
MINISOFT FOR IBM

MS3270 AND MS5250

MiniSoft, Inc.
1024 First street
Snohomish, WA 98290
U.S.A.

1-800-682-0200
360-568-6602
Fax: 360-568-2923

MiniSoft Marketing AG
Papiermuhleweg 1
Postfach 107
Ch-6048 Horw
Switzerland

Phone: +41-41-340 23 20
Fax: +41-41-340 38 66
minisoftag@centralnet.ch

Internet access:

sales@minisoft.com
support@minisoft.com
<http://www.minisoft.com>
<ftp://ftp.minisoft.com>

DISCLAIMER

The information contained in this document is subject to change without notice.

MiniSoft, Inc. makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. MiniSoft, Inc. or its agents shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishings, performance, or use of this material.

This document contains proprietary information which is protected by copyright. All rights are reserved. No part of this document may be photocopied, reproduced, or translated to another programming language without the prior written consent of MiniSoft, Inc.

©1997-8 by MiniSoft, Inc., Printed in U.S.A.

1st version of this manual printed November 19, 1997.

2nd version of this manual printed February 9, 1998.

3rd version of this manual printed ????

All product names and services identified in this document are trademarks or registered trademarks of their respective companies and are used throughout this document in editorial fashion only and are not intended to convey an endorsement or other affiliation with MiniSoft, Inc.

LICENSE AGREEMENT

In return for payment of a onetime fee for this software product, the Customer receives from MiniSoft, Inc. a license to use the product subject to the following terms and conditions:

- The product may be used on one computer system at a time: i.e., its use is not limited to a particular machine or user but to one machine at a time.
- The software may be copied for archive purposes, program error verification, or to replace defective media. All copies must bear copyright notices contained in the original copy.
- The software may not be installed on a network server for access by more than one personal computer without written permission from MiniSoft, Inc.

Purchase of this license does not transfer any right, title, or interest in the software product to the Customer except as specifically set forth in the License Agreement, and Customer is on notice that the software product is protected under the copyright laws.

90-Day Limited Warranty

MiniSoft, Inc. warrants that this product will execute its programming instructions when properly installed on a properly configured personal computer for which it is intended. MiniSoft, Inc. does not warrant that the operation of the software will be uninterrupted or error free. In the event that this software product fails to execute its programming instructions, Customer's exclusive remedy shall be to return the product to MiniSoft, Inc. to obtain replacement. Should MiniSoft, Inc. be unable to replace the product within a reasonable amount of time, Customer shall be entitled to a refund of the purchase price upon the return of the product and all copies. MiniSoft, Inc. warrants the medium upon which this product is recorded to be free from defects in materials and workmanship under normal use for a period of 90 days from the date of purchase. During the warranty period MiniSoft, Inc. will replace media which prove to be defective. Customer's exclusive remedy for any media which proves to be defective shall be to return the media to MiniSoft, Inc. for replacement.

ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS IS LIMITED TO THE 90-DAY DURATION OF THIS WRITTEN WARRANTY. Some states or provinces do not allow limitations on how long an implied warranty lasts, so the above limitation or exclusion may not apply to you. This warranty gives you specific rights, and you may also have other rights which vary from state to state or province to province.

LIMITATION OF WARRANTY: MiniSoft, Inc. makes no other warranty expressed or implied with respect to this product. MiniSoft, Inc. specifically disclaims the implied warranty of merchantability and fitness for a particular purpose.

EXCLUSIVE REMEDIES: The remedies herein are Customer's sole and exclusive remedies. In no event shall MiniSoft, Inc. be liable for any direct, indirect, special, incidental, or consequential damages, whether based on contract, tort, or any other legal theory.

TABLE OF CONTENTS

INTRODUCTION	1-1
Overview	1-1
Features	1-1
Session Management and FTP Utilities	1-2
File Transfer	1-2
Printing	1-2
System Requirements	1-3
USING MINISOFT FOR IBM	2-1
Connecting to the Host	2-1
Session Manager	2-3
Creating a Session	2-3
Copying a session	2-4
Deleting Sessions	2-5
KEYBOARDING	3-1
Key Mapping	3-1
Keyboard	3-4
Current Key	3-4
3270 Default Keyboard Assignments	3-7
5250 Default Keyboard Assignments	3-8
Accelerator Keys	3-10
PRINTING	4-1
Host Print	4-1

Connecting the Host Print Session	4-2
Host Printer Command Line	4-3
Saving and Retrieving a Host Printer Session	4-3
Print to File	4-4
Print Screen	4-5
Print Setup	4-6
TECHNICAL REFERENCE	5-1
Record Log File Selection	5-1
Mouse	5-2
SCRIPTING	6-1
Running a Script File	6-1
Start Recording Script File	6-2
Script Functions	6-2
Edit Script File	6-10
HLLAPI.....	7-1
HLLAPI Function Reference	7-3
COLOR	8-1
Color	8-1
PREFERENCE	9-1
Prefences	9-2
Print Font Setup	9-4
Accelerator Key Operation	9-4
FONT.....	10-1

FILE TRANSFER 11-1

- Record Trace File 11-4
- Replay Log File Selection 11-4
- Open Session 11-5
- Status Bar 11-5
- Application Menu 11-6
- Capturing Screens 11-8
- Toolbar Buttons 11-8

INDEX 1

1

INTRODUCTION

OVERVIEW

Minisoft for IBM is an IBM 3270 and 5250 terminal emulator designed specifically for Windows 95/98 or NT 3.51/4.0.

Minisoft for IBM has a full array of features to support reliable / fast host access to your IBM mainframe using the Internet. Its small footprint, lightning-fast load, quick screen update and easy-to use connection interface takes the burden out of host access. Included in Minisoft for IBM is a 32-bit FTP client.

Installation is quick and easy. Minisoft for IBM is compatible with Microsoft Terminal Server.

FEATURES

- ◆ Accurate emulation
- ◆ Easy session management
- ◆ Fast screen updates
- ◆ Host printing (3270E)
- ◆ IND\$FILE transfer
- ◆ Keyboard mapping
- ◆ HLLAPI

- ◆ Script language
- ◆ User configurable toolbar
- ◆ NFT 32-bit FTP Client

SESSION MANAGEMENT AND FTP UTILITIES

With easy-to-use session management, a full set of features, and a comprehensive array of FTP utilities, Minisoft for IBM includes all the tools necessary for productive host access, Internet management, and file transfer.

The innovative Session Manager enables creation, modification, and saving of named sessions, including associated settings. Sessions may be started by clicking on an icon in the program group or from the session manager inside the program. The session manager automatically adds or deletes session icons whenever sessions are created or deleted.

FILE TRANSFER

IND\$File transfer is included and supports Write Structured Field and Cutmode protocol. Both ASCII and binary transfers are available.

The FTP client enables connections to remote systems for browsing through directories and files. File transfers in either direction are supported.

A traditional FTP interface (NFT Screenshot) and a FTP utility with an Explorer interface are included.

PRINTING

Print screen, and host (3270 Extended / 3278) printing are supported. Both named LU and generic LU printing are available.

SYSTEM REQUIREMENTS

- ◆ PROCESSOR: 486, Pentium or compatible PC
- ◆ RAM: 16 MB minimum
- ◆ DISK SPACE: 9 MB
- ◆ VIDEO: Windows 95 or NT 4.0 compatible adapter, 800x600 resolution or higher recommended
- ◆ OPERATING SYSTEM: Windows 95/98 or NT 3.51/4.0
- ◆ I/O: TCP/IP network connection

Chapter 1: Introduction

Chapter 2: Display Emulation

- *Toolbar buttons
- *Session manager
 - Creating a session
 - Open session
 - Copy Session
 - Delete Session
- *Scripts
 - Running a script File
 - Start recording a script file
 - Edit Script Files
 - Script Functions
- *Color dialog box
- *Status Bar
- *Font

Chapter 3: Transfer Files

- *Connecting to the Host
- *Setting File Transfer
- *Transferring a File

Chapter 4: HLLAPI

- *HLLAPI Function Reference

Chapter 5: Emulation Keyboards

- *Key Mapping
 - Keyboard
 - Current Key
- *3270 Default Keyboard Assignments
- *5250 Default Keyboard Assignments
- *Accelerator Keys
 - Accelerator Key Operations

Chapter 6: Printer Emulation

- *Host Print
- *Connecting the Host Print Session
 - Host print command line
 - Print setup
 - Print Font setup
 - Saving and retrieving a host print session
 - Print to file
 - Print screen

- ??? Record Log File Selection
- Mouse
- Preferences
- Record Trace File
- Replay Log File Selection
- Application

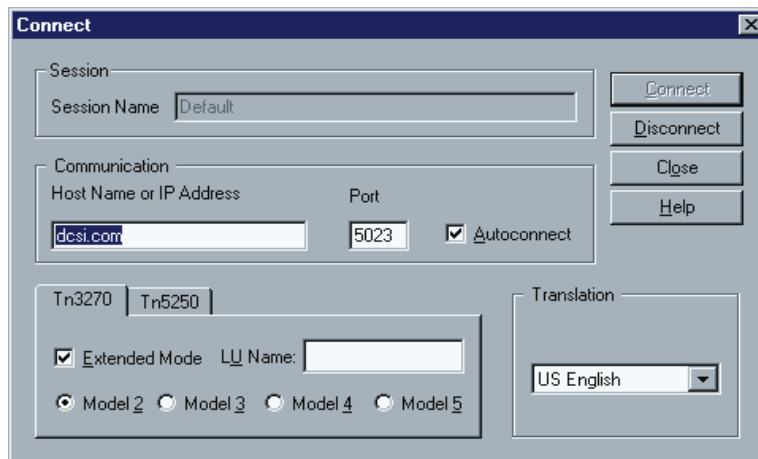
2

USING MINISOFT FOR IBM

CONNECTING TO THE HOST

Minisoft for IBM includes a *Session Manager* for creating, saving, and starting customized sessions (see *Session Manager* in chapter 2). However, to connect quickly to a host without having to configure a session:

1. Click on *Connect - Connect* to display the Connect dialog box:



2. Enter the host name in the Host Name or IP Address field. A valid host name can be an IP address, a name in a local host file, or a name resolvable by a Domain Name Server (DNS).

3. Set the port number in the Port field. The default port number is 23, which is the default number for the Telnet service/protocol. But any number in the range from 1 to 32768 can be used.
4. Select a terminal, either Tn3270 or Tn5250.

If Tn3270 emulation is desired, select the mode and model in the Tn3270 terminal tab. Extended mode informs the host that the terminal emulator supports extended attributes (underline, blinking, reverse and eight colors). The different models are 2 (24×80), 3 (32×80), 4 (43×80) and 5 (27×132).

If using Tn5250 emulation, select a terminal size. The terminal size is based on the number of rows times the number of columns. The following options are available:

24x80

27x132

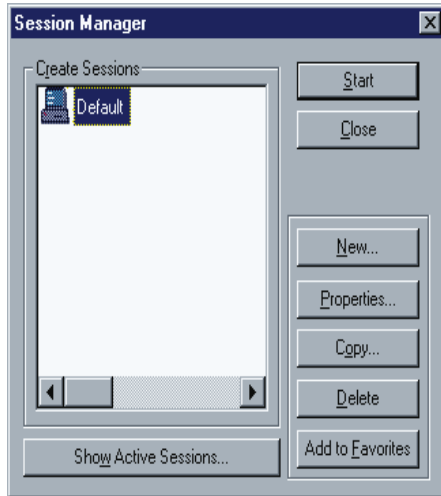
Note: The LU name is used for printing when connection to a specific LU pool is desired. The same LU name must be entered in the Host Print program. The LU name is not necessary if a generic pool of LUs are used. If Specific LU printing is used, the system administrator must supply the correct LU name. If host printing is not required, ignore this menu item.

5. Select the language translation from the Translation list box. The following language translation between EBCDIC and Ansi are available: Austrian, Belgian, Danish, Finnish, French, French Canadian, German, Italian, Netherlands, Norwegian, Portuguese, Spanish, Swedish, Swiss, UK English and US English.
6. If AutoConnect is enabled the host will be connected when the program is started.

SESSION MANAGER

Using the connection manager, sessions can be created, edited, started, and deleted. Whenever a session is created, deleted or renamed, the session icon in the program group is also added, deleted or renamed.

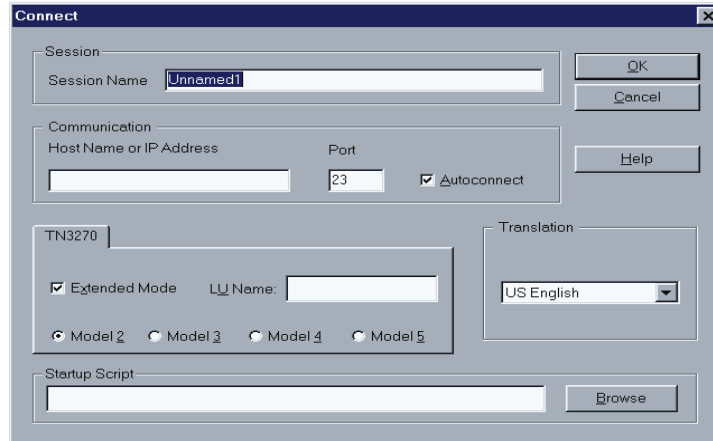
To start a session, select a session name and click Start. Double-clicking a session name also starts a session.



CREATING A SESSION

To create a new session:

1. Click on *Connect - Session Manager*.
2. Click the New button. The *Session Properties* dialog box for the new session is displayed.



3. Enter a new Session Name if desired. Otherwise the default name, Session X, is assigned.
4. Adjust property parameters as necessary.
5. Click OK. The new session name displays in the Create Sessions list.

COPYING A SESSION

The Copy feature is a quick way to create a session with similar properties to an existing session.

To copy a session:

1. Click on *Connect - Session Manager*.
2. Select a session to copy.
3. Click the *Copy* button. The *Session Properties* dialog box for the new session displays.
4. Enter a new Session Name if desired. Otherwise a default name, Copy X, is assigned.

5. Adjust property parameters as necessary.
6. Click *OK*. The new Session Name displays in the Create Session list.

DELETING SESSIONS

To delete a session:

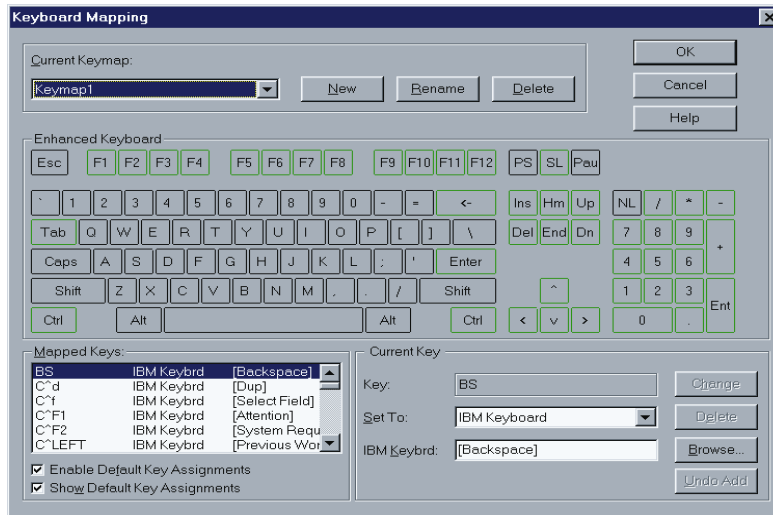
1. Click on *Connect - Session Manager*.
2. Select a session to delete.
3. Click the *Delete* button.

3

KEYBOARDING

KEY MAPPING

Through the *Keyboard Mapping* dialog box, the keyboard can be configured to perform many different functions. The emulator selects the default keyboard mapping based on the terminal tab (Tn3270 or Tn5250) selected in the *Connect* dialog box.



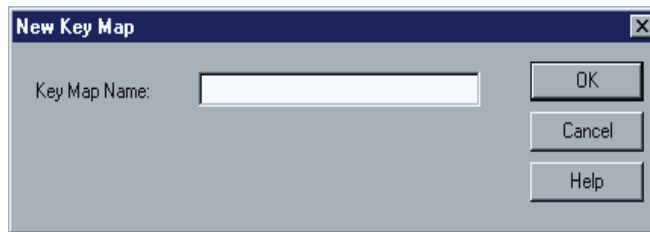
CURRENT KEYMAP

This section deals with the keymaps as whole entities. The name of the current keyboard map is listed in the Current Keymap list box.



New Button

Creates a new keymap. When this button is clicked, the Keymap Name dialog box appears. Enter the keymap name, then click OK.



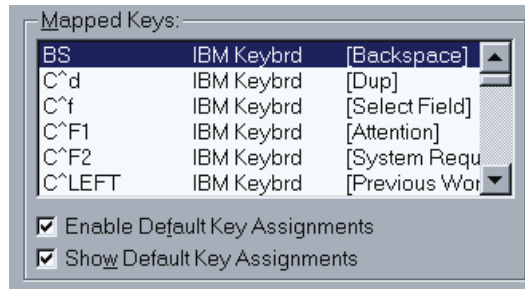
Rename Button

Changes the current keymap's name. When clicked, this button displays the *Keymap Name* dialog box. Change the keymap's name, then click OK.

Delete Button

Deletes the currently selected keymap. All of the data for the current keymap is deleted from the SETTINGS.INI file.

MAPPED KEYS



Enable Default Key Assignments

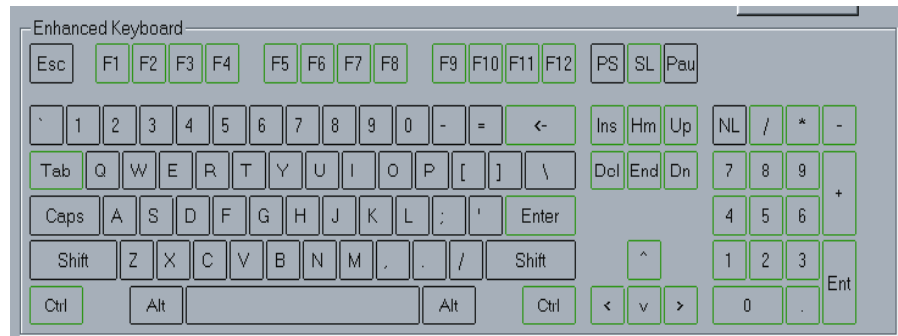
Disables the default definitions of the keys. Each predefined key definition is then set to UNMAPPED.

Show Default Key Assignments

Toggles the display of the key definitions in the Mapped Keys window. This option does not disable the definitions which will still display in the Current Key section.

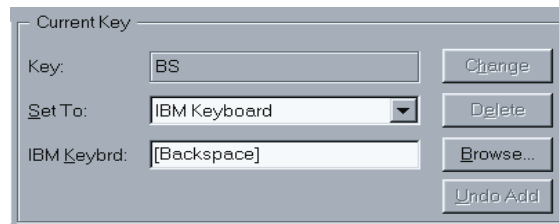
Note: See 3270 Default Keyboard Assignments, 5250 Default Keyboard Assignments, and Accelerator Keys.

KEYBOARD



The middle of the dialog box displays the layout for the default keyboard. To select a key to be defined, use the mouse to click on the desired key(s). You can precede each definition with Alt, Shift, Ctrl or any combination of the three.

CURRENT KEY



Each keystroke selection made in the keyboard section is displayed in this section.

Set To Options

IBM Keyboard

Nothing

String

Unmapped

IBM Keyboard

Performs a IBM keyboard function. Enter a valid name in the IBM Keybrd field or click Browse and select a function from the list.

Navigation Keys:

Enter	Tab
Back Tab	Home
Up Arrow	Down Arrow
Left Arrow	Right Arrow
Next Word	Previous Word
Move to EOT	

Edit Keys:

Backspace	Insert
Delete	Erase Input
Erase Field	Erase to EOF
Clear	Dup
Mark	New Line

PF Keys:

PF1-PF24

PA Keys:

PA1-PA3

Select Keys:

Select Up	Select Down
Select Left	Select Right
Select Word	Select Field

5250 Keys:

Roll Up	Roll Down
---------	-----------

Field Plus	Field Minus
Field Exit	Test Request
Help	

Other:

Reset	Attention
System Request	Print Screen
Cent Sign	Host Print

Nothing

Ignores any key action (disables the key definition). Nothing can be entered in the Nothing field.

String

Defines a string to be sent to the host. A simple ASCII string can be entered in the String field. To enter special characters, enclose the ASCII value in angle brackets.

Example: String: <027>OP or String: <ESC>OP

Both examples send the escape sequence Control [OP (^[OP).

String:<<BELL>>

Sends <BELL>. Double angle brackets prevent conversion to numeric values.

String:<%x44>

Converts the string from its Hex value to D.

3270 DEFAULT KEYBOARD ASSIGNMENTS

ENTER	Return or Enter
RESET	Left Control
PA1	PageUp
PA2	PageDown
PA3	Shift Ctrl F3
CLEAR	Scroll Lock
SYS REQ	Ctrl F2
ATTENTION	Ctrl F1
DUP	Ctrl D
FIELD MARK	Ctrl F
ERASE TO EOF	End
ERASE FIELD	Shift End
ERASE INPUT	Shift Ctrl Delete
INSERT	Insert
TAB	Tab
BACK TAB	Shift Tab
NEW LINE	Right Control
DELETE	Delete
BACKSPACE	Backspace
HOME Home	Arrow
UP	Up Arrow
DOWN	Down Arrow
LEFT	Left Arrow
RIGHT	Right Arrow
PF1	F1
PF2	F2
PF3	F3
PF4	F4
PF5	F5
PF6	F6

PF7	F7
PF8	F8
PF9	F9
PF10	F10
PF11	F11
PF12	F12
PF13	Shift F1
PF14	Shift F2
PF15	Shift F3
PF16	Shift F4
PF17	Shift F5
PF18	Shift F6
PF19	Shift F7
PF20	Shift F8
PF21	Shift F9
PF22	Shift F10
PF23	Shift F11
PF24	Shift F12
PRINT SCREEN	Ctrl P
CENT SIGN	Ctrl Z

5250 DEFAULT KEYBOARD ASSIGNMENTS

ENTER	Return or Enter
RESET	Left Control
PA1	PageUp
PA2	PageDown
PA3	Shift Ctrl F3
CLEAR	Pause
SYS REQ	Ctrl F2
ATTENTION	Ctrl F1

DUP	Ctrl D
FIELD MARK	Ctrl F
ERASE TO EOF	End
ERASE FIELD	Shift End
ERASE INPUT	Shift Ctrl Delete
INSERT	Insert
TAB	Tab
BACK TAB	Shift Tab
NEW LINE	Right Control
DELETE	Delete
BACKSPACE	Backspace
HOME	Home Arrow
UP	Up Arrow
DOWN	Down Arrow
LEFT	Left Arrow
RIGHT	Right Arrow
PF1	F1
PF2	F2
PF3	F3
PF4	F4
PF5	F5
PF6	F6
PF7	F7
PF8	F8
PF9	F9
PF10	F10
PF11	F11
PF12	F12
PF13	Shift F1
PF14	Shift F2
PF15	Shift F3
PF16	Shift F4
PF17	Shift F5
PF18	Shift F6
PF19	Shift F7
PF20	Shift F8

PF21	Shift F9
PF22	Shift F10
PF23	Shift F11
PF24	Shift F12
PRINT SCREEN	Ctrl P
CENT SIGN	Ctrl Z
+	Shift Keypad +
-	Shift Keypad -
Roll Down	Page Up
Roll Up	Page Down
Field Exit	Shift Return
Help	Shift Pause
Field Plus	Keypad Plus
Field Minus	Keypad Minus
Test Request	Control Shift Pause

ACCELERATOR KEYS

Capture Text to File

Alt C

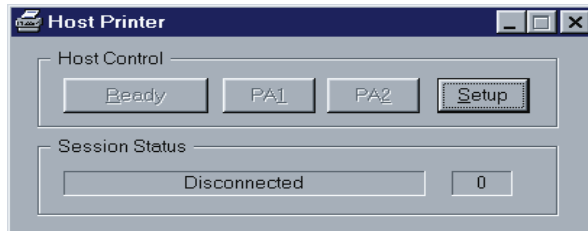
Print Ctrl P

4

PRINTING

HOST PRINT

Host printing is accomplished through an external application called Host Printer. Host Printer can be started from the Host Printer icon in the program group or by clicking *File - Host Printer*.

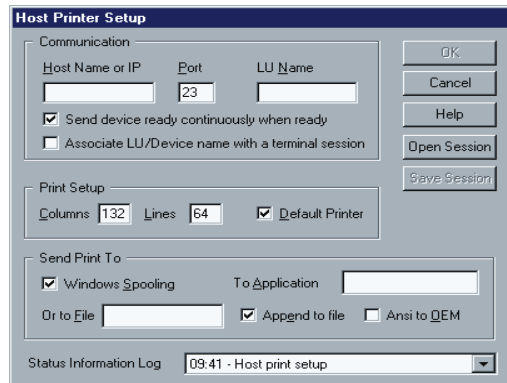


Host print (LU type 1 and 3) are supported according to RFC 1646. Host printing can be redirected to spooling an application or a file. The last host print session information is saved and used the next time Host Print is started.

The Host Print application has a status pane, a receive count pane and four push buttons: Ready, RA1, PA2, and Setup. Ready, RA1, and PA2 are only available when connected.

CONNECTING THE HOST PRINT SESSION

1. Configure the host print options by clicking on Setup to display the *Host Print Setup* dialog box.



The image shows the 'Host Printer Setup' dialog box. It is divided into three main sections: Communication, Print Setup, and Send Print To. The Communication section has fields for Host Name or IP, Port (set to 23), and LU Name. It also has checkboxes for 'Send device ready continuously when ready' (checked) and 'Associate LU/Device name with a terminal session' (unchecked). The Print Setup section has fields for Columns (132) and Lines (64), and a checked 'Default Printer' checkbox. The Send Print To section has a checked 'Windows Spooling' checkbox, a 'To Application' field, and an 'Or to File' field. It also has checkboxes for 'Append to file' (checked) and 'Ansi to OEM' (unchecked). On the right side, there are buttons for OK, Cancel, Help, Open Session, and Save Session. At the bottom, there is a 'Status Information Log' dropdown menu showing '09:41 - Host print setup'.

2. Enter the host name or IP Address.
3. If you are using a specific LU printing, the name of the LU Pool must be entered into the LU Name field. The LU Pool name is usually supplied by your system administrator.

To use generic LU printing leave the LU Name blank.

4. Set the number of lines and columns.
5. Configure where to send the printing information.
6. Click on *OK* to connect.

HOST PRINTER COMMAND LINE

Host Printer can be called from a command line with a session file as an argument.

Example:

“C:\Minisoft\MSforIBMHostPrint.exe Jupiter”

Starts Host Printer using the Jupiter.hpibm session file.

SAVING AND RETRIEVING A HOST PRINTER SESSION

For ease-of-use, Host printer sessions can be saved and retrieved later.

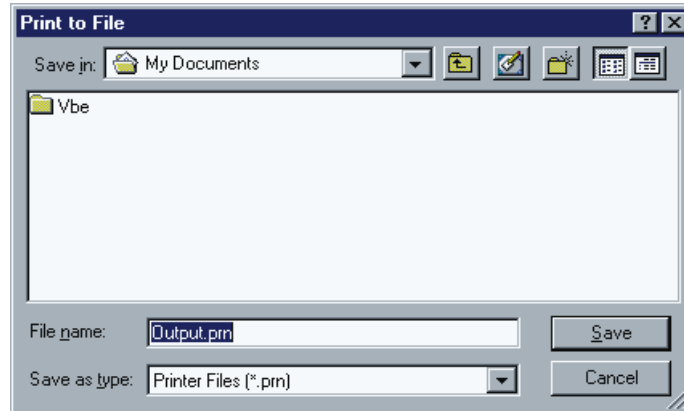
To save a host print session, click on *Save Session*. The *Save Host Print Session* dialog box appears. Enter or select a name for the session file and then click Save.

To retrieve a host print session, Click on *Open Session* and select the session file desired. Host Printer will connect automatically to the host when the session is opened.

PRINT TO FILE

The Print to File option is used to print information to a file rather than to a printer. When Print to File box is enabled and OK is clicked, the *Print to File* dialog box appears as shown below.

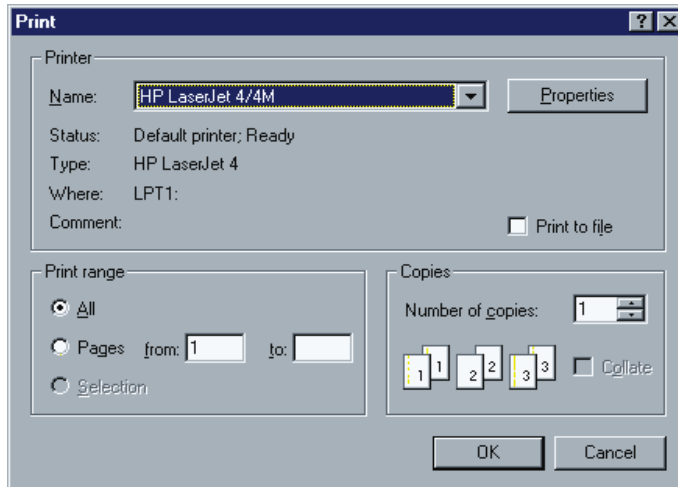
Select or enter the name of the file. The default extension .PRN is used.



PRINT SCREEN

To print a screen, click on *File - Print*. A standard Windows print dialog box appears.

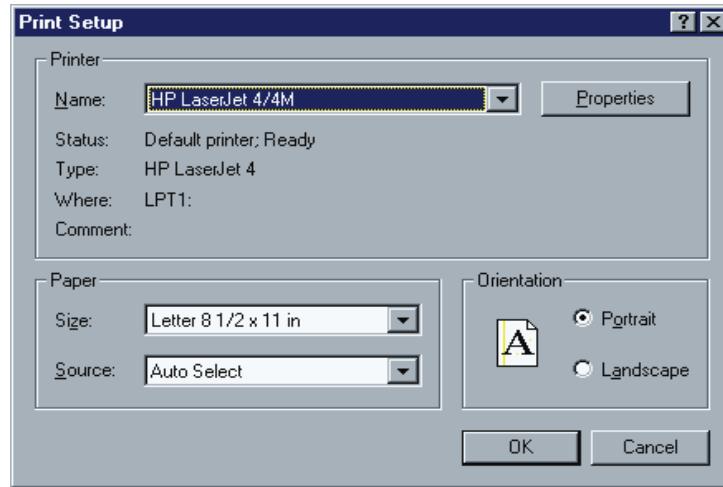
Click *OK* to print the screen.



Note: If Open Print Dialog On Print Screen has been enabled in the *Preferences* dialog, each print screen operation will start with this dialog box. If it has not been enabled, each print screen operation will go to the default printer.

PRINT SETUP

Choose printer and set printer options.

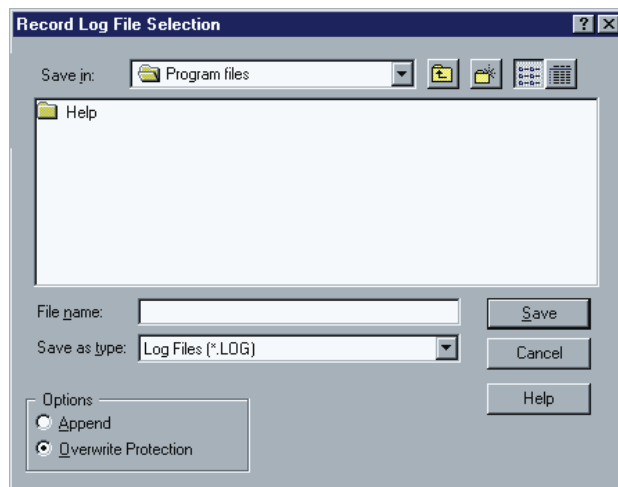


5

TECHNICAL REFERENCE

RECORD LOG FILE SELECTION

The Record Log feature records all data sent to the emulator from the host into a file on the PC.



Append

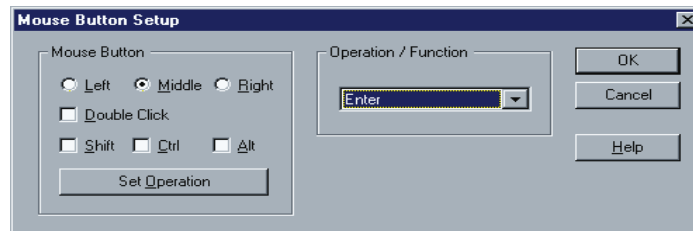
When the append mode is selected, the data recorded is appended to the end of the existing log file.

Overwrite Protection

When enabled, prompts for overwrite confirmation if the specified log file already exists.

MOUSE

The *Mouse* dialog box is used to setup left, middle, and right mouse button function/operations. The mouse buttons can be used with any combination of Alt, Shift, Ctrl, and double-click.



There are three options in this section to select:

- ◆ Mouse button.
- ◆ Clicking action: single click or double-click.
- ◆ Additional keys such as Ctrl, Alt, or Shift.

Set Operation

Once the configuration for the selected operation is selected, click Set Operation.

Operation/Function

Select an item to configure from the list:

Move Cursor	Enter
Tab	Delete Character
Erase Field	Erase to End of Field
Copy to Clipboard	Paste from Clipboard

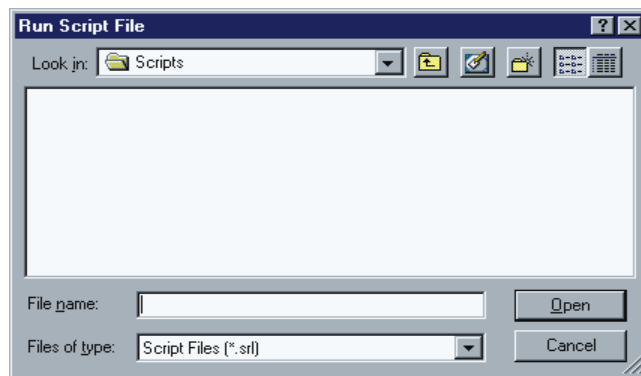
Select All	Select Word
Select Field	Print Image
Capture to File	Stop Capture to File
Start/Stop Recording	Start/Stop Script
Start/Stop Trace	Connect
Disconnect	PF1 - PF12
CLEAR	PA1 - PA3

6

SCRIPTING

RUNNING A SCRIPT FILE

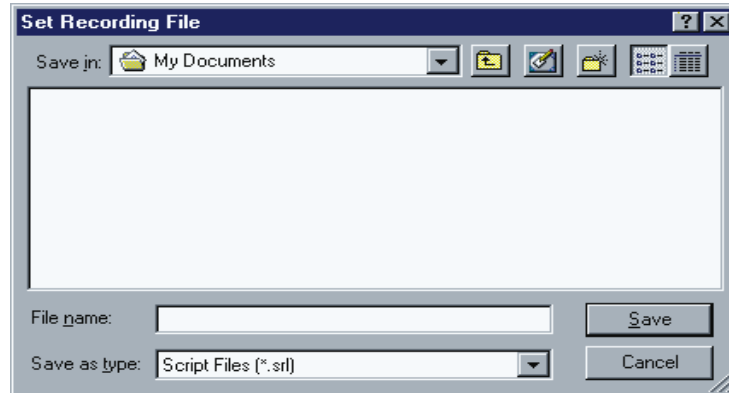
The *Run Script File* feature starts a script. This menu item is replaced with the *Stop Script* menu item when a script is running.



START RECORDING SCRIPT FILE

Starts the recording of keyboard events into a script file. Recording a script is a good way to automate repetitive tasks or create a base script file for a more complex task.

During recording, the letter R appears on the status line.



SCRIPT FUNCTIONS

The rules for script functions are as follows:

- ◆ Script function names are not case sensitive.
- ◆ Empty lines can be used.
- ◆ Comment lines must start with a #.

Script files will terminate if an error occurs. The line number where the error was detected displays on the Status Line. If the script terminates normally the text Script file closed displays on the status line.

Functions:

TypeString

Types the string parameter on the screen, starting from the current cursor

position. If the current input field is less than the string length, the rest of the string continues in the next input field. Script parameters from the command line can be used.

Example:

TypeString (“This is a text string”) or TypeString (“\$SP3”)

FunctionKey

Types a function key at current cursor position.

Function key names available:

ENTER	TAB
BACKTAB	HOME
UP	DOWN
LEFT	RIGHT
DELETE	NEWLINE
BACKSPACE	ERASETOEOF
ERASEFIELD	ERASEINPUT
ATTENTION	SYSREQ
CLEAR	INSERT
DUP	FMARK
RESET PA1	PA2
PA3	PF1 - PF24

Example:

FunctionKey (CLEAR)

WaitFor

Suspends the script until the event parameter has occurred.

Events available:

CONNECT (No wait if connected, otherwise wait until connected)

UNLOCK (Keyboard reset/unlock by host system)

KEYBOARD (Any keyboard input)

Example:

WaitFor (UNLOCK)

WaitForText

Suspends the script until the text string parameter has been found on the screen. The text is case sensitive.

Example:

WaitForText ("User name:")

WaitForTime

Suspends the script for the specified number of milliseconds. If no system timer is available this function terminates the script.

Example:

WaitForTime (3500)

WaitForCursorPos

Suspends the script until the cursor is in an exact screen position. The first parameter is the row position and the second parameter is the column position. The first screen position is row 1 and column 1.

Example:

WaitForCursorPos (20, 34)

TypeInput

Pops up an input dialog box with information text specified by the string parameter. The user input from the dialog box is handled by the TypeString function.

Example:

TypeInput ("Please type your password")

OpenSession

Replace the current session with a new session. The current session, if connected, is then disconnected.

If Autoconnect is enabled in the new session, the new session connects. All session parameters are replaced. The script file continues with the next script function.

Parameter: Session file name

Example:

OpenSession ("host.nmt")

OpenPrintSession

Starts a new instance of the emulator.

Parameter: Print session file name

Example:

OpenPrintSession ("hp.nmp")

ConnectHost

Starts a new connection. No other session setup parameters are changed. This button is grayed out when the emulator is not connected.

Parameter: Host name

Example:

ConnectHost ("MFRAME")

StartScript

Closes the current script at this position and starts a new script. The default script file type extension is SRL (Script Recording Language).

Example:

StartScript ("cics.srl")

StartApplication

Launches a Windows Application. Include any parameters necessary for the application after the application name. The script will terminate if the application does not start successfully.

Example:

StartApplication ("NOTEPAD hostcap.txt")

SetModelType

Set Model Type. There cannot be an established connection.

IBM-3278-2	model 2, no extended attributes
IBM-3278-3	model 3, no extended attributes
IBM-3278-4	model 4, no extended attributes
IBM-3278-5	model 5, no extended attributes
IBM-3278-2-E	model 2, extended attributes
IBM-3278-3-E	model 3, extended attributes
IBM-3278-4-E	model 4, extended attributes
IBM-3278-5-E	model 5, extended attributes

Example:

SetModelType (3)

SetExtendedMode

Set Extended Mode - TRUE or FALSE. There cannot be an established connection.

Example:

SetExtendedMode (TRUE)

SetTelnetPort

Set Telnet Port. There cannot be an established connection.

Example:

SetTelnetPort (1023)

SetLuName

Set LU Name. There cannot be an established connection.

Example:

SetLuName ("XYZNN")

Disconnect

Disconnect current session if connected. No session setup parameters are changed. There are no parameters.

Example:

Disconnect ()

StartCapture

Open a file for screen image capture. There are no parameters.

Example:

StartCapture ("hcap.txt")

StopCapture

Disable and close current capture file. There are no parameters.

Example:

StopCapture ()

SetAutoCapture

Enable or disable auto capture on attention keys.

Parameters: TRUE or FALSE.

Example:

SetAutoCapture (TRUE)

ReceiveHostFile

Transfer host file to PC.

Parameters: Local file name and host file name.

Example:

ReceiveHostFile ("C:\temp\info.txt", "hostfn")

SendLocalFile

Transfer local file to host.

Parameters: Local file name and host file name.

Example:

SendLocalFile ("C:\temp\info.txt", "hostfn")

CaptureImage

Capture current screen image to file. There are no parameters.

Example:

CaptureImage ()*TextInImage*

Test if a text sequence is in the current screen image. It can be used with the While/EndDo / If/EndIf block.

Example:

TextInImage ("MORE INFO")*NoTextInImage*

Test if a text sequence is not in the current screen image. It can be used with the While/EndDo / If/EndIf block.

Example:

NoTextInImage ("END OF DATA")

While/If-blocks can not be nested. But several If-blocks can be used in a While-block and several While-blocks can be used in an If-block.

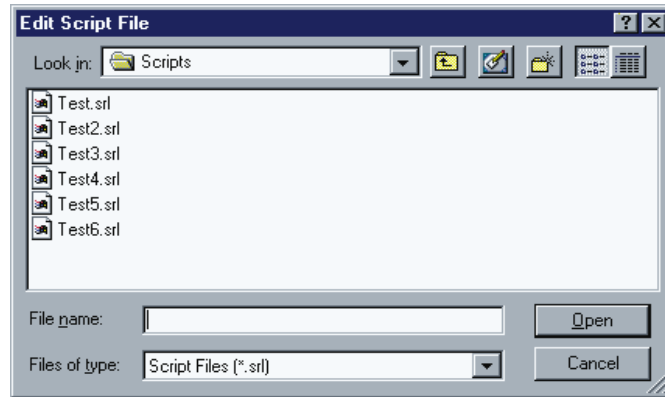
Example:

```
While TextInImage ("MORE INFO")
  # Other script functions
  CaptureImage ()
  FunctionKey (ENTER)
EndDo

If TextInImage ("END OF TEXT")
  StopCapture ()
  StartApplication ("notepad cap.txt")
EndIf
```

EDIT SCRIPT FILE

Edit a script file with the Notepad editor. See heading Script Functions



Minisoft for IBM supports the standard implementation of IBM's High-Level Language Application Program Interface known as HLLAPI. HLLAPI provides the means for external programs to communicate with IBM host computers via a 3270 terminal emulation program.

Windows HLLAPI provides the standard IBM HLLAPI style API for the 32-bit versions of Windows. This includes standard IBM HLLAPI style routines along with a set of Windows-specific extensions.

CONNECTING AN HLLAPI APPLICATION TO MINISOFT FOR IBM

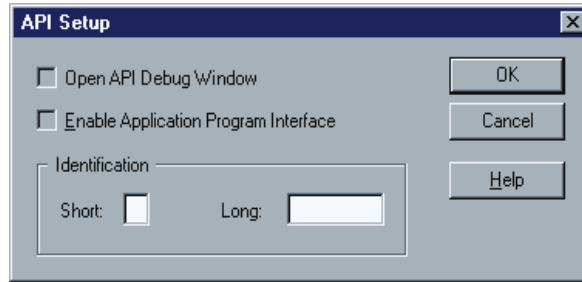
Multiple HLLAPI applications can connect to Minisoft for IBM. However, only one HLLAPI application can connect to one Minisoft for IBM session at a time.

Before you begin, ensure the Minisoft for IBM path is defined in your Autoexec.bat file or the IBMAPI.DLL file is local to your HLLAPI application.

To connect an HLLAPI application to Minisoft for IBM:

1. Start a Minisoft for IBM session.
2. Connect to the host.

3. In the Setup menu, click *API Setup* to open the API Setup dialog box:



4. Enter a short or long name parameter. Short is used by a HLLAPI application to identify a host session. Any single capital letter (A-Z) can be used as a short name value. There is no default value for this parameter. Standard HLLAPI applications use the letter E as a Short name value; some applications may use a different letter.

Note: If there is a value in the Short box in the HLLAPI names group, make sure it is the same HLLAPI short name that your application uses. If there is no value in this box, type one in.

HLLAPI Long name is for the convenience of the user – this is not used by the HLLAPI application. Values for the Long name box can be up to eight characters long and can include letters, numbers, and other characters. There is no default value for this parameter.

5. After entering the Short or Long name, the Open Debug Window and Enable Application Program Interface check boxes are enabled. Check Open Debug Window to display the API Debug window, which displays debugging information as calls to the API are made.

Check Enable Application Program Interface to enable HLLAPI for the session. Once enabled and connected, your HLLAPI application can connect to the session and begin using the API.

Note: The short name is a unique name used by an HLLAPI application to identify a session. Any single letter, from A to Z, can be used as a Short name

value. Case is not significant since Minisoft for IBM recognizes 'A' and 'a' as the same name.

6. Click OK to close the API Setup dialog box.
7. Switch out of Minisoft for IBM, and run your HLLAPI application.

To connect multiple HLLAPI applications to Minisoft for IBM, follow these steps for every HLLAPI application you wish to start.

HLLAPI FUNCTION REFERENCE

CONNECT PRESENTATION SPACE

Description:

Connect Presentation Space connects and links an HLLAPI application to an Presentation Space (PS). Minisoft for IBM allows up to 16 presentation spaces to be run simultaneously.

Prerequisite:

An presentation space must be configured with a unique short name before any HLLAPI applications can connect to it.

Call Parameters:

Your client application program must pass the following parameters to this function:

Function number	1
Data string	A letter of the alphabet (A-Z) which represents the presentation space short name.
Data length	Not applicable.
PS position	Not applicable.

Return Codes:

0 APIOK	Function was successfully completed. The presentation space is ready for input.
1 APIDISCONNECTED	The specified presentation space is invalid. API Setup dialog for the correct HLLAPI short name.
4 APIBUSY	Successful connection, but presentation space is busy.
5 APILOCKED	Successful connection, but presentation space is locked.
9 APISYSTEMERROR	System Error.
11 APIUNAVAILABLE	Presentation space is in use by another session and unavailable.

*DISCONNECT**Description:*

Disconnect Presentation Space (2) disconnects your client application from its currently connected presentation space.

After calling this function, you cannot call functions that depend on a connection to a presentation space: You will be automatically disconnected from your currently connected presentation space when you connect to another presentation space.

Your client application should always disconnect when exiting.

Prerequisite:

Connect Presentation Space (1)

Call Parameters:

Function Number	2
Data String	Not applicable
Length	Not applicable
PS Position	Not applicable

Return Codes:

0 APIOK	Successful disconnect
1 APIDISCONNECTED	Client is currently not connected to any presentation space
9 APISYSTEMERROR	System error

SEND KEY*Description*

Send Key (3) is used to send a string of up to 255 keystrokes to the currently connected presentation space as if they had been entered by an operator. The keyboard must be unlocked and not in a busy state. Send Key (3) allows sessions to exchange keystrokes that are not available from the keyboard, or are not represented by an ASCII value. For example, an operator may not be able to produce the keystrokes necessary for a session, so Send Key (3) can produce the appropriate keystroke.

Prerequisites

Connect Presentation Space (1).

Call Parameters

Function Number	3
Data String	The string can be a maximum of 255 characters including

keystrokes and ASCII mnemonics which the client sends to host presentation space.

PS Position

Not applicable

Length

String length, if STRLEN is used. Not used if STREOT option is used and in such case, the string must end with an EOT.

Return Codes

0 APIOK

The keystrokes were successfully sent to the host presentation space.

1 APIDISCONNECTED

Your client is not connected to presentation space.

2 APIINVALIDPARAMETER

An incorrect parameter was sent to HLLAPI.

4 APIBUSY

Host presentation space busy, not all keystrokes could be sent.

5 APILOCKED

Host presentation space inhibited, not all keystrokes could be sent.

9 APISYSTEMERROR

System error.

WAIT

Description

The Wait (4) function determines whether the presentation space can accept keystrokes from a HLLAPI application.

Prerequisite

Connect Presentation Space (1)

Call Parameters

Function Number	4
Data String	Not applicable
Length	Not applicable
PS Position	Not applicable

Return Codes

0 APIOK	Function was successful. The keyboard is unlocked and ready for input.
1 APIDISCONNECTED	The client application is not connected to a valid presentation space.
4 APIBUSY	Timed out on XXXX.
5 APILOCKED	Keyboard is locked.
9 APISYSTEMERROR	System error.

*COPY PRESENTATION SPACE**Description*

The Copy Presentation Space (5) function copies the contents of the entire presentation space into a string buffer. The size of the buffer will be the size of the presentation space unless you decide to copy extended attributes (EABs), in which case it will be more. If you copy EABs, the string length must be at least twice the size of the presentation space for the EAB values to fit into the buffer.

Prerequisite

Connect Presentation Space (1)

Call Parameters

Function Number	5
-----------------	---

Data String	The data buffer defined in the client application program should be as large as the presentation space of the entire display buffer. If EABs are set up to be copied, this buffer must be at least double the size for the model.
Length	Not applicable.
PS Position	Not applicable.

Return Parameters

Contains the entire contents of the presentation space. The size of this parameter depends on the value you entered in the Data String parameter.

Return Codes

0 APIOK	Function was successful. HLLAPI copied the contents to the client application buffer.
1 APIDISCONNECTED	The client application is not connected to the presentation space.
4 APIBUSY	The contents were copied to the buffer. The presentation space is waiting for a response.
5 APILOCKED	The contents were copied to the buffer; keyboard is locked.
9 APISYSTEMERROR	System error.

SEARCH PRESENTATION SPACE

Description

The Search Presentation Space (6) function looks for the specified string. The search can start from a specified position on the presentation space and scan forward or backward from that position. The method of scanning is specified by Set Session Parameters (9), SRCHALL, SRCHFROM, SCRCHFRWD, SRCHBKWD.

This function is useful if you are looking for a particular keyword on a host screen.

Prerequisite

Connect Presentation Space (1)

Call Parameters

Function Number	6
Data String	Data string to search for, which you define in the client program. The string is case sensitive.
Length	Target buffer length.
PS Position	Presentation space position where to start the search function (SRCHFRWD), or to end (SRCHBKWD). If SRCHALL is set, this parameter is ignored.

Return Parameters

This function returns a length parameter.

Length:

- 0 The Search function did not find the string.
- >0 The returning length is the starting presentation position where the string was found.

Return Codes

0 APIOK	The function was successful (string was found).
1 APIDISCONNECTED	Your client application is not connected to a host presentation space.
2 APIINVALIDPARAMETER	Incorrect parameter.

7 APIINVALIDCURSORPOS	Invalid presentation space position.
9 APISYSTEMERROR	System error.
24 APIUNFORMATTED	The function could not find the string.

QUERY CURSOR POSITION

Description

The Query Cursor Position (7) function returns the cursor position in the currently connected presentation space.

Prerequisite

Connect Presentation Space (1)

Call Parameters

Function Number	7
Data String	Not applicable
Length	Not applicable
PS Position	Not applicable

Return Parameter

Length Presentation space position of the cursor location.

Return Codes

0 APIOK	The function was successful.
1 APIDISCONNECTED	Your application program is not connected to a presentation space.
9 APISYSTEMERROR	System error.

COPY PRESENTATION SPACE TO STRING

Description

The Copy Presentation Space (8) to String function copies the presentation space to a string in your client application.

Prerequisite

Connect Presentation Space (1)

Call Parameters

Function Number	8
Data String	Defined data string to hold the entire content of the presentation space. Since this function copies from the presentation space to a string, be sure that the buffer is large enough to hold the data specified in the length parameter.
Length	Length of the target data string.
PS Position	Position where the copying should begin.

Return Parameter

Data String	Contents of the copied string from the presentation space.
-------------	--

Return Codes

0 APIOK	The presentation space was copied to the pre-allocated buffer in your client application program. The keyboard is unlocked.
1 APIDISCONNECTED	Your application program is not connected to a valid presentation space.

2 APIINVALIDPARAMETER	The string length specified in your application is incorrect.
4 APIBUSY	The content was copied. The presentation space is waiting for a host response.
5 APILOCKED	The content was copied. The keyboard is locked.
7 APIINVALIDCURSORPOS	Your application program calculated an invalid presentation space position.
9 APISYSTEMERROR	System error.

SET SESSION PARAMETERS

Description

The Set Session Parameters function lets the HLLAPI client program specify default values for several HLLAPI functions.

The values are divided into a number of sets which affect the HLLAPI functions.

Values are in effect until one of the following events occurs:

- ◆ A new call to Set Session Parameters (9) defines a new value.
- ◆ The session parameters are reset using Reset System (21) function.

Prerequisites

None.

Call Parameters

Function Number	9
Data String	A text string containing the session parameters. These parameter can be contained in a list delimited by commas or spaces.

Length	Length of source data string.
PS Position	Not applicable.

NOATTR/ATTR

Specifies whether to translate character attributes to ASCII values or not. This parameter affects Copy Presentation Space (5), Copy Presentation Space to String (8), and Copy Field String (34) functions.

<i>Parameter</i>	<i>Description</i>
ATTRB	Keep original values in their original EBCDIC code.
NOATTRB	(default) Translates non-ASCII values to blanks (20H).

EAB/NOEAB

Specifies if a copy function should include extended attributes (EABs). These functions are used with Copy Presentation Space (5), Copy Presentation Space to String (8), Copy String to Presentation Space (15), and Copy Field to String (34).

<i>Parameter</i>	<i>Description</i>
EAB	Data string includes extended attribute bytes (EABs), producing two characters for each screen character. Allocate a buffer twice the size of the presentation space.
NOEAB	EABs are not included in data string. Allocate the size of the presentation space (default).

STREOT/STRLEN

Determines how the length of a data string is determined.

<i>Parameter</i>	<i>Description</i>
STRLEN	(default) The client application copies a string with an explicit length.

STREOT Strings end with a character defined in the EOT parameter.

AUTORESET/NORESET

This parameter is used by Send Key (3) and Get Key (51).

<i>Parameter</i>	<i>Description</i>
AUTORESET	(default) Automatically sends Reset with strings with Send Key (3) function to unlock the input inhibited condition that might exist.
NORESET	Does not send a Reset in a data string.

SEARCHALL/SRCHFROM

Specifies how HLLAPI should search the presentation space. These parameters are used with Search Presentation Space (6) and Search Field (30).

<i>Parameter</i>	<i>Description</i>
SRCHALL	Searches the entire presentation space or field (default).
SRCHFROM	Searches from a specified beginning location of the presentation space or field.

SEARCHFRWD/SRCHBKWD

<i>Parameter</i>	<i>Description</i>
SCRCHFRWD	Searches forward through the presentation space (default).
SRCHBKWD	Searches backward through the presentation space.

Wait Parameter

Specifies the characteristics of a wait period.

<i>Parameter</i>	<i>Description</i>
------------------	--------------------

TWAIT	(default) For Wait (4), waits up to 60 seconds before timing out on XCLOCK or XSYSTEM. For Get Key (51) TWAIT waits until a key is in the buffer (normal or AID key based on Start Keystroke Intercept (50) option, D or L), and then returns control to the client HLLAPI program.
NWAIT	No waiting period. For Wait (4), checks status and returns immediately. For Get Key (51) immediately, keystroke not available.

FPAUSE/IPAUSE

Specifies type of pause. This parameter is used by the Pause (18) function.

Parameter Description

FPAUSE	Full pause. The HLLAPI client program can set the duration of the pause in the Pause (18) function (default).
IPAUSE	Interruptible pause; Start Host Notification (23) or any host system event will end a pause.

QUERY SESSIONS

Description

The Query Sessions function returns a 12 byte data string for each started presentation space. The data string contains information about presentation space names and size of the presentation space. It also returns a data length parameter which holds the number of active presentation spaces.

Prerequisites

None.

Call Parameters

Function Number	10
-----------------	----

Data String	Data string for session parameters.
Length	12 bytes * the number of started host presentation spaces. Max size is 12 * 16 (192).
PS Position	Not applicable.

Return Parameters

Length The number of started host presentation spaces.

Each 12-byte string contains:

1	Presentation space short name.
2-9	Presentation space long name.
10	'H' = host.
11-12	Presentation space size. For example, an IBM model 2 is 1920.

Return Codes

0 APIOK	Successful
2 APIINVALIDPARAMETER	Invalid string length
9 APISYSTEMERROR	System error

RESERVE PRESENTATION SPACE

Description

The Reserve function blocks an operator from entering any information on the presentation space via the keyboard. You should use this function if you need total control of the presentation space and need to prevent other input.

Prerequisite

Connect Presentation Space (1).

Call Parameters

Function Number	11
Data String	Not applicable.
Length	Not applicable.
PS Position	Not applicable.

Return Codes

0 APIOK	The Reserve function was successful.
1 APIDISCONNECTED	Your client application is not connected to an presentation space.
9 APISYSTEMERROR	System error.

RELEASE PRESENTATION SPACE*Description*

The Release function unlocks a presentation space that was locked using the Reserve (11) function.

Prerequisite

Connect Presentation Space (1) and Reserve (11)

Call Parameters

Function Number	12
Data String	Not applicable.
Length	Not applicable.
PS Position	Not applicable.

Return Codes

0 APIOK	The Release function was successful.
---------	--------------------------------------

1 APIDISCONNECTED	Your client application is not connected to an presentation space.
9 APISYSTEMERROR	System error.

COPY OPERATOR INFORMATION AREA

Description

The Copy Operator Information Area function returns the contents of the operator information area (OIA) from the currently connected presentation space. A 103-byte string returns the data to your client HLLAPI application.

Prerequisite

Connect Presentation Space (1)

Call Parameters

Function Number	13
Data String	A 103 byte long data string, pre-allocated in your application program.
Length	103 bytes.
PS Position	Not applicable.

Return Parameters

Data String: Data string containing the contents of the identified OIA. The OIA data string contains the following information:

Byte:

1	OIA Format Byte. This byte is always 1 for 3270.
2 - 81	OIA Image Groups. These bytes contain hexadecimal codes of the OIA image symbols.

82 - 103 The OIA group indicators. Can be an ASCII character string translatable into graphics characters.

Return Codes

0 APIOK	OIA copying is successful. Presentation space is unlocked for input.
1 APIDISCONNECTED	The HLLAPI application is not connected to a valid presentation space.
2 APIINVALIDPARAMETER	The HLLAPI application passed an invalid string length. The OIA could not be copied.
4 APIBUSY	The OIA was copied, but presentation space is busy.
5 APILOCKED	The OIA was copied, presentation space is locked.
9 APISYSTEMERROR	System error.

COPY STRING TO PRESENTATION SPACE

Description

The Copy String to Presentation Space (15) function copies an ASCII data string to a defined position on the presentation space.

Prerequisite

Connect Presentation Space (1)

Call Parameters

Function Number	15
Data String	A string of ASCII characters you want to copy into an unformatted position on presentation space.

Length	Contains the total length of the string to copy, including EABs, if the EAB parameter is selected.
PS Position	Presentation space position where copying begins.

Return Codes

0 APIOK	The Copy String to Presentation space (15) function was successful. The string was copied.
1 APIDISCONNECTED	Copy not configured or the client application does not have a connection.
2 APIINVALIDPARAMETER	Parameter error.
5 APILOCKED	The presentation space is prohibited, field is protected, or non-ASCII data was sent to the presentation space.
6 APITRUNCATED	The function was successful, the string was copied, but data is truncated at end of screen, or end of field.
7 APIINVALIDCURSORPOS	The presentation space position is invalid. Check the size of the presentation space.
9 APISYSTEMERROR	System error.

PAUSE

Description

The Pause (18) function is used to add a pause to a host session sequence. The function halts for a specified amount of time, or until a host event satisfies the pause.

Prerequisites

A Start Host Notification (23) function call is required if the IPAUSE parameter is selected using Set Session Parameters (9).

Call Parameters

Function Number	18
Data String	Not applicable.
Length	Sets the duration of the pause. This parameter must be an integer between 0 and 65535. The pause is set in half-second increments. For example, 1 = 0.5 seconds, 60 = 30 seconds, 120 = 1 minute (60 seconds), etc.
PS Position	Not applicable.

Return Codes

0 APIOK	Pause duration has expired.
9 APISYSTEMERROR	System error.
26 APIHOSTUPDATE	Presentation space or OIA has been updated. Use Query Host Update (24) for more information.

*QUERY SYSTEM**Description*

The Query System (20) function lets your HLLAPI application determine settings and other values related to the system. The function returns a 35-byte long string with HLLAPI-related information such as version number, level number, date, etc.

Prerequisites

None.

Call Parameters

Function Number	20
Data String	A 35-byte data string. The return parameter contains the system information.
Length	Not applicable (35 is implied).
PS Position	Not applicable.

Return Parameter

This function returns a 35 byte long string with system information.

*Data String**Bytes*

1	HLLAPI version number.
2-3	HLLAPI level number.
4-9	HLLAPI version date (mmddy).
10-12	Not applicable.
13	Hardware Base (always 'U').
14-35	Not applicable.

Return Codes

0 APIOK	The Query System function was successful.
1 APIDISCONNECTED	HLLAPI not connected to presentation space.

2 APIINVALIDPARAMETER	Parameter error, Invalid string length. You must allocate a 35-byte string to receive the returning data string parameter.
9 APISYSTEMERROR	System error.

RESET SYSTEM

Description

The Reset System (21) function reinitializes the HLLAPI to its starting state. This function can also be used to ensure that HLLAPI is loaded.

You can use this function when you start or terminate your HLLAPI program.

Prerequisites

None.

Call Parameters

Function Number	21
Data String	Not applicable.
Length	Not applicable.
PS Position	Not applicable.

Return Codes

0 APIOK	The Reset function was successful.
1 APIDISCONNECTED	HLLAPI not loaded. This return code is useful to insure that HLLAPI is loaded.
9 APISYSTEMERROR	System errors.

QUERY SESSION STATUS

Description

The Query Session Status (22) function returns information about a presentation space. The function returns the status information in an 18-byte data string.

The following information is included in the return parameter of Query Session Status (22).

- Presentation space names
- Session type
- EAB and PSS information
- Session usage
- Presentation space row/column information
- Host code number

Prerequisites

None.

Call Parameters

Function Number	22
Data String	18 bytes.

The first byte must be either:

1. A letter A-Z, which indicates the presentation space short name.
2. A " " (space) which indicates the currently connected presentation space.

Length	18 bytes.
PS Position	Not applicable.

Return Parameters

Data String:

- 1 Presentation space short name.
- 2 - 9 Presentation space long name.
- 10 Presentation space type information.
 How this is implemented.
 D = 3270 Host.
 F = 5250 Host.
- 11 Presentation space Characteristics.
- Bit 0: 0 = No EABs
 1 = EABs are on
- 1-7: Not used
- 12 - 13 Binary value. Number of presentation space rows.
- 14 - 15 Binary value. Number of presentation space columns.
- 16 - 18 Not applicable.

Return Codes

- | | |
|-----------------------|--|
| 0 APIOK | Query Session Status function was successful. |
| 1 APIDISCONNECTED | Invalid presentation space short name |
| 2 APIINVALIDPARAMETER | Invalid string parameter length. Length must be set to 18. |
| 9 APISYSTEMERROR | System error. |

*START HOST NOTIFICATION**Description*

The Start Host Notification (23) function starts monitoring the presentation space or operator information area for updates. Following this call, your application must call the Query Host Update (24) function to check when the host updated the presentation space or OIA.

Prerequisites

None.

Call Parameters

Function Number	23
Data String	A six-byte long string.
1	presentation space short name, or blank or null (request against the currently connected presentation space).
2	One of the following characters: B = Both. Notification of OIA and presentation space updates. O = Notification of any OIA updates. P = Notification of any presentation space updates.
3 - 6	Ignored by HLLAPI. No error is caused if the program has data in these positions.
Length	6
PS Position	Not applicable.

Return Codes

0 APIOK	The Start Host function was successful.
---------	---

1 APIDISCONNECTED	Invalid presentation space ID.
2 APIINVALIDPARAMETER	Invalid parameters.
9 APISYSTEMERROR	System error.

QUERY HOST UPDATE***Description***

The Query Host Update (24) function allows your application to determine if the host has updated the presentation space, or operator information area (OIA) since:

Start Host Notification (23)

Previous Query Host Update (24) function was called.

Prerequisites

Your application must call the Start Host Notification (23) function before calling this function.

Call Parameters

Function Number	24
Data String	One character A-Z short name for the presentation space. A space indicates a request for updates to the requested presentation space.
Length	Not applicable (1 is implied).
PS Position	Not applicable.

Return Codes

0 APIOK	No updates have been made since last function call.
1 APIANOTCONNECTED	Invalid presentation space ID.

8 APINOTAVAILABLE	Your application needs to call Start Host Notification (23).
9 APISYSTEMERROR	System error
21 APIOIAUPDATED	The OIA was updated.
22 APIPSUPDATED	The presentation space was updated.
23 APIOIAPSUPDATE	Both OIA and presentation space were updated.

STOP HOST NOTIFICATION

Description

The Stop Host Notification (25) function disables the capability of the Query Host Update (24) function to determine if the host presentation space or OIA has been updated. This function can also be used to stop host events from affecting the Pause function.

Prerequisites

Start Host Notification (23)

Call Parameters

Function Number	25
Data String	Short name of the presentation space or blank or null indicating a request for the current presentation space.
Length	Not applicable (1 is implied).
PS Position	Not applicable.

Return Codes

0 APIOK	Stop Host Notification was successful.
---------	--

1 APIANOTCONNECTED	Invalid presentation space short name.
8 APINOTAVAILABLE	No Start Notification was issued.
9 APISYSTEMERROR	System error.

SEARCH FIELD

Description

The Search Field (30) function searches for a text string in protected or unprotected fields of the presentation space. When the specified text string is found, the function returns the presentation space position.

This function is case-sensitive and requires an exact match of the text string parameter and the presentation space content.

Prerequisite

Connect Presentation Space (1)

Call Parameters

Function Number	30
Data String	Text string to search presentation. End the string with EOT if the client uses STREOT. See Set Session Parameters (9).
Length	Specifies the length of the string you are searching. Not applicable if EOT is used to end the string.
PS Position	Specifies the beginning presentation space position to search from. If SRCHFROM is set, it also specifies the position within the field.

Return Parameters

Length:

- 0 String was not found. The returning length is set to 0.
- > 0 If the returning length is greater (>) than 0, the returning length is the starting presentation space position where the string was found.

Return Codes

0 APIOK	The Search Field (30) function was successful.
1 APIDISCONNECTED	Invalid presentation space position.
2 APIINVALIDPARAMETER	Parameter error. String length set to zero or an EOT character was not found in the calling data string.
7 APIINVALIDCURSORPOS	Invalid presentation space position.
9 APISYSTEMERROR	System error.
24 APIUNFORMATTED	String not found.

FIND FIELD POSITION

Description

The Find Field Position (31) function returns the position of a target field at the currently connected presentation space. This function is useful for searching for either protected or unprotected fields in a field formatted presentation space.

Prerequisite

Connect Presentation Space (1)

Call Parameters

Function Number	31
Data String	Defines the field type. [space] [space] This field (“ ”) T[space] This field (“T ”) N[space] Next field (“N ”) either protected or unprotected P[space] Previous field (“P ”) either protected or unprotected
NP	Next protected field (“NP”)
NU	Next unprotected field (“NU”)
PP	Previous protected field (“PP”)
PU	Previous unprotected field (“PU”)
Length	Not applicable (2 is implied)
PS Position	Contains the presentation space position in a field where the function starts the search.

Return Parameters

Length:

- 0 If return code 28: No such field found. If return code 24: unformatted host presentation space.
- > 0 The returning length is the starting presentation space position where the field was found, defined as the first position after the attribute byte.

Return Codes

0 APIOK	The Find Field Position (31) function was successful
1 APIDISCONNECTED	Not connected to an presentation space
2 APIINVALIDPARAMETER	Parameter error

7 APIINVALIDCURSORPOS	Invalid presentation space position
9 APISYSTEMERROR	System error
24 APIUNFORMATTED	No such field or presentation space not formatted
28 APIZEROLENFIELD	Field length = 0

COPY STRING TO FIELD

Description

The Copy String to Field (33) function copies a string of characters, defined in your client application, into a specific unprotected field in the presentation space. It can only be used in a field-formatted presentation space with unprotected fields.

Prerequisite

Connect Presentation Space (1).

Call Parameters

Function Number	33
Data String	String of data in your application program you want to copy to the presentation space. End string with EOT if STREOT is specified. See Set Session Parameters (9).
Length	Length of data. Applicable only if STRLEN is used.
PS Position	Position on presentation space where the string will be copied. The function transfers the string and places it in the beginning of the unprotected field.

Return Codes

0 APIOK	The Copy String to Field (33) was successful, the string was copied.
---------	--

1 APIDISCONNECTED	Not connected to presentation space.
2 APIINVALIDPARAMETER	Parameter error.
5 APILOCKED	The presentation space field was protected or locked or non-ASCII characters was sent to the field.
6 APITRUNCATED	Copied; data truncated. The field may be shorter than the copied string.
7 APIINVALIDCURSORPOS	Invalid presentation space position.
9 APISYSTEMERROR	System error.
24 APIUNFORMATTED	Unformatted host presentation space.

COPY FIELD TO STRING

Description

The Copy Field to String (34) function copies ASCII characters from a protected or unprotected field in the presentation space to a string defined in your application. The presentation space must be field-formatted.

Prerequisite

Connect Presentation Space (1)

Call Parameters

Function Number	34
Data String	String to which the program copies the contents of the field.
Length	Length of the field. If the field contains EABs, then the length must be twice the length of the field.

PS Position Any position within the field of the presentation space. The copying will always start from the beginning of the field.

Return Parameter

Data String: This string contains the data that was copied from the field. The first byte of the returned string contains the start byte of the field in the presentation space. The length of this string is either the number of bytes in the specified field , or the number of bytes specified in the Length parameter. If the Length parameter is shorter than the actual string, the function returns an error code (6), and the string is truncated.

Return Codes

0 APIOK	The Copy Field to String (34) function was successful.
1 APIDISCONNECTED	Not connected to an presentation space.
2 APIINVALIDPARAMETER	Parameter error.
6 APITRUNCATED	Data string and field lengths are not equal size. Data is truncated.
7 APIINVALIDCURSORPOS	Invalid presentation space.
9 APISYSTEMERROR	System error.
24 APIUNFORMATTED	Unformatted presentation space.

SET CURSOR POSITION

Description

The Set Cursor Position (40) function puts the cursor at a desired location in the presentation space.

Prerequisite

None.

Call Parameters

Function Number	40
Data String	Not applicable.
Length	Not applicable.
PS Position	Desired cursor position in the presentation space.

Return Codes

0 APIOK	The Set Cursor (40) function positioned the cursor successfully at the specified position.
1 APIDISCONNECTED	Not connected to a presentation space.
4 APIBUSY	presentation space is busy.
7 APIINVALIDCURSORPOS	Invalid presentation space position. The cursor location was outside the boundaries of the presentation space.
9 APISYSTEMERROR	System error.

*SEND FILE**Description*

The Send File (90) function transfers a file from your PC to the host presentation space using the IND\$FILE host file transfer function.

Prerequisite

Connect Presentation Space (1).

Call Parameters

Function Number	90
Data String	Send command string. Maximum 128 bytes.
Length	Length of parameters string.
PS Position	Not applicable.

Return Codes

2 APIINVALIDPARAMETER	Parameter error.
3 APIFTCOMPLETE	File transfer complete.
9 APISYSTEMERROR	System error.
302 APIFTFILENOTFOUND	File not found.
303 APIFTNOACCESS	Access denied.

RECEIVE FILE*Description*

The Receive File (91) function transfers a file from the host presentation space to the PC using the IND\$FILE host file transfer function.

Prerequisites

None.

Call Parameters

Function Number	91
Data String	Receive command string. Maximum 128 bytes.
Length	Length of parameters string.

PS Position Not applicable.

Return Codes

2 APIINVALIDPARAMETER	Parameter error.
3 APIFTCOMPLETE	File transfer complete.
9 APISYSTEMERROR	System error.
302 APIFTFILENOTFOUND	File not found.
305 APIFTNOACCESS	Access denied.

CONVERT POSITION

Description

The Convert (99) function converts a position on the presentation space to a display row/column coordinate, or a display row/column coordinate to an presentation space position.

Prerequisites

None.

Call Parameters

Function Number	99
Data String	2-byte data string.

1. Use one of the following:

a letter A-Z, which indicates presentation short name.

a “ “ (space) which indicates the currently connected presentation space.

2. P or R:

P = converts from Position to Row/Column.

R = converts from Row/Column to Position. See Length and PS Position parameter below.

Length	Row number on the presentation space. Valid row value is between 1 and the maximum number of rows depending the model. Not applicable when P is specified in the String parameter.
PS Position	When R is specified, column number on presentation space. Valid column value is between 1 and the max presentation space position, usually 1920. When P is specified, the presentation space position. A valid value must be between 1 and maximum presentation space buffer size.

Return Parameters

The Convert (99) function does not return parameters.

Return Codes

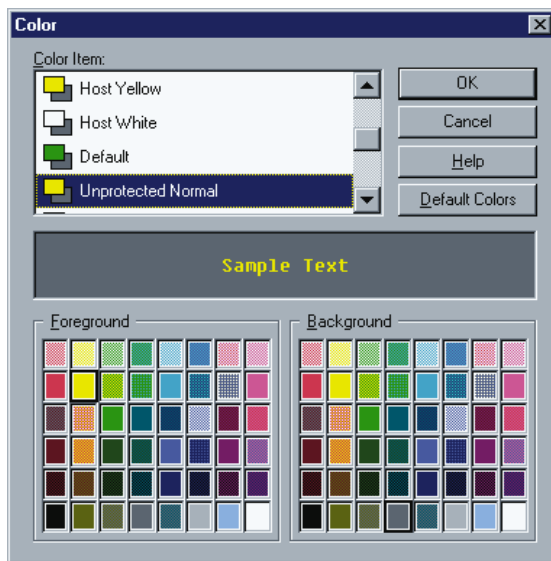
0	Incorrect presentation space position or column.
> 0	The converted presentation space position or column.
9998 APICINVALIDPS	Invalid presentation space short name.
9999 APICINVALIDOPT	Invalid character in the Length parameter. Letter is not P or R.

8

COLOR

COLOR

The Color dialog box controls the way that the text and background colors appear on the screen.



Color Item

To change a color, first select the item. The text color and background color display in the color bar. Select a new foreground or background if desired. When you have finished, you can continue with additional color items or click *OK* to save your changes and exit the dialog box.

Color Bar

Displays the colors for the selected color item.

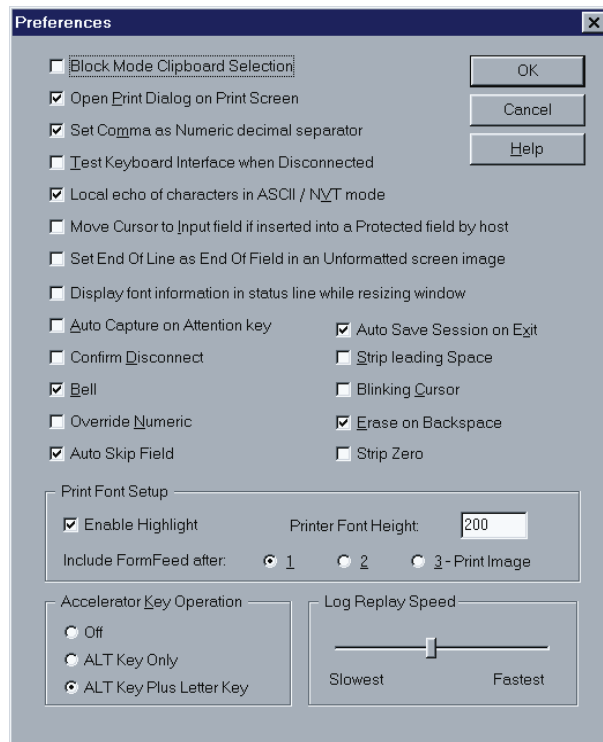
Default Colors Button

Resets all color items to the default colors.

9

PREFERENCE

The *Preferences* dialog box contains key miscellaneous options.



PREFERENCES

Block Mode Clipboard Selection

If enabled, the block of text, starting at the beginning of any selected line and ending at the end of any selected line, is copied to the Clipboard.
If disabled, only the selected characters are copied to the Clipboard.

Open Print Dialog on Print Screen

Opens the *Print* dialog box before each *File - Print* operation. If disabled, the default printer and setup is used without first prompting the *Print* dialog box.

Set Comma as Numeric decimal separator

If enabled a comma is allowed as a numeric decimal point. If disabled, input stops when the comma is entered.

Test Keyboard Interface when Disconnected

This testing is for testing the key messaging. When enabled, key messages are displayed on the screen. Testing occurs only when disconnected.

Local echo of characters in ASCII/NVT mode

If enabled, when connected in NVT mode, keyboard input is echoed to the screen by the emulator.

Move Cursor to Input field if inserted into a Protected field by host

If enabled, the cursor moves to the first input field if the host positions the cursor to a protected field.

Set End of Line as End of Field in an Unformatted Screen Image

If enabled, the emulator terminates a line marked with an End of Field attribute. This option is used only when the emulator is receiving unformatted data.

Auto Capture on Attention Key

If enabled, pressing the Attention Key appends the screen image to the capture file. Pressing Attention is the same as clicking *File - Capture Screen Contents*.

Auto Save Session on Exit

Automatically saves the session and changes to the configuration upon exit.

Confirm Disconnect

If selected, prompts the user to confirm the action before disconnecting.

Bell

Toggles the “ringing” of the bell sent by the host.

Override Numeric

If enabled, any alphanumeric character can be used in a numeric input field.

Auto Skip Field

Automatically skips to the next input field when the current input field is full.

Strip Leading Space

If enabled, any leading spaces in an input field that has been updated are stripped before the field is sent to the host.

Blinking Cursor

Toggles the blinking of the cursor.

Erase on Backspace

Sets the Backspace Key to erase (delete) characters.

Strip Zero

Strips empty fields preventing them from being sent to the host.

PRINT FONT SETUP

Enable Highlight

If enabled, highlighted text will be highlighted when printed.

Printer Font Height

Set the height of the printer font.

ACCELERATOR KEY OPERATION

Specifies the operation of the accelerator keys.

Off

Uses the mouse to make menu bar selections. All default emulator key definitions are available when this option is selected.

Alt Key Only

Pressing and releasing the Alt key moves the cursor up to the menu bar. The arrow keys or the underlined letter can then be used to select the option of interest.

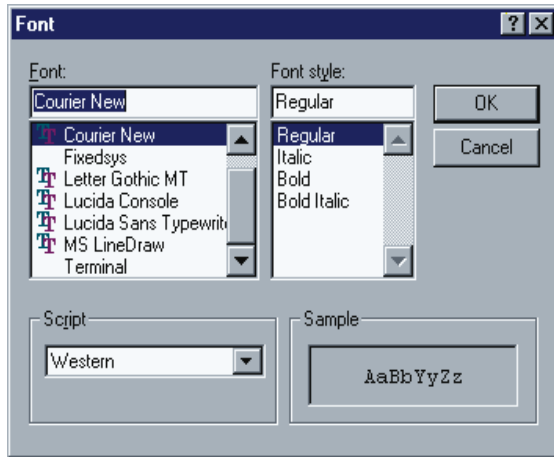
Alt Key + Letter Key

Pressing the Alt key plus the underlined letter of the menu bar option displays the associated drop down menu or dialog box.

10

FONT

Select from any TrueType font installed on your system.



Font

Select a font from the list of fonts installed on your system.

Font Style

Select a font style from the list of: Regular, Oblique, Bold, Bold Oblique.

Script

Displays the type of font selected.

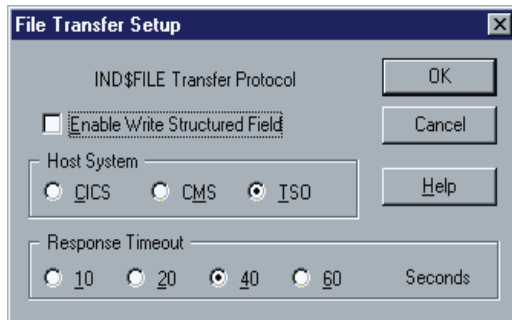
Sample

Displays a sample line of text using the selected font.

11

FILE TRANSFER

Prior to doing a file transfer, make sure the correct operating system is selected in the File Transfer Setup dialog box:



SETTING FILE TRANSFER

1. Transfer files using the IND\$FILE protocol.
2. Set host system, file transfer timeout and enable/disable Write Structured Field.
3. Set specifications:

Enable Write Structured Field

Enabling the write structured field results in a faster file transfer. This method requires the display session on the host to be configured for extended data stream. The emulator must also be set for extended attributes.

Host System

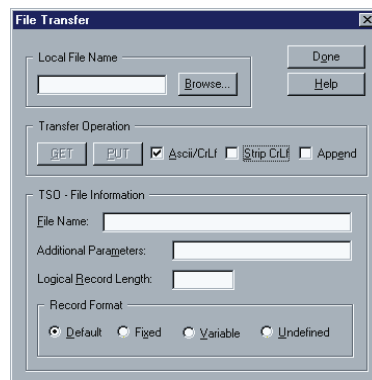
The IBM host systems available are: CICS, CMS and TSO.

Response Timeout

Select a timeout response after a transfer has been initiated. If no response is received before the specified timeout, an error occurs and the transfer is aborted.

TRANSFERRING A FILE

To transfer a file select *Transfer, File Transfer* from the menu bar. The dialog box shown below will appear:



1. Enter or select a local file name from the PC. When sending a file (PUT) select the file to be sent. When receiving a file select (GET) the file on the PC.
2. Select the following options for your Transfer Operation:

Ascii/CrLf

Enables ASCII carriage return/linefeed translation on file transfers.

Strip CrLf

Strips the carriage return and line feed from the file being sent to the host. Used only for ASCII files.

Append

If the PC file already exists, the file contents are to be appended to it. Otherwise, the existing file is overwritten by the contents of the transferring file.

3. Enter the name of the host file in the File Information area. When receiving a file, select the file name from the host. When sending a file, select the name of the file to be sent to the host.
4. Fill in the text box *Additional Parameters* for any parameters specific to the IND\$FILE program on your host system. The contents of this text box are appended to the end of the transfer command.
5. Set Logical Record Length for fixed length records or the maximum logical record length for record composed of varying lengths (from 1 - 32760).

This option is ignored when APPEND is enabled.

4. Select the setup options desired. Options not available for the selected operating system will be grayed out.

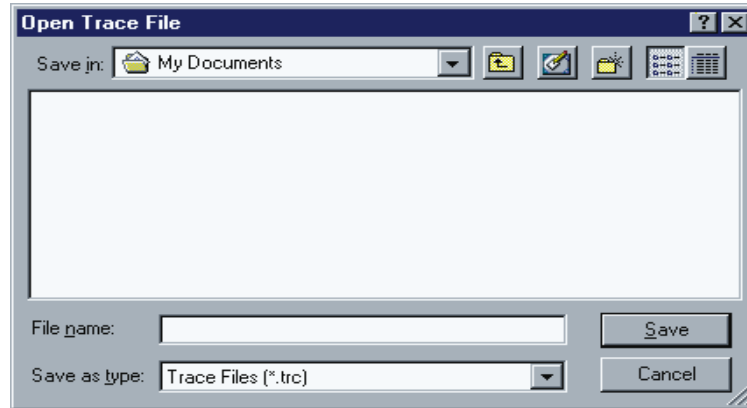
Options Available:

<i>Default</i>	Default format for the operating system.
<i>Fixed</i>	Fixed record lengths. The size of the record is set by the Logical Record Length setup option.
<i>Variable</i>	Variable length records. The maximum size of the record is set by the Logical Record Length setup option.
<i>Undefined</i>	Unknown format.

5. Click on *Get* to receive a file from the host or *Put* to send a file to the PC from the host.

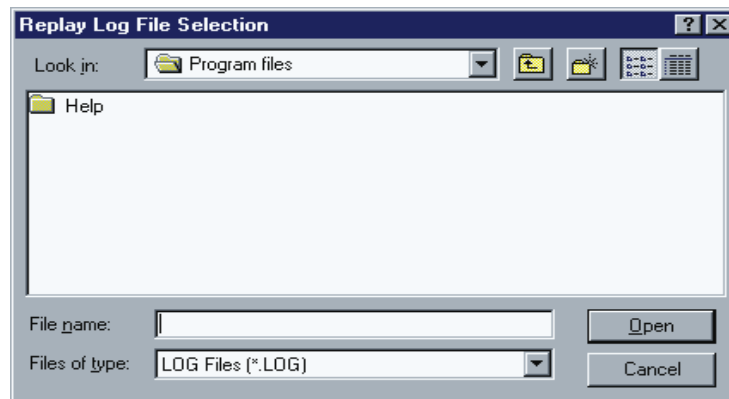
RECORD TRACE FILE

The *Record Trace File* feature traces all data received and sent by the host. The trace is saved to a specified file. During a trace, the letter T appears on the status line.



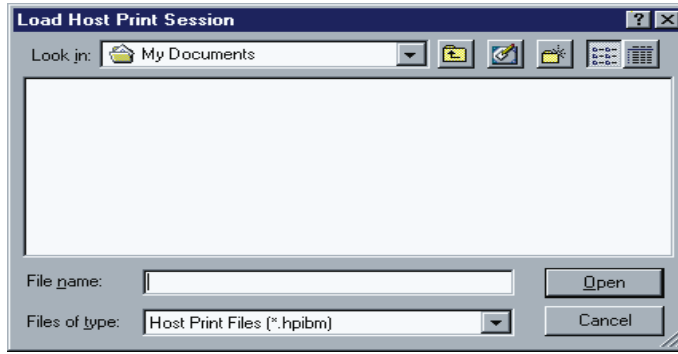
REPLAY LOG FILE SELECTION

This dialog box allows you to replay an existing log file.



OPEN SESSION

The Open Session displays a file open dialog box containing a list of the Host Print session files. When Host Print exits the current session information is saved in a session file named after the host name.



STATUS BAR

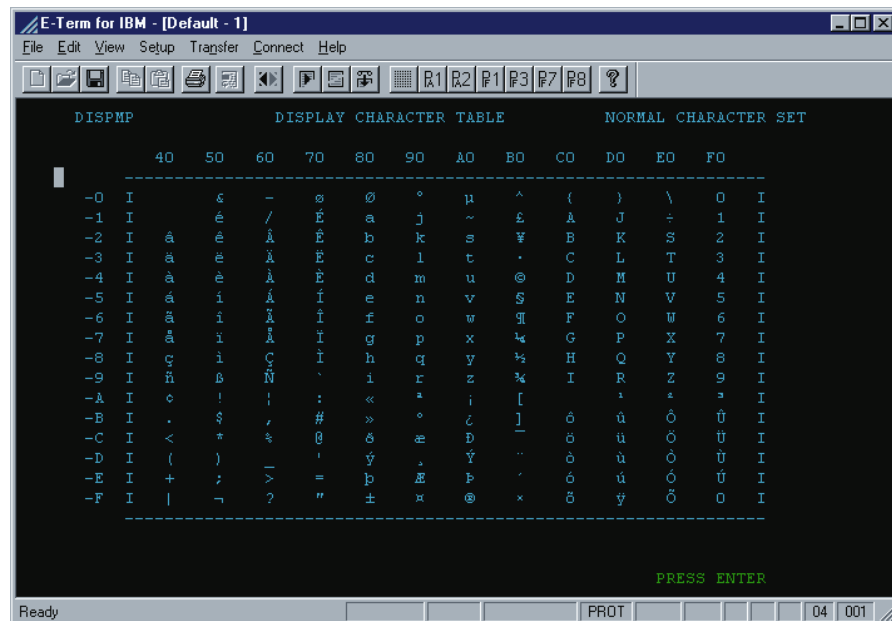
Toggles the display of the Status Bar. The Status Bar (also known as the Status Line) displays the following information from left to right:

- ◆ Temporary status messages.
- ◆ The keyboard lock status. LOCK displays in the pane when the keyboard is locked. The RESET key unlocks the keyboard. Type ahead is supported. A progress-bar control is included.
- ◆ Script/Recording file name.
- ◆ Protected status. PROT displays in the pane when the cursor is in a protected field. Keyboard input is prohibited in protected state.
- ◆ Numeric status. NUM displays in the pane when the cursor is in a numeric field. Only Numeric Characters are allowed. To override, select the Override Numeric option in the Preferences dialog box.
- ◆ When the keyboard is in insert mode, the word INS displays.

- ◆ Character C displays during a capture.
- ◆ Character R displays during a recording.
- ◆ Character T displays when Trace is enabled.
- ◆ Cursor row position.
- ◆ Last pane shows the cursor column position.

APPLICATION MENU

Toggles the display of the application menu. To restore, click on the *Control Menu* in the upper left corner of the application window and select *Show Application Menu*. Or press Ctrl Alt M.



Include Form Feed After

When it is desirable to send multiple screen images to a single “page”, enable this option. The printer will not send the form feed (eject the paper) until after the number of screen images specified have been sent to the printer. Up to three images can be sent to a single page.

Log Replay Speed

The log replay speed ranges from the slowest speed, (400 characters per second) to the fastest (10,000 CPS).

Communication

Enter the host name, the port (default 23) and LU name (optional).

Send Device Ready Continuously When Ready

If enabled, this option continually sends “Device Ready” messages to the host. The messages are sent only when the Host Print application is idle.

Associate LU/Device Name with a Terminal Session

If enabled, the emulator associates the specified LU/Device name with the terminal session. If this option is not selected, the Host Printer application performs a TELNET connect to the specified LU.Device name.

The LU name is used for printing when connection to a specific LU pool is desired. The same LU name must be entered in the Tn3270 tab in the *Connect* dialog box. The LU name is not necessary if a generic pool of LUs are used. If Specific LU printing is used, the system administrator must supply the correct LU name.

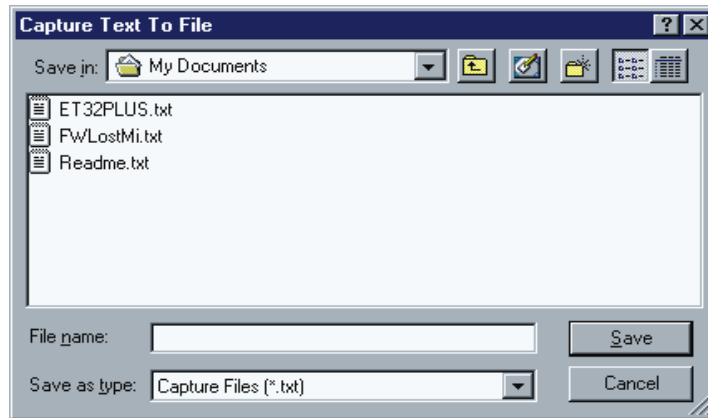
Status Information Log

The Status Information Log is a list box with session log information. The Default print job file name is MMDDHHMM.0nn if print output is not to a file name.

CAPTURING SCREENS

To capture a screen or a set of screens to a file:

1. Click on *File - Capture Text to File*. The *Capture Text To File* dialog box will display:

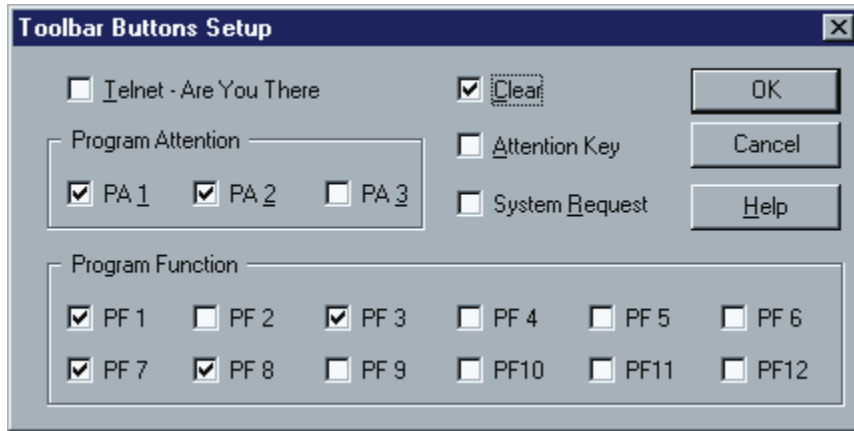


2. Select or enter a filename for the capture file.
3. Click on *File - Capture Screen Contents* to append screen images to the capture file. The menu command changes to *Stop Capture Text to File* and the letter *C* appears in the status line.
4. Click on *File - Stop Capture Text to File* to close the capture file.

TOOLBAR BUTTONS

The Toolbar contains shortcuts for program features. Some toolbar buttons are fixed and cannot be edited. The optional buttons can be added to or removed from the toolbar from the Toolbar Buttons dialog box. Toolbar buttons and their

functions may be enabled/disabled during scripting and recording or as a result of the session state.



INDEX

Symbols

3270 keyboard assignments 3-7
5250 keyboard assignments 3-8
5250 keys 3-5

A

Accelerator key operation 9-4
Accelerator keys 3-10
Application Menu 11-6

C

Color preferences 8-1
Connecting host print session 4-2
Connecting to the host 2-1
Copying a session 2-4
Creating a session 2-3

D

Deleting a session 2-5

E

Edit keys 3-5

F

Features 1-1
File transfer 11-1
 setting transfer 11-1
 transferring a file 11-2
Font options 10-1

H

HLLAPI 7-1
 connecting to Minisoft for IBM 7-1
HLLAPI function reference 7-3
Host print session 4-2
Host print setup 4-2
 print to file 4-4
 retrieving host print session 4-3
 saving host print session 4-3
Host printer command line 4-3
Host printing 4-1

I

IND\$File transfer 1-2

K

Keyboard mapping 3-1

L

Language translation 2-2

M

Mouse button setup 5-2

N

Navigation keys 3-5

O

Open session 11-5

P

- PA keys 3-5
- PF keys 3-5
- Preferences 9-1
- Print font setup 9-4
- Print screen 4-5
- Print setup 4-6

R

- Record log file selection 5-1
- Record trace file 11-4
- Replay log file selection 11-4
- Retrieving host printer session 4-3

S

- Saving host printer session 4-3
- Script files 6-1
 - recording 6-2
 - running 6-1
- Script functions 6-2
- Select keys 3-5
- Session manager 2-3
 - copying a session 2-4
 - creating a session 2-3
 - deleting a session 2-5
- Status bar 11-5
- System requirements 1-3