

The Simul-Task 386 process has received the signal "signal number". See signal(2) for the particular signal in the UNIX System User Manual. These signals can come from one of two sources:

- The UNIX System kill command has been run and sent a signal to the process. The UNIX System allows any process to kill another one, and Simul-Task 386 has been informed that it is being killed.
- An unrecoverable program exception. This is an internal Simul-Task 386 error. Reboot.

HARD_FUNC_ERR

INT 13h Function = "UNIX System function code n".

Failed attempt to input or output to the C: or D: pseudo volume with the "INT 13h" instruction in a way that is not supported by Simul-Task 386. The value of "UNIX System function code n" gives the hexadecimal value of the AH register for the "INT 13h" call. This program cannot currently be run on Simul-Task 386.

HARD_INIT_ERR

Error "UNIX System error message" opening file "UNIX System file name" as drive "drive letter".

The pseudo volume "Drive Z: (U:) or Y: (H:)" for fixed disk C: or D: cannot be opened. This file is specified by the C: or D: entry in the Simul-Task 386 configuration file. If the UNIX System error message" is "No such file or directory", (2), then the C: or D: file "UNIX System file name" does not exist. The C: or D: entry in the Simul-Task 386 configuration file may be set to a file that does not exist. Fix the Simul-Task 386 configuration file, or create a copy of the C: or D: pseudo volume. If the "UNIX System error message" is "Permission denied" (13), then the

permissions of file "UNIX System file name" does not allow it to be read. The file must be readable.

Read the file at the shell, for example, with the `od(1)` command. Other possible causes are in `open(2)` of the UNIX System User Manual.

HARD_SEEK_ERR

`Seek(); lseek()` failed, `errno = "UNIX System error message"`.

Error reading or writing on the C: or D: pseudo volume. The error is "UNIX System error message" . Possible causes are in `lseek(2)` of the UNIX System User Manual.

Invalid color screen mode 'mode value' being set

Simul-Task 386 is unable to emulate the value "mode value" that was sent to the CGA mode select register. This program cannot be run under Simul-Task 386.

INVL_MAGIC

Invalid magic request "hex value".

An 8086 process sent output to a special Simul-Task 386 port (0xFCE0) that is set up for communications with the UNIX System that failed. The hexadecimal value "hex value" indicates the invalid magic request. Do not use this port. Should not occur under normal Simul-Task 386 operation.

IO_READ_ERR_PR_CMOS

Bad read from "port number"

An 8086 process executed an "in" instruction at the port given by the hex value "port number", which is in register DX or the immediate field in the "in" instruction. It is in the range of the CMOS ports (0x70 to 0x7F) but it is not a valid or emulated Simul-Task 386 port. The invalid 8086 input/output instruction should be fixed, even though Simul-Task 386 should probably ignore input/output to invalid ports.

Error Messages

IO_READ_ERR;PR_PAGE

Bad read from "port number"

An 8086 process executed an "in" instruction at the port given by the "hex value" at "port number", which is the port number in register DX or the immediate field in the "in" instruction.

This value is in the range of the CMOS ports (0x70 to 0x7F) but it is not a valid or emulated Simul-Task 386 port. The invalid 8086 input/output instruction should be fixed, even though Simul-Task 386 should probably ignore input/output to invalid ports.

IO_WRITE_ERR;PR_CMOS

Bad write of "hex value" to "port number"

An 8086 process executed an "out" instruction with hexadecimal value "hex value" at port "port number", which is in register DX or in the immediate field in the "in" instruction. This value is in the range of the CMOS ports (0x70 to 0x7F) but it is not a

valid or emulated Simul-Task 386 port. The invalid 8086 input/output instruction should be fixed, even though Simul-Task 386 should probably ignore input/output to invalid ports.

IO_WRITE_ERR;PR_PAGE

Bad write of "hex value" to "hex port number"

An 8086 process executed an "out" instruction with hexadecimal value "hex value" to the port "port number", which is the port number in register DX or the immediate field in the "in" instruction. This value is in the range of the CMOS ports (0x70 to 0x7F) but it is not a valid or emulated Simul-Task 386 port. The invalid 8086 input/output instruction should be fixed, even though Simul-Task 386 should probably ignore input/output to invalid ports.

ISR_ERR

Invalid interrupt service routine number

Simul-Task 386 has improperly emulated an "int 1" (trace trap), "int 3" (breakpoint), or "int 4" (overflow trap) instruction.

Perhaps a very small stack pointer was created. See the 80386 Programmer's Reference Manual for other reasons that might

cause a general protection trap. Make sure that the offending 8086 program works correctly when running MS-DOS without Simul-Task 386.

KBD_INT_ERR

Couldn't set keyboard INTTYPE.

Failed attempt to set the terminal into a special mode. The modifications necessary for Simul-Task 386 to run on the terminal have not been put into the UNIX System kernel. Simul-Task 386 cannot run on this terminal.

line 'line number'; input sequence too long (max; 'nnnn')

An input key mapping in the terminal configuration file has too many input key characters to be mapped from. Only "nnnn" input characters may be used for mapping. Shorten the number of input characters to allow Simul-Task 386 to run on this terminal. If this input character sequence is what a single key on the terminal sends, then it will not possible to provide a mapping for this key on the terminal.

line 'line number'; output sequence too long (max; 'nnnn')

An output character mapping in the terminal configuration file at line "line number" is too long. Shorten the number of output characters "nnnn" to allow Simul-Task 386 to run on this terminal, which may require changing the character that gets generated when the source character is given.

line 'line number'; scancode sequence too long ('nnnn')

An input key mapping in the terminal configuration file has too many scan codes mapped to it. Only "nnnn" codes may be mapped to a key sequence. Shorten the number of scan codes to allow Simul-Task 386 to run on this terminal.

line 'line number' too many mappings (max: 'nnnn')

The term configuration line has too many input key mappings. These are lines which begin with the string "input:" and indicate key sequences to be mapped to scan code sequences. Delete

Error Messages

some of these input key mappings to allow Simul-Task 386 to run on this terminal as only "nnnn" scan code sequences allowed.

line 'line number'; unknown terminal capability 'string'

A line in the terminal configuration file begins with the invalid label "string". The label should be the word "input:", "output:", or the name of a terminal capability. If it is neither of these, fix or delete the offending line.

Map to NULL failed

Error setting up access to a null page of memory. Internal error; call your system administrator.

MOUSE_INIT_ERR

Error attempting to initialize mouse.

Failed attempt to initialize the mouse device. The mouse driver in the UNIX System kernel does not have the proper support for the mouse device. Most likely, it is currently not possible to use this mouse device in Simul-Task 386.

MOUSE_OPEN_ERR

Error "UNIX System error message" opening mouse file "UNIX System file name".

Error opening the mouse device given by MOUSE in the Simul-Task 386 configuration file. This file contains the input and output to the mouse ports. The MOUSE entry in the Simul-Task 386 configuration file may be set to a file that does not exist. Fix the Simul-Task 386 configuration file, or create the right device. If the "UNIX System error message" is "Permission denied" (13), then either:

- The MOUSE file "UNIX System file name" is already being used by another user. Only one user may use a MOUSE device at a time.

- The file permissions of file "UNIX System file name" may not allow it to be read or written. The file must be readable and writable. Read the file at the shell, for example, with the `od(1)` command. Other possible causes are in `open(2)` of the UNIX System User Manual.

NO_TERM_ERR

TERM variable is not set.

For Simul-Task 386 to run on any terminal besides the console, the environment variable **TERM** must be set. No **TERM** environment variable was found. Be sure to set and export the **TERM** environment variable in the shell to the type of terminal that you are using.

No virtual terminals available

Trying to use another virtual terminal for Simul-Task 386, a sub-shell, or another process, and all allocated ones are used. Terminate one or more processes running on the virtual terminals before additional virtual terminals can be used. Use the **ps** command to determine what processes are running on the virtual terminals.

No virtual terminal support; 'UNIX System error message'

Failed attempt to open the virtual terminal on the AT console. The UNIX System kernel may not have the proper AT console driver support. In this case, it is not possible to use the console with this UNIX System kernel with Simul-Task 386.

QCK_REST_ERR

Cannot open image file; Create new image file (`dos -s`).

The bootimage file specified with the `"-r"` option for restoring a pre-booted Simul-Task 386 image cannot be opened. The boot image file probably does not exist. The file permissions may not allow access to the file. Read the file at the shell, for example with the `od(1)` command. The possible causes are in `open(2)` of the UNIX System User Manual.

QCK_REST_ERR_MAGIC

Invalid image file; Create new image file (dos -s).

Error while restoring the Simul-Task 386 image that was specified with the "-r" restore boot image flag. The file is either not a Simul-Task 386 boot image, or is older than the Simul-Task 386 configuration file. The Simul-Task 386 configuration file contains the value given by the environment variable **VPIXCNF**, or is the string **vpix.cnf**. Create a new boot image file with the "-s" flag.

QCK_REST_ERR_READ

Cannot read image file; Create new image file (dos -s).

Read error restoring the Simul-Task 386 image that was specified with the "-r" restore boot image flag. The saved Simul-Task 386 image has probably become corrupted. An uncorrectable disk error may have occurred. Creating a new boot image file with the "-s" flag may correct the problem. See read(2) in the UNIX System User Manual for any other reasons.

QCK_REST_ERR_TERM

Cannot start terminal configuration file.

When restoring a boot image file with the "-r" flag, Simul-Task 386 was unable to find the configuration file. Either the environment variable **VPIXCNF** should be set, or the file **vpix.cnf** should be created. The configuration file could have been found but did not have the necessary permission to be used. The Simul-Task 386 configuration file contains the value given by the environment variable **VPIXCNF**, or is the string **vpix.cnf**.

QCK_REST_ERR_VER

Bad image file version number; Create new image file (dos -s).

Wrong version of file. The boot image is either newer or older than the command **dos**. Create a new boot image file with the "-s" flag.

PRN_CLOSE_ERR

Printer close error.

Error while waiting for the printer daemon process to terminate. The value "UNIX System error message" indicates the UNIX System error number of the failed wait system call. The possible causes are in wait(2) of the UNIX System User Manual. Problem with the Simul-Task 386 program.

PRN_GEN_ERR

General printer error.

Failed attempt to read or write a printer or parallel port with a port number that is not implemented. The invalid 8086 input/output instruction should be fixed, even though Simul-Task 386 should probably ignore input/output to invalid ports.

PRN_OPEN_ERR

Printer open error.

Error executing the printer daemon process given by LPT1, LPT2 or LPT3 in the Simul-Task 386 configuration file. This process outputs to the printer or parallel ports. It represents a UNIX System program that processes the output printer characters, and sends them to their destination. The fork system call has failed because the UNIX System is out of resources. Contact the system administrator. If the UNIX System error is "No more processes" (11), the system administrator should increase the number of UNIX System processes allowed. If the UNIX System error is "Not enough space" (12), the system administrator should increase the amount of swap space.

PRN_PUT_ERR

Printer put error.

Error writing data to the printer, parallel port, or the printer daemon process. Could be a unrecoverable disk error. See write(2) for a discussion of possible write errors.

ROMF_OPEN_ERR

Error "UNIX System error message" opening ROM file "UNIX System filename".

The ROM or EGAROM file "UNIX System filename" cannot be opened. If the "UNIX System error message" is "No such file or directory" (2), then the ROM file "UNIX System filename" does not exist. It may be that the ROM or EGAROM entry in the Simul-Task 386 configuration file is set to a file that does not exist. Fix the Simul-Task 386 configuration file, or provide a copy of the ROM or EGAROM. If the "UNIX System error message" is "Permission denied" (13), then the permissions to the ROM or EGAROM file "UNIX System file name" does not allow it to be read. The file must be readable. Read the file at the shell, for example, with the `od(1)` command. Other possible causes are in `open(2)` of the UNIX System User Manual.

ROMF_READ_ERR

Error "UNIX System error message" reading ROM file "UNIX System file name".

Read error reading in the ROM or EGAROM file specified with the "-r" restore boot image flag. The ROM or EGAROM file has become corrupted. An uncorrectable disk error may have occurred. Create a new copy of the ROM or EGAROM file. See `read(2)` in the UNIX System User Manual for other reasons.

SET_SCAN_A_ERR

Can't set scan mode for AT keyboard.

Failed attempt to set the IBM AT console into a special mode. Modifications necessary for Simul-Task 386 to run on the console have not been put into the UNIX System kernel. Simul-Task 386 cannot run on this terminal.

SET_SCANS_ERR

Can't set scan mode in keyboard.

Failed attempt to set the terminal into a special mode. Modifications necessary for Simul-Task 386 to run on the terminal have not been put into the UNIX System kernel. Simul-Task 386 cannot run on this terminal.

TERM_OPEN_ERR

Error 'UNIX System error message' opening terminal configuration file 'UNIX System filename.'

The terminal configuration file "UNIX System filename" cannot be opened. If the "UNIX System error message" is "No such file or directory" (2), then the terminal configuration file "UNIX System filename" does not exist. One cause may be that the **TERM** environment variable is set to a terminal type that is unknown by Simul-Task 386. Create the terminal configuration file for this terminal type. If the "UNIX System error message" is "Permission denied" (13), then the permissions to the terminal configuration file **UNIX System filename** does not allow it to be read. The file must be readable. Read the file at the shell, for example, with the cat(1) command. Other possible causes are in open(2) of the UNIX System User Manual.

terminal configuration file 'UNIX System file name' has errors

The terminal configuration file given by "UNIX System file name" has a structure defect. Subsequent error messages will elucidate.

Unable to fork new process; 'UNIX System error message'

Error forking to execute a shell. The value "UNIX System error message" indicates the UNIX System error number of the failed fork system call. The possible causes are in fork(2) of the UNIX System User Manual. Contact the system administrator. If the UNIX System error is "No more processes" (11), the system administrator should increase the number of UNIX System processes allowed. If the UNIX System error is "Not enough space" (12), the system administrator should increase the amount of swap space.

Error Messages

v86init() failed; 'UNIX System error message'

The system call used to initialize Simul-Task 386 has failed. If the error is "Not owner" (1), then Simul-Task 386 has not been installed properly. Simul-Task 386 must be setuid to root. If the UNIX System error is "Not enough space" (12), the system administrator should increase the amount of swap space. Other errors may indicate an internal Simul-Task 386 problem.

WAIT_ERR

Wait failed "UNIX System error message".

Error waiting for a sub-shell to terminate. The value "UNIX System error message" indicates the error number of the failed wait system call. The possible causes are in wait(2) of the UNIX System User Manual. Problem with the Simul-Task 386 program.

Warning: couldn't set current directory to *dir_name*

A DOSMOUNT command tried to set the current directory to a non-existent directory.

Warning: couldn't translate environment variable *var_name*

A DOSMOUNT command argument used an invalid shell environment variable.

Appendix C: Glossary

Glossary

C-1

Glossary

This appendix contains an alphabetical list of terms that may be unfamiliar to you. Further information may be found by looking for each term in the Index and checking the index references.

AT-compatible software

Software designed to run on the IBM PC/AT or compatible such as the AT&T 6386 WGS.

background Background and foreground have special meaning under Simul-Task 386. Specifically, "foreground" denotes "displayed on the screen." Conversely, "background" relates to non-viewed applications when they are running. More precisely, background is a type of program execution in which you request the shell to execute a command away from what you are presently doing, that is to say, "in the background." See *foreground*.

bootimage The Simul-Task 386 Bootimage File contains the memory and device state of the DOS process after MS-DOS has booted. The bootimage is later used for quick loading the Simul-Task 386 system and quickly restarting MS-DOS. For example, when you type "123" at the shell prompt, the Simul-Task 386 bootimage is used to start Lotus 1-2-3. Each Simul-Task 386 user can have a bootimage file as specified in the configuration file.

console The console is the PC display and keyboard.

environment The situation while using an operating system such as the UNIX System or MS-DOS. For example, the shell environment includes such components as your shell prompt string, specifics for backspace and erase characters, and commands for sending output from your terminal to the computer.

foreground Background and foreground have special meaning under Simul-Task 386. Specifically, "foreground" relates to viewed and "background" relates to non-viewed applications when they are running. Foreground is the normal type of program execution. In the foreground mode, the shell waits for a command to end before prompting you for another command. See *background*.

Glossary

- hot key A system hot key is assigned to allow users to manage virtual terminals and the DOS process. The system hot key is `<ALT> - <SYSRQ>`. Depending on the key that follows the hot key, users can create, navigate, and zoom in and out of the virtual terminals as well as manage the Simul-Task 386 DOS process.
- ill-behaved applications Application programs that sometimes ignore high-level interfaces. To improve performance, ill-behaved applications take advantage of the unprotected PC environment to manipulate hardware control registers. On larger machines, these registers are accessible only to the operating system; but on PCs they can be bypassed.
- multi-tasking The ability of an operating system to execute more than one program at a time. Multi-tasking is a virtual concept that makes it appear as though the user has an individual PC. The UNIX System is an operating system that allows multi-tasking.
- multiuser The ability of an operating system to support several users at the same time. The UNIX System is a multiuser operating system.
- pipng A method of redirecting the output of any command to be the input of another command. Piping is named for the character (`|`) that redirects the output. For example, the shell command `who | wc -l` pipes output from the `who(1)` command to the `wc(1)` command, telling you the total number of people logged into your multiuser UNIX System.
- pipeline A series of filters separated by the pipe character (`|`). The output of each filter becomes the input of the next filter in the line. The last filter in the pipeline writes to standard output.
- printer A program routine that stays in memory and intercepts and processes printer I/O for printer spooling.
- process Generally a program that is at some stage of execution. In the UNIX System, a process also refers to the execution of a computer environment, including contents of memory,

register values, name of the current directory, status of files, and information recorded at login time.

pseudo-volume

An exact sector-by-sector image of an MS-DOS format fixed or floppy disk, stored as a binary file in a UNIX file system.

scan code mapping

Scan codes are the values that a PC sends when a key is pressed on the keyboard. Simul-Task 386 has the ability to translate or "map" keyboard input into scan codes.

session

A Simul-Task 386 session is a UNIX System process or a group of related UNIX System processes. Simul-Task 386 functionality provides multiple sessions on a system. The standard UNIX System environment allows for a single session comprised of a single process running in the foreground and processes designated to run concurrently in the background to the foreground process.

system hot key

See *hot key*.

virtual memory

A data storage method that, through an elaborate software and hardware memory addressing scheme, allows a PC to act as if it has much more memory than is physically present. To a user, the memory looks like actual physical memory. To the computer, this memory does not physically exist.

virtual PC

A PC simulated by software. In the context of the Simul-Task 386 product, virtual PC actually means virtual AT-compatible PC because Simul-Task 386 runs AT-compatible software in a protected multi-tasking environment.

virtual terminal

A virtual terminal which is available on the console allows sharing of the console display, keyboard, and the mouse by up to eight different processes. Only one of the virtual terminals is considered active, and only the process running in the active virtual terminal produces output to the screen and receives input from the keyboard or mouse. Processes

running on an inactive virtual terminal can produce output but cannot receive input from the keyboard/mouse until the virtual terminal is made active by the user.

virtual terminal manager (VTM)

The name given to the ability within Simul-Task 386 to have more than one task assigned to the screen at the same time.

VP/ix configuration file

The VP/ix configuration file is used for custom tailoring the MS-DOS environment. It is a text file containing many different 1-line entries, each of which consists of configuration information which is parsed and used by Simul-Task 386 when it starts a DOS process. Peripheral device assignments, bootimage specification, and BIOS/ROM specification can be done in the configuration file. The default name for this file is **vpix.cnf**, but it can be changed.

well-behaved

Application programs that use a high-level interface to the hardware. Such interfaces are provided by lower layers of systems software, such as MS-DOS and other operating systems or the BIOS .

window

Throughout this book, "window" means an X Windows window. A window is opened under X by an X Windows system client process which may request input and output through that window. A Simul-Task 386 process running from the console may run in an X Window as well as in a virtual terminal.

zooming

Zooming applies only to DOS processes started under X Windows. It refers to the changing of modes in which DOS processes run. A DOS process running inside an X window can zoom out to run under a virtual terminal by opening a virtual terminal, copying the video buffer, and switching the input and output to the virtual terminal. The zoomed-out DOS process can be zoomed back into the X window. This feature allows graphics intensive and EGA-mode DOS applications to run under X Windows.

Appendix D: Corresponding Commands: UNIX System and MS- DOS

Corresponding Commands: UNIX System and MS-DOS

D-1

Corresponding Commands: UNIX System and MS-DOS

This appendix contains a table of the MS-DOS commands (arranged in alphabetical order) that readily correspond to UNIX shell commands. Syntax of both is shown.

MS-DOS	UNIX System
<code>cd [drive] [path]</code>	<code>cd [dirname]</code>
<code>chdir</code> (See <code>cd</code> .)	
<code>chkdsk [drive] [file]/v</code>	<code>/etc/fsck [options]</code>
<code>comp file1 file2</code>	<code>cmp [options] file1 file2</code>
<code>copy file1 file2</code>	<code>cp file1 file2</code>
<code>date [mm-dd-yy]</code>	<code>date</code>
<code>del file</code>	<code>rm file1 [[file2...filen]</code>
<code>dir [file]/W/P</code>	<code>ls [options] [[file1...filen]</code>
<code>diskcopy [source] [target]</code>	<code>dd [option=value]</code>
<code>erase</code> (See <code>del</code> .)	
<code>find /V/C/N [string] [file]</code>	<code>grep [options]expression file1...filen</code>
<code>format drive /S/1/8/V/B</code>	<code>format</code> <code>/etc/mkfs [options]</code>
<code>md [drive] [path]</code>	<code>mkdir dirname</code>
<code>mkdir</code> (See <code>md</code> .)	
<code>more</code>	<code>morefile</code>
<code>prompt [text]</code>	<code>PS1=" text "</code>
<code>rename oldname newname</code>	<code>mv file1 file2</code>
<code>rd [drive] [path]</code>	<code>rmdir dirname</code>
<code>rmdir</code> (See <code>rd</code> .)	
<code>sort /R + [column]</code>	<code>sort [options] [[file1...filen]</code>
<code>tree [drive]/F</code>	<code>ls -R</code>
<code>type file</code>	<code>cat file</code>
<code>ver</code>	<code>uname -a</code>
<code>vol</code>	<code>labelit [options]</code>

Index

A

Assigning Remote Disks:
 DOSMOUNT
 Syntax,7-5

C

Configuring Simul-Task 386,
 Direct Device Access,3-14
 Enabling Peripheral Devices,-
 3-11
 The DOSPATH Variable,3-7
 The Master vpix.cnf File,3-1
 The VPIXCNF Variable,3-9
 The VPIXTERMDIR Variable,-
 3-10

Converting Text Files: ADDCR,
 RMCR,
 Example,7-2
 Syntax,7-2

D

DDA Installation and Removal,
 DDA Support Tools,3-17
 VDI UNIX System Device
 Driver,3-17
Direct Device Access,
 DDA Installation and
 Removal,3-17
Displaying UNIX System Directories:
 XDIR,
 Example,7-3
 Syntax,7-3

E

Enabling Peripheral Devices,
 Diskettes,3-12
 Mouse,3-11
 Printers,3-11
 Pseudo-Volume Fixed Disks,-
 3-12
 Serial Ports,3-13

F

File Name Mapping,
 Examples,6-13
 Mapping Algorithm,6-12

G

Giving Users Access to Simul-Task
 386,
 If MS-DOS Has Been
 Installed,2-10
 Removing Access to Simul-
 Task 386,2-11

I

Installing and Upgrading the MS-
 DOS Operating System,
 Install/Update Procedure,2-12
Installing Simul-Task 386,
 Before You Begin,2-2
 Giving Users Access to Simul-
 Task 386,2-8
 Installing and Upgrading the
 MS-DOS Operating System,-
 2-11

Installing Simul-Task 386

(Continued)

Installing the IEM Tools

Package,2-21

Overview of Steps in the

Installation Procedure,2-3

Removing Simul-Task 386,2-18

Step-by-Step Installation

Procedure,2-5

Upgrading Simul-Task 386

from Version 1.0,2-4

Introduction to Simul-Task 386,

What Is Simul-Task 386?,1-1

O

Optional Features,

Option Descriptions,9-2

Overriding Terminal Definitions,

Example,9-3

Troubleshooting ,9-4

Overview of Steps in the Installation
Procedure,

With FACE,2-3

Without FACE,2-3

R

Running More Than One Task,

Background Console Tasks,5-7

Creating Virtual Terminals,5-7

VTM On a Serial Terminal,5-8

VTM On the Console Display,-
5-6

Running MS-DOS and UNIX System
Applications,

Piping and Redirection,5-5

Running More Than One

Task,5-6

Running MS-DOS and UNIX System
Applications *(Continued)*

Running MS-DOS Programs

from the UNIX System,5-1

Running MS-DOS Programs

with Simul-Task 386,5-1

Running UNIX System

Commands From MS-DOS,-
5-8

Running MS-DOS Programs from the
UNIX System,

Avoiding Problems,5-4

Example,5-2

Operational Explanation,5-2

Prerequisites,5-2

Running UNIX System Commands

From MS-DOS,

Example,5-9

S

Simul-Task 386 Fundamentals,

Basic Concepts,4-4

Before You Begin,4-1

Changing Your Password,4-2

Logging In,4-2

Logging Out,4-2

Privileged Users,4-3

System Prompts,4-3

The Importance of Your
.profile,4-5

What Applications Can You
Run?,4-4

System File Access From Within
MS-DOS,

An Example of H: and U:

Usage,6-10

Differences Between Drives H:
and U:,6-10

System File Access From Within
 MS-DOS (*Continued*)
 The \$HOME Virtual Disk (H:),-
 6-9
 The UNIX System Virtual Disk
 (U:),6-9
 Troubleshooting,6-11

T

Terminal Capabilities,
 pccompat,A-6
 Screen Attribute Definitions,-
 A-4
 Special Keys,A-5
 svpix and fvpix.,A-5
 Terminal Descriptions,
 ASCII Input to Scan Code
 Mappings,A-6
 Example Syntax,A-1
 Necessary Terminal
 Information,A-1
 Output Character Mappings,-
 A-8
 PC Key/Terminal Key
 Sequence Mappings,A-9
 Terminal Capabilities,A-2
 The DOSPATH Variable,
 DOSPATH Example,3-8
 The Master vpix.cnf File,
 Contents of the vpix.cnf File,-
 3-2
 Example vpix.cnf File,3-2
 The Simul-Task 386 Command Line,
 Example,6-4
 The Simul-Task 386 Menu,
 Menu Options,6-6
 The VPIXCNF Variable,

The VPIXCNF Variable (*Continued*)
 Example VPIXCNF,3-9
 Search Precedence,3-9
 The VPIXTERMDIR Variable,
 Override Order,3-10

U

Using with the XWIN System,
 Using Simul-Task 386 with the
 XWIN System,8-1
 Using Serial Terminals,
 Optional Features,9-1
 Overriding Terminal
 Definitions,9-2
 Required Features ,9-1
 Using Simul-Task 386,
 Basic Considerations,6-1
 Command Line Options:
 doscmd,6-7
 File Name Mapping,6-12
 Installing MS-DOS
 Applications,6-2
 System File Access From
 Within MS-DOS,6-8
 The Simul-Task 386 Command
 Line,6-3
 The Simul-Task 386 Menu,6-4
 Using Simul-Task 386 with the
 XWIN System,
 Configuring Simul-Task 386
 Window Attributes,8-4
 Exceptions and Limitations,8-3
 Starting Simul-Task 386 in an
 XWIN window,8-1
 Terminating Simul-Task 386
 Under XWIN,8-3
 Zooming In and Out,8-2

Utilities,

- Assigning Remote Disks:
 - DOSMOUNT,7-5
- Converting Text Files:
 - ADDCR, RMCR,7-1
- Displaying UNIX System
 - Directories: XDIR,7-3
- DOSPATH and UNIXPATH
 - Delimiters,7-8
- Enabling Remote Disks:
 - REDIR,7-4
- Getting at MS-DOS Partitions
 - from Simul-Task 386,7-7
- QUIT,7-9
- The EMS Driver,7-8
- The Redirector,7-4

W

- What Is Simul-Task 386?,
 - About This Book,1-2
 - Features of Simul-Task 386,1-1
 - Getting Started,1-2
 - Notational Conventions,1-4