««

Introduction

This booklet is one of several manuals that we have provided to help you install and use *X IATE*. We assume that you are familiar with fundamentals of the X Windows system and its operating environment. You should also have some understanding of airline communication principles, particularly interchange/terminal addressing and display/printer options.

As a first step, we will present an overview of the entire *X IATE* system, including both the Gateway and the Terminal. Instructions for installing the Gateway are provided in a separate document.

Overview

The *X IATE* system comprises several components, including communications software and hardware that provides connectivity to the Host, and terminal emulation software that provides airline terminal functionality.

The Terminal works in conjunction with the X IATE gateway product installed on your Server. Your Server is connected to a modem via a standard RS-232 cable.

The Terminal supports screen prints to several types of airline industry printers such as the TI810 and Data Products 120.

A typical *X IATE* system is shown below:

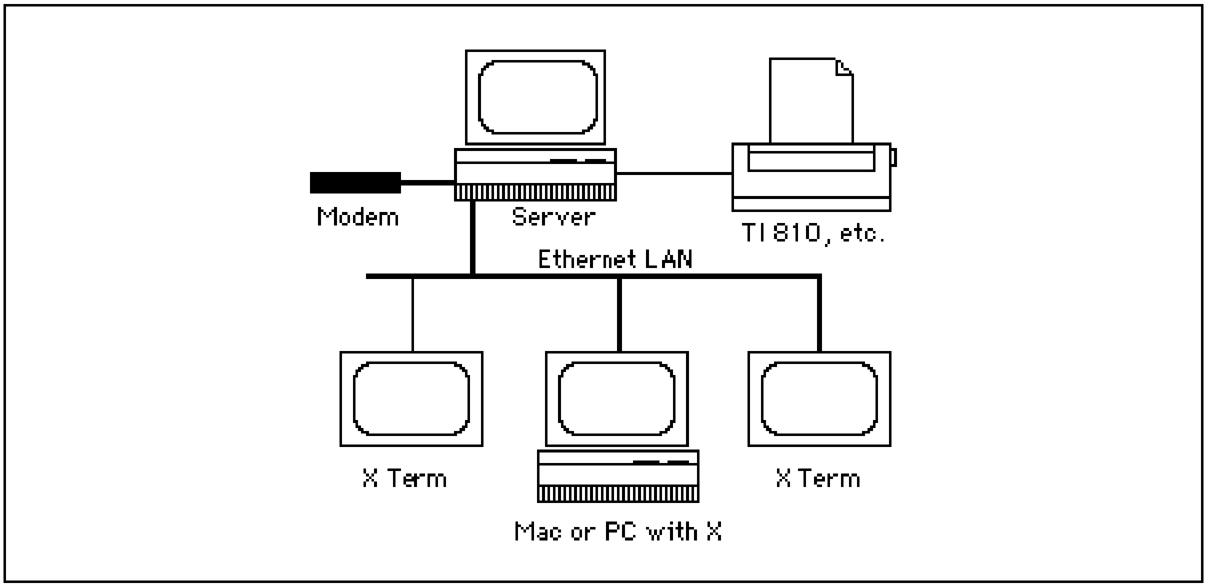

Diagram of typical LAN-based *X IATE* system

««

Preparation

Before installing the Terminal you must set-up your X Window system and install the IATE Gateway. You must also identify the Object Name that is defined for this terminal within the Gateway configuration. For a Macintosh gateway, you must also set up MacTCP and the IATEtcp Router. The procedure for setting up a Macintosh Gateway and Router is given in the *Mac Multi IATE Reference Manual.* UNIX gateway information for Sun systems is given in the *IATE ALC and X.25 Gateway Installation Manual for Sun Systems.*

The LAN

The Terminal communicates with the Gateway over a Local Area Network (LAN), or possibly a long-range network connection, using the TCP/IP protocol stack. Ensure that the network is up and running before you try to install or start *X IATE*. Use the UNIX utility **ping** to check your connection to the gateway machine. For example, suppose that the IP address of your Gateway machine is 192.0.1.18. Enter

```
   ping 192.0.1.18
```

and press the **Enter** key. The utility should report receipt of responses from the Gateway machine. If no responses appear, check the IP addresses and any other configuration parameters on both sides of the connection. (If the **ping** command is not in your directory search path, you may need to specify the path to the command, such as **/usr/sbin/ping**.)

IATE Object Names

Installation of the the Terminal will require an IATE CRT Object Name, and optionally a Printer Object Name. Before beginning the installation process, be sure to determine the CRT Object Name(s) that were created for this terminal as part of the gateway configuration process. If you wish to have this terminal take any available TA from a “group” of TAs which has been defined at the Gateway, then you should use the name of the group as the Object Name.

««

Installing the Terminal

Installation of *X IATE* involves the following four steps, described in detail below.

- Transferring *X IATE* files from the distribution diskette
- Installing the Terminal fonts
- Distributing the Terminal software files to users
- Configuring the Terminal

Transferring files from the distribution diskette

X IATE software and font files must be extracted from the **tar** archive(s) on the distribution diskette(s) and installed onto a UNIX system. This step can be performed as part of Gateway installation, extracting the files onto the same machine that will run the Gateway. For Sun systems, follow the extraction procedures given in the *IATE ALC and X.25 Gateway Installation Manual for Sun Systems.*

After extracting the Gateway and Terminal file sets, the *X IATE* files can be found in the **xiate_term** subdirectory under the IATE product directory where you extracted the files. You should then transfer the Terminal files to any separate machine(s) where the Terminal will be used. To transfer the files to those machine(s), it may be appropriate to use a direct file transfer utility such as **ftp** or **rcp**, or else to use **tar** with floppy diskettes or tapes.

Installing X IATE Fonts

There are several ways to run X IATE in a client-server environment. The workstation you are using for terminal emulation might be connected to an X server on a SPARCstation gateway machine via TCP/IP. Or, all the software needed to run X may be downloaded to a workstation. Whatever the case, the IATE fonts must be installed on the font server machine which provides the fonts to the workstation. The workstation’s font server machine may be the workstation itself, or the machine running the X server, or some other machine. In addition, if you plan to use OpenWindows on a SPARCstation, OpenWindows will need the fonts installed for its use as well.

If you choose to use a non-SPARC workstation as a terminal emulator, please read the manufacturer's documentation about installing fonts, and install the `.bdf` fonts distributed with the IATE software on the workstation's font server.

Installing IATE Fonts for Sun O.S. 4.x:

These commands assume that the `OPENWINHOME` environment variable is correctly set to the home directory for OpenWindows version 3. You must have super-user privileges to execute these commands.

In this example, the **iate_product** distribution directory resides under **/home**. Substitute the name of the directory where you installed the IATE distribution files.

```
   cd $OPENWINHOME/lib/fonts
   cp /home/iate_product/fonts/*.bdf .
   cp /home/iate_product/fonts/*.fb .
   $OPENWINHOME/bin/bldfamily
```

If you are already running OpenWindows, exit and restart it after installing the fonts.

Installing IATE Fonts for Solaris 2.2 (*not* 2.3 or later):

These commands assume that OpenWindows version 3 is installed under **/usr/openwin**
You must have super-user privileges to execute these commands.

In this example, the **iate_product** distribution directory resides under **/home**. Substitute the name of the directory where you installed the IATE distribution files.

```
   cd /usr/openwin/lib/fonts
   cp /home/iate_product/fonts/*.bdf .
   cp /home/iate_product/fonts/*.fb .
   $OPENWINHOME/bin/bldfamily
```

If you are already running OpenWindows, exit and restart it after installing the fonts.

Installing IATE Fonts for Solaris 2.3 or later:

These commands assume that OpenWindows version 3 is installed under **/usr/openwin**
You must have super-user privileges to execute these commands.

In this example, the **iate_product** distribution directory resides under **/home**. Substitute the name of the directory where you installed the IATE distribution files.

```
   cd /usr/openwin/lib/X11/fonts/misc
   cp /home/iate_product/fonts/*.pcf.Z .
   /usr/openwin/bin/mkfontdir
```

If you are already running OpenWindows, exit and restart it after installing the fonts.

Converting older IATE font files to “pcf” format for Solaris 2.3 or later:

If you are working from an older distribution of XIATE, you may not have the “pcf” formatted font files needed for Solaris 2.3 or later. In that case, you can use the OpenWindows **bdftopcf** utility to create the “pcf” formatted files, as follows:

These commands assume you are running the “csh” C-Shell, not the Bourne shell. Note that two back-slanted quote marks are required on the next-to-last line — *not* forward-slanted apostrophes. The shell will display the question marks; do not type them in.

```
cd /usr/openwin/lib/X11/fonts/misc
cp /home/iate_product/fonts/*.bdf .
foreach f in (iate*.bdf alc*.bdf)
? /usr/openwin/bin/bdftopcf $f -o `echo $f | sed s/bdf/pcf/`
? end
```

Distributing X IATE Files to Terminal Users

XIATE runs in an OpenWindows environment. The IATE and ALC fonts should be installed as described above.

XIATE terminal users will be able to access the terminal by typing "`xiate`" from their home directories. To make this possible, create symbolic links in the home directory of each user, and copy the XIATE file into each directory, as follows:

For SABRE installations:

```
   ln -s /home/iate_product/xiate_term/iatekey.sabre .iatekey
   ln -s /home/iate_product/xiate_term/.iatehelp.sabre.sun .iatehelp
   ln -s /home/iate_product/xiate_term/xiate xiate
   cp /home/iate_product/xiate_term/XIATE XIATE
```

For all other installations:

```
   ln -s /home/iate_product/xiate_term/iatekey.apollo .iatekey
   ln -s /home/iate_product/xiate_term/.iatehelp.apollo.sun .iatehelp
   ln -s /home/iate_product/xiate_term/xiate xiate
   cp /home/iate_product/xiate_term/XIATE XIATE
```

Configuring the Terminal

The first time a given user starts the Terminal, it displays a dialog box requesting the name of the Object, Server, and Gateway to which the terminal should attach. If this dialog box does not appear, the **DISPLAY** environment variable may not be set correctly. For example, if the user's X display client machine has IP address 192.0.1.18, then **DISPLAY** should be set to: `{192.0.1.18}:0`

The user must enter the appropriate Object, Server, and Gateway names into the startup dialog box. For example, suppose that the user intends to use a CRT object named **term1** on an IATE Gateway running on machine **gatesys**, through TCP service name **ialcserver**. Then the user must enter **term1**, **ialcserver**, and **gatesys** into the three fields of the dialog box. (To use a group of TAs, use the group name in place of the object name. Refer to the Gateway documentation for more information about object definitions.)

Once the Gateway access configuration is set in this way, the terminal will remember the three names entered. On each subsequent startup, the terminal will attempt to connect to the same object (or group) as before.

If it becomes necessary to change the object name, gateway name or service name, delete the **.xiate.cfg** file in the user's home directory. On its next subsequent startup, the terminal will request new names.

Additional configuration parameters can be set in the user's **XIATE** file (all caps). Lines in this file that begin with "XIATE" (all caps) define X resource settings for *X IATE*. These lines generally take the following form (with exceptions noted below):

```
   XIATE*name: value
```

`name` is the name of the resource, as given in the list below; and `value` is the value to be configured for the resource. Changes made to a user's private **XIATE** file will affect that user only. If you want one or more users to share the same customized settings, you can create a shared **XIATE** file referenced by symbolic links in each user's home directory.

The class name for *X IATE* is "XIATE". Resources' instance names start with a lower case letter (`drawfont`, for example).

The class name "XIATE" appears in the **XIATE** file at the beginning of each line that sets the value of a resource. Settings that use this name can affect all terminals run by the user who owns the **XIATE** file. Alternatively, you can specify a specific program file's name, to override settings for that program file only. For example, suppose you create a symbolic link to the **xiate** program in the home directory of user **tom**, and call that link **tomterm**. Then **tomterm** acts as a separate program file, to which you can assign specific resource settings. Any resource settings in the XIATE file that use the name "tomterm" will override any corresponding "XIATE" settings whenever the user runs **tomterm**.

X IATE uses several of the general-purpose X Windows and Motif or OpenLook resources (Font, background, foreground, display, geometry, etc.) Please see the X Server manual and the Motif or OpenLook documentation for more information on these resources. *X IATE* also responds to a number of unique resources. All but `printerName` are optional and have defaults that will be used if the resource is not set explicitly. `printerName` must be specified before the user can print the contents of the terminal screen (an operation known as a "screen print" or "local print"). `printerName` has no default.

Resource Name	Default	Description
`drawfont`	`iate8`	X Font for terminal display. This can be any Character type font available on the X Server. (We recommend a fixed-width font.) *X IATE* includes `iatesize` for SABRE, and `alcsize` for other host types. `size` is the point size, generally 6 or 8.
`blinkfont`	`iate8B`	IATE X font used to display the blinking attribute sent from the host. The size of this font should be the same as the `drawfont`. *X IATE* includes boldface fonts `iatesizeB` and `alcsizeB` which can be used for this purpose.

`rows`	15	The number of rows in each terminal window. This value can range from 15 to 33 inclusive. It should be set to the number of display rows (defined at the airline host) for the terminal's TA, *plus one* or more.
`cols`	64	The number of columns in each terminal window. This value can range from 64 to 80 inclusive. It should be set to the number of columns (defined at the airline host) for the terminal's TA, or more.
`printerName`		Name of printer object: this must be specified in order to execute screen prints.
`blockCursor`	`False`	Set this to `True` if you want to the cursor to display as a block covering a character space, instead of a line or caret between characters in terminal windows.
`showReturns`	`False`	Set this to `True` when operating in FOS mode. The `True` setting causes Carriage Returns (at the end of each line) to be displayed as a visible symbol. The SOM character also appears.
`switchClearBackspace`	`False`	Set this to `True` if you wish to swap the functions of the **Backspace** and **Clear** keys. In that case, the key that normally clears the terminal screen will perform a destructive backspace, while the key that normally performs a destructive backspace will clear the terminal screen.
`serverPort`	0	The system administrator may change this when there are multiple X servers on a network.
`startAction`		You may use this resource to specify the name of a file from which the terminal should read and execute programmable function key commands at startup.
`endAction`		You may use this resource to specify the name of a file from which the terminal should read and execute programmable function key commands at shutdown.

Note: In a "startAction" or "endAction" file, or any key or file that gets called by the "startAction" or "endAction" file, you must not use or call function key #30 or you may get undefined results. Beyond that restriction, you may use all of the programmable function key commands, including calling other function keys or function key files.

Using the Terminal

While the airline host system itself hasn't changed, the power of the X Window System has been leveraged to make working with the host easier and more efficient.

You can still use all the the same host commands that you are familiar with. But now you have a variety of additional Terminal commands that will save you time and effort in many of your routine tasks. It isn't necessary to remember this new set of commands. Most of them are readily available on one of Terminal's pull-down menus. The pull-down menus, and the commands they contain, are described later in this section.

The Status Window

The first line inside the Status Window displays information about the status of the session with airline host. The status messages that may be displayed here include:

Sys Avail The host system is available for use.

Sys Unavail The system is unavailable. Either the host, the gateway, or the network is down.

KyBd Locked A message or command has been sent to the Host, but a response has not yet been received. If no response is returned by the host, the terminal's Reset key (usually mapped to **Ctrl-R** or **Esc** on the keyboard) may be pressed to unlock the keyboard.

Push Rpt This message appears when (**1**) the terminal has received a message from the host with an error, (**2**) a message from the host is longer than the window size the operator has selected, or (**3**) the message from the host has invalid screen positioning commands. If this message appears, press the Reset key and then re-enter the last command.

Umsg The terminal has received an unsolicited message indicator from the host. Press UMSG to display the text of the unsolicited message.

Reenter Data cannot be sent to the host. If this message appears, press Reset and re-enter the last command. This is usually caused if there is no SOM on the screen when the Enter key is pressed.

Insert Indicates that the terminal is in insert mode. Any characters typed will be inserted between the existing characters as long as this message appears in the Status Window. Each line can accommodate a maximum number of characters determined by the `cols` resource setting (default 64). Any characters pushed beyond that limit by the insert function will be lost.

Logging Indicates that Capture Session is on.

Printing Indicates that a screen print is in progress.

Protect Indicates that the terminal is in protected screen mode.

The Broadcast Service Line occupies the lower portion of the Status window. It is used for two purposes: (**1**) to support special message functions; and (2**)** to display prompts used during execution of programmable function keys. See Appendix A for additional details.

The File Menu

The **File** pull-down menu contains the following commands:

New Window Opens a new Terminal window for the currently active Host session.

Close Window Closes the currently active Terminal window. This is the same as clicking on the window menu's Close item.

Print Allows the user to print the contents of the active Terminal window to an active printer on the network (specified by the `printerName` resource discussed earlier).

Quit Exits the Terminal program. (This does not sign you off of the Host).

The Edit Menu

The **Edit** pull-down menu contains the following commands:

Clear Removes the highlighted text from the active Terminal window without placing the text on the clipboard.

Add Returns to Copies
Ensures that there is a carriage return at the end of each line being copied out of the window. While the Add Returns to Copies function is on, a mark appears to the left of this command.

Note: The **Select** and **Paste** functions are provided by buttons on the mouse. Please see your Window Manager user documentation. Within the Terminal, the **Select** function is assigned to the left button on the mouse and the **Paste** function is assigned to the center button.

The Sessions Menu

The **Sessions** pull-down menu contains the following commands:

Capture Session Toggle
Switches the Session Capture function on and off. The Session Capture function saves all text sent to and from the Host into a file. While the Session Capture function is on, the label "**Logging**" appears in the Status window.

Look At Session Displays a window that allows you to scroll through the text saved to the Session file.

Print Session Prints the contents of the Session file buffer.

Delete Session File Deletes the current capture session file.

The Function Keys Menu

The **Function Keys** pull-down menu contains the following commands:

Enter Key Definition Allows you to enter or edit the definition of Programmable Function Keys (PF Keys). See Appendix A: PF Keys for more information.

Save Function Keys Allows you to save the current PF Key definitions under a specified file name for later retrieval. A window appears in which you can enter the file name. Then press the OK button to save.

Load Function Keys Allows you to load a file containing PF Key definitions. A window appears in which you can enter the file name. Then press the OK button to load.

View Function Key Labels
Displays a window with list of PF Key numbers and labels so that you may easily recall the PF Key definitions. You can double-click on any one of the displayed definitions to invoke and execute its programmed command string.

There are several ways to define the PF Keys. See Appendix A, below, for more information.

The Configure Menu

The **Configure** pull-down menu contains the following commands:

Display Returns (FOS) This command acts as a toggle to select whether or not Carriage Returns (at the end of each line) will be displayed as visible symbols. Some hosts require the display of the return character; others do not.

Configure Server Connection

Displays the Object, Service, and Gateway machine names for this Host session. These are labeled “Object Name”, “Server Name”, and “Host Name” in the displayed output. Press either the OK or Cancel button to dismiss the window that displays the names.

Appendix A — PF Keys

Programmable Function Keys (PF Keys) allow you to script commands and command sequences that would otherwise have to be entered manually.

After a key is defined, it may be executed by either pressing the appropriate key on the keyboard or selecting the key from a pull-down menu. PF Keys can be nested up to 12 times.

Defining PF Keys

One PF key definition method makes use of the **Alt** key. On any line in the Terminal window, hold down the **Alt** key while pressing the appropriate PF Key (the keys labeled **F1** through **F30**) to save that line in the selected PF Key. A star will appear at the end of the line to indicate that the text between the SOM and the cursor was loaded into the selected PF Key. A message will also appear in the Status Window indicating which PF Key was loaded.

PF Keys may also be created using the **Enter Key Definition** command on the Function Keys pull-down menu. When this command is selected, a dialog window will appear. To define a PF Key, fill in the three fields on the PF Key definition dialog:

Number Enter the number of the PF Key that you are defining. The valid range of numbers is from 1 to 30.

Label Enter the label by which you will refer to this PF Key. The label may be up to 9 characters long. The label will appear beside this key number on the **Key Labels** pull-down menu.

Text Enter the definition of this PF Key. The text may be up to 255 characters long. See the next section, PF Key Commands, for additional instructions.

To define another PF Key, first click on the **Save** button; then enter a new number in the Number field; press the **Tab** key, or the **Return** or **Enter** key on the main part of the keyboard (not the Enter key on the numeric keypad), and complete the new definition. When you have finished defining PF keys, click on the **Save & Quit** button, to save your updated definitions and exit the PF Key definition dialog. Or, click on the **Quit** button to exit without saving the current definition.

PF Key Commands

All function key commands include the Field Mark (displayed as the `^` character on the screen) followed by a letter or number code which represents a command. When defining your function keys, use **Ctrl+F** to designate a Field Mark.

The definitions of several commands use the term "*insertion point*". This refers to the current location of the cursor within the terminal screen.

You can use any of the following command codes to define a PF key, by entering the commands into the text area of the **Define Function Keys** dialog box (illustrated earlier under the description of the **PF Keys** menu).

nn	Fixed field length
A	Send entire screen to host
C	Position insertion point (cursor)
D	Insert optional text
E	Auto-enter
F	Execute PF Key
H	Wait for reply from host
I	Search for tab character
K	Erase to end of page
O	Input optional text
R	Send the next host reply to specified window
S	Move SOM to specified location
T	Display prompting text
V	Fill in variable length field
W	Wait for input
X	Execute commands in a specified file

Command Code Examples

***Note:* The following examples use formats from a variety of CRS hosts. Although the examples are intended to be general, the correct formats for your particular host may differ from the formats presented here. Consult appropriate format documentation for your host system if necessary.**

nn This command copies a fixed number of characters from the screen beginning from the SOM position, into the character string generated by pressing a PF Key. For example, if the contents of PF1 are:

```
†SR1†03LAX
```

and the user enters "`DFW`" at the SOM and then presses the PF1 key, the screen displays:

```
1DFWLAX
```

A This command is used to include all data in the terminal window in any message to the host. For example, if a form was displayed in the terminal window for the user to fill in and PF1 was defined as `†A†E`, pressing the PF1 key would send the form to the host.

C This command moves the insertion point to the specified location in the terminal window. Forms include:

C*	Place the insertion point at the SOM.
C*nn*	Place the insertion point at column 1 of line number *nn*.
C-*cc*	Place the insertion point at column *cc* of the current line.
C*nn*-*cc*	Place the insertion point at the specified line *nn* and column *cc*.

For example:

```
†C20-45DFWLAX
```

would position the insertion point at line 20, column 45 and write

```
DFWLAX
```

starting from that point.

Important: The insertion point need not necessarily be positioned at the SOM when a PF key is pressed. Use of the SOM positioning and cursor positioning commands eliminates that requirement. Just as with any other PF Key command, data to fill in fields is taken from the area between the old SOM and the insertion point that was in effect when the PF Key was pressed.

D This command allows the user to replace data in a PF key command with data the user enters. The data in the PF Key command that may be replaced is delimited with a Field Mark. The end of the replacement data (not limited to the same number of characters as are being replaced) must either be delimited by a Field Mark or be at the end of the PF Key data (with the insertion point at the end of the data when the PF Key was pressed). If the user enters no data, no replacement occurs.

For example: If the contents of PF1 are:

```
    †SR1DFWLAX†D6P
```

and the user enters

```
    4P
```

and then presses the PF1 key when at the SOM, the following text is displayed in the terminal window:

```
    1DFWLAX4P
```

If user presses the PF1 key without entering any data, the following text is displayed:

```
    1DFWLAX6P
```

E This function sends the text between the SOM and the current insertion point to the host, as if the user had pressed the Enter key. For example, if the contents of PF1 are:

```
    ALAXDFW†E
```

and the user presses the PF1 key with the insertion point at the current SOM, the availability display command

```
    ALAXDFW
```

is displayed on the screen and sent to the host.

F This function allows the user to execute a PF Key from inside another PF Key function. The PF key is defined with two digits (e.g. as F03, not F3). For example,if the contents of PF1 are:

```
    ADFW†F03†E
```

and the contents of PF3 are:

```
    LIT
```

and the user presses PF1 at the current SOM, then the command

```
    ADFWLIT
```

is displayed on the screen and sent to the host.

H This function stops execution of the current PF Key until a response is received from the host. This function normally follows the **E** PF Key function. For example, if the contents of PF1 are:

```
    1DFWIAH†E†H1DALHOU†E
```

and the user presses the PF1 key at the current SOM, the command

```
    1DFWIAH
```

is displayed on the screen and sent to the host. When the response is received, the second command

```
    1DALHOU
```

is displayed on the screen and sent to the host. While waiting for the host reply, the message "**Holding for host reply. Press Reset (Esc) to end.**" appears in the Status Window. Pressing the Reset key (usually mapped to **Esc**) cancels the execution of the PF key.

I This command searches for the next host tab character in the terminal window, or the next user-defined tab stop in the terminal window, and moves the insertion point to that location. If no tab position can be found, the insertion point moves to the bottom right corner of the terminal window.

K This function erases all characters between the current insertion point and the bottom right corner of the terminal window.

O This command optionally replaces a field in a PF Key command with a replacement field entered by the user. The **O** PF Key command must be followed by a field number, for example, **O0**, **O1**, **O2**, **O3**, etc. Field numbers must be in the range 0 - 9. Each optional field in the PF Key definition must be terminated by a Field Mark (†). If the user enters an optional field number and no data, then that field is dropped from the PF text.

Replacement field data the user enters on the terminal screen must be preceded by a "check mark" (√) and the number of the field to be replaced. The data must be terminated by a Field Mark.

For example, if the contents of PF1 are:

```
A†O1TUL††O2IAH††O36P††E
```

and the user presses the PF1 key, the command

```
ATULIAH6P
```

is sent to the host.

As another example, if the user enters

```
√2LAX†
```

and presses the PF1 key, the command

```
ATULLAX6P
```

is sent to the host.

As a final example, if the user enters

```
√1SMF†√39A†
```

and presses the PF1 key, the command

```
ASMFIAH9A
```

is sent to the host.

R*nn* This command sends the next host reply to window *nn*. If a host reply command `†H` is not subsequently given, the `R` command is ignored. If window *nn* does not exist, the user is asked whether to cancel the function key or send the output to the current window. For example, if the contents of PF1 are

```
1LAXORD†E†R02†H
```

then the terminal would send the following text to the host:

```
1LAXORD
```

and the reply would appear in terminal window #2.

S This command moves the SOM to the specified location in the terminal window. Forms include:

`†S*` Position SOM at the current cursor location. (For some hosts, this works only when SOM characters and carriage returns are being displayed — otherwise an error occurs and key execution stops.)

`†Snn` Position SOM at column 1 of line number *nn*.

`†Snn-cc` Position SOM at the specified line *nn* and column *cc*. (For some hosts, this works only when SOM characters and carriage returns are being displayed — otherwise an error occurs and key execution stops.)

`†SR` Position SOM at the start of the line below the current line. (If already at the bottom line, the SOM "wraps around" to the top line.)

For example, if the contents of PF1 are:

```
†SR†C*1DFWLAX†E
```

then pressing PF1 with the insertion point anywhere in a terminal window causes the SOM to be placed at the beginning of the next line. Then the insertion point is placed at the current SOM (see the definition of the **C** PF Key command), and the command

```
1DFWLAX
```

is displayed on the screen and sent to the host.

Important: The insertion point need not necessarily be positioned at the SOM when a PF key is pressed. Use of the SOM positioning and cursor positioning commands eliminates that requirement. Just as with any other PF Key command, data to fill in fields is taken from the area between the old SOM and the insertion point that was in effect when the PF Key was pressed.

T This command displays on the second line of the status box the text following the **T** command. The text must be terminated by either the Field Mark or the end of the key definition. The "Wait for Input" PF Key command (**W**) often follows the **T** PF Key command.

For example, if the contents of PF1 are:

```
A†TENTER DESTINATION†SFO†W†E
```

If the user presses the PF1 key,

```
ASFO
```

is displayed on the screen, and

```
ENTER DESTINATION
```

is displayed on the second line of the status bar. If the user then enters:

```
DFW†
```

then

```
ASFODFW
```

is sent to the host.

V This command optionally fills in fields in a PF Key command with fields entered by the user. Multiple fields may be replaced. If the user enters multiple fields, they should be separated by Field Marks. For example, if the contents of PF1 are:

```
    1†V†V†E
```

If the user enters:

```
    DFW†LAX
```

and then presses the PF1 key, the command

```
    1DFWLAX
```

is sent to the host.

W This command stops further processing of a PF Key command until the user enters data. The user must terminate the input with a Field Mark. The message “**Press Reset (Esc) or end input with Field Mark**” appears in the Status Window.

For example, if the contents of PF1 are:

```
    1DFW†W†E
```

and the user presses the PF1 key, the command

```
    1DFW
```

is displayed.

The user must enter data terminated by a Field Mark before the PF Key will complete. Pressing the Reset key (usually mapped to **Esc**) cancels the execution of the PF key.

X This command executes commands stored in a specified text file. The file should be saved in the same directory as the terminal. The file name immediately follows the **X** command, and is terminated by a Field Mark or the end of the PF Key definition. Lines in the file are executed just as if they were part of a PF Key definition. The name of the file must be followed by a Field Mark. Inside the file, the Field Mark should be entered as the “**^**” character (Shift-6). All letters that are part of a message to the host should be entered as upper case letters.

The first characters of the first line in the file should be the SOM/Cursor Reset command `^SR^C*` (or `†SR†C*`). The first characters of each subsequent line in the file should be the Cursor Reset command `^C*` (or `†C*`). Each host command (such as “`1MSPORD`”) should be followed by the commands `^E^H` (or `†E†H`). The message, “**Holding for host reply. Press Reset (Esc) to end.**”, appears in the Status Window while the **X** command is executing. Pressing the Reset key (usually mapped to **Esc**) cancels the execution of the PF key.

For example, if the contents of PF1 are:

```
†XBOSTON.PF
```

and the contents of the file **BOSTON.PF** are the following three lines:

```
^SR^C*1MSPBOS^E^H
^C*1BOSATL^E^H
^C*1ATLMSP^E
```

and you press the PF1 key, the command:

```
1MSPBOS
```

is sent to the host and the response is displayed on the screen. Next, the command:

```
1BOSATL
```

is sent to the host and the response is displayed on the screen. Finally, the command:

```
1ATLMSP
```

is sent to the host and the response is displayed on the screen.

Y This function sends the text between the SOM and the current insertion point to an application that has been linked to the terminal. For example, if the contents of PF1 are:

```
ALAXDFW†Y
```

and the user presses the PF1 key with the insertion point at the current SOM, the command

```
ALAXDFW
```

is displayed on the screen and sent to the application.

X IATE™

InnoSys Airline Terminal Emulator for the X Window System / Sun OpenWindows

User's Guide

InnoSys
INCORPORATE

NO WARRANTIES OF ANY KIND ARE EXTENDED BY THIS DOCUMENT. The information herein and the IATE™ products themselves are furnished only pursuant to and subject to the terms and conditions of a duly executed Product License. INNOSYS SPECIFICALLY DISCLAIMS ALL WARRANTIES, WHETHER IMPLIED OR EXPRESSED, INCLUDING BUT NOT LIMITED TO THOSE OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. InnoSys has no responsibility, financial or otherwise, for any result of the use of this document and/or the associated product, including direct, indirect, special and/or consequential damages. The information contained herein is subject to change without notice.

INSCC, INSCC-S, and INSCC-LC cards have been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If a computer with INSCC, INSCC-S, or INSCC-LC card(s) installed does cause harmful interference to radio or television reception, which can be determined by turning the computer off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient the receiving antenna
- Increase the separation between the computer and receiver (radio or TV)
- Connect the computer into an outlet on a circuit different from that to which the receiver is connected.

Note that shielded cables must be used with INSCC, INSCC-S or INSCC-LC cards in order to meet the FCC Class B emission limits. Changes or modifications to the INSCC, INSCC-S or INSCC-LC cards not expressly approved by InnoSys Inc. could void the customer’s right to operate the equipment.

Contents